User's Manual

32

RH850/C1M-A1, RH850/C1M-A2

User's Manual: Hardware

Renesas microcontroller
RH850 Family

Renesas Electronics
www.renesas.com

Rev.1.20 May, 2018

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
6. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1 November 2017)

Notes for CMOS devices

(1) Voltage application waveform at input pin:

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between VIL (MAX) and VIH (MIN) due to noise, etc., the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between VIL (MAX) and VIH (MIN).

(2) Handling of unused input pins:

Unconnected CMOS device inputs can be cause of malfunction. If an input pin is unconnected, it is possible that an internal input level may be generated due to noise, etc., causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using pull-up or pull-down circuitry. Each unused pin should be connected to power supply or GND via a resistor if there is a possibility that it will be an output pin. All handling related to unused pins must be judged separately for each device and according to related specifications governing the device.

(3) Precaution against ESD:

A strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it when it has occurred. Environmental control must be adequate. When it is dry, a humidifier should be used. It is recommended to avoid using insulators that easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors should be grounded. The operator should be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with mounted semiconductor devices.

(4) Status before initialization:

Power-on does not necessarily define the initial status of a MOS device. Immediately after the power source is turned ON, devices with reset functions have not yet been initialized. Hence, power-on does not guarantee output pin levels, I/O settings or contents of registers. A device is not initialized until the reset signal is received. A reset operation must be executed immediately after power-on for devices with reset functions.

(5) Power ON/OFF sequence:

In the case of a device that uses different power supplies for the internal operation and external interface, as a rule, switch on the external power supply after switching on the internal power supply. When switching the power supply off, as a rule, switch off the external power supply and then the internal power supply. Use of the reverse power on/off sequences may result in the application of an overvoltage to the internal elements of the device, causing malfunction and degradation of internal elements due to the passage of an abnormal current. The correct power on/off sequence must be judged separately for each device and according to related specifications governing the device.

(6) Input of signal during power off state:

Do not input signals or an I/O pull-up power supply while the device is not powered. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Input of signals during the power off state must be judged separately for each device and according to related specifications governing the device.

How to Use This Manual

Readers This manual is intended for users who wish to understand the functions of the RH850/C1M-A and design application systems using the following RH850/C1M-A microcontrollers:

Purpose This manual is intended to give users an understanding of the hardware functions of the RH850/C1M-A shown in the *Organization* below.

Organization This manual is divided into two parts: Hardware (this manual) and Architecture (RH850G3MH User's Manual: Software (R01US0143E)).

Hardware	Software
Pin functions	Overview
CPU function	Processor Model
On-chip peripheral functions	Register Reference
Flash memory programming	Exceptions and Interrupts
	Memory Management
	Instruction Reference
	Reset
	Appendix

How to read this manual It is assumed that the readers of this manual have general knowledge in the fields of electrical engineering, logic circuits, and microcontrollers.

To understand the overall functions of the RH850/C1M-A.

→ Read this manual according to the Contents.

To understand the details of an instruction function

→ See RH850G3MH Family User's Manual: Software (R01US0143E) available separately.

Conventions

Data significance: Higher digits on the left and lower digits on the right

Active low representation: xxx (overscore over pin or signal name)

Memory map address: Higher addresses on the top and lower addresses on the bottom

Note: Footnote for item marked with Note in the text

Caution: Information requiring particular attention

Remark: Supplementary information

Numeric representation: Binary ... xxxx or $xxxx_B$

Decimal ... xxxx

Hexadecimal ... $xxxx_H$

Prefix indicating power of 2 (address space, memory capacity):

K (kilo): $2^{10} = 1,024$

M (mega): $2^{20} = 1,024^2$

G (giga): $2^{30} = 1,024^3$

Description of Registers

Each register description includes register access, register address, and register value after a reset, a bit chart, illustrating the arrangement of bits, and a table of bits, describing the meaning of the bit settings.

The standard format for bit charts and tables are described below.

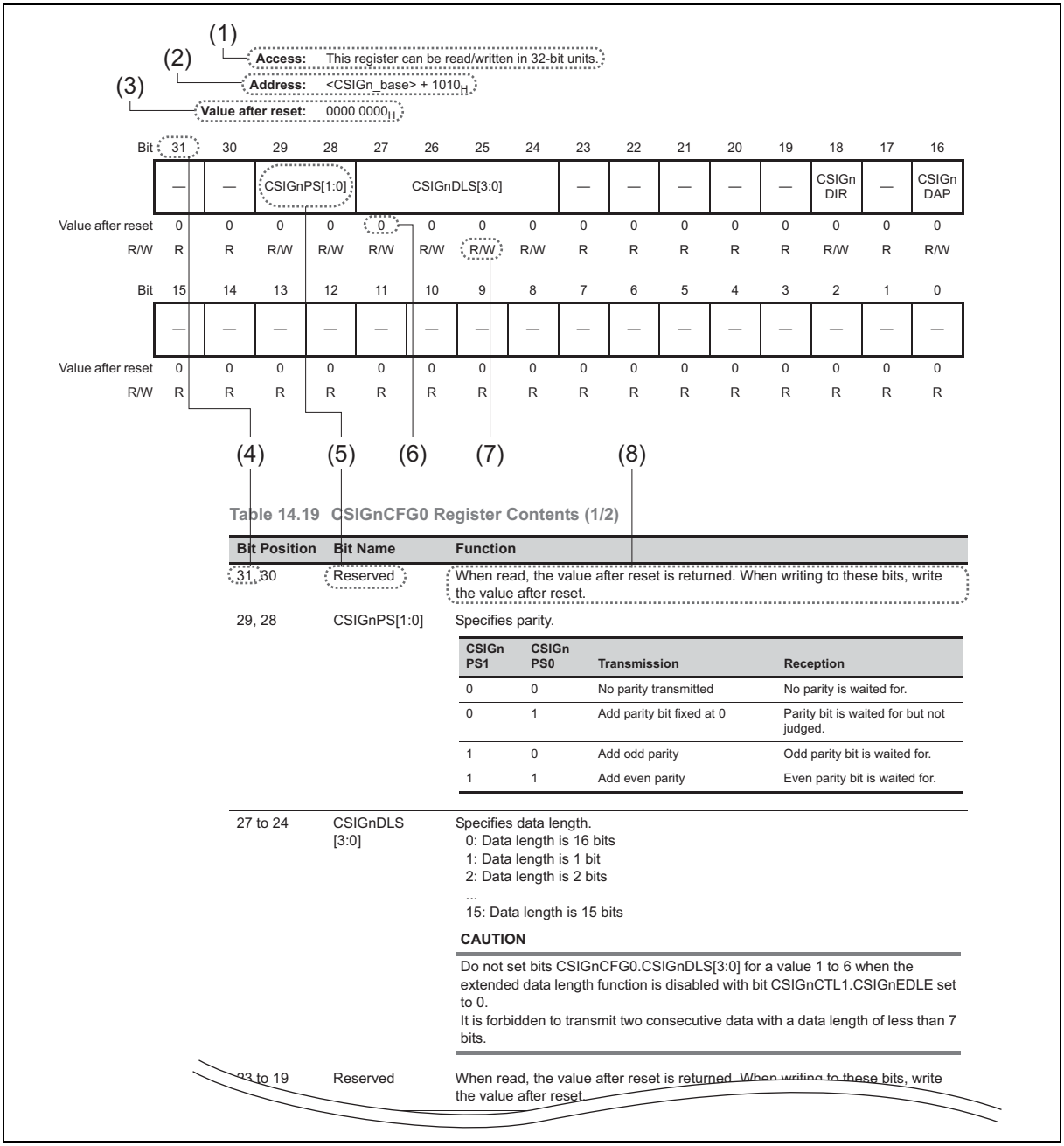


Table 14.19 CSIGnCFG0 Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
29, 28	CSIGnPS[1:0]	Specifies parity.
27 to 24	CSIGnDLS [3:0]	Specifies data length. 0: Data length is 16 bits 1: Data length is 1 bit 2: Data length is 2 bits ... 15: Data length is 15 bits
23 to 19	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

CSIGn PS1	CSIGn PS0	Transmission	Reception
0	0	No parity transmitted	No parity is waited for.
0	1	Add parity bit fixed at 0	Parity bit is waited for but not judged.
1	0	Add odd parity	Odd parity bit is waited for.
1	1	Add even parity	Even parity bit is waited for.

CAUTION

Do not set bits CSIGnCFG0.CSIGnDLS[3:0] for a value 1 to 6 when the extended data length function is disabled with bit CSIGnCTL1.CSIGnEDLE set to 0.
It is forbidden to transmit two consecutive data with a data length of less than 7 bits.

(1) Access

The register can be accessed in the bit unit indicated here.

(2) Address

This is the register address.
For base address, see description of base address in each section.

(3) Value after a reset (in hexadecimal notation)

This is the value of all bits of the register after a reset. Values for bytes are given as numbers in the range from 0 to 9 and letters from A to F or as X where they are undefined.

(4) Bit position

This is the bit number.
The bits are numbered from 31 to 0 for 32-bit registers, 15 to 0 for 16-bit registers, and 7 to 0 for 8-bit registers.

(5) Bit name

Bit name or field name is indicated.
When clearly identifying the digits of a bit field is required, do so by using a form such as CSIGnDLS[3:0] above.
Indicate reserved bits by using a dash (—).

(6) Value after a reset (in binary notation)

This is the bit values after a reset.

0	:	The value after a reset is 0.
1	:	The value after a reset is 1.
—	:	The value after a reset is undefined.

(7) R/W

This is the bit attribute of all bits of the register.

R/W	:	The bit or field is readable and writable.
R	:	The bit or field is readable. Note that all reserved bits are indicated as R. When written, the value specified in the bit chart or the value after a reset should be written. In case of writing to writable registers that also include non-reserved bits with the R-attribute, writing to the R-attribute bits will be ignored unless otherwise specified.
W	:	This bit or field is writable. When read, the value is undefined. If a value is indicated in the bit chart, the value is returned.

(8) Function

This is function of the bit.

Table of Contents

Section 1 Overview

The RH850/C1M-A is a series of single-chip microcomputers in the RH850 Family from Renesas Electronics. This section covers the features of the RH850/C1M-A.

1.1 Features of RH850/C1M-A Products

The CPU of these products is the G3MH of the RH850 Family running at 320 or 240 MHz for high-speed processing.

These products incorporate ROM, RAM, DMA, various timers including timers for motor control, various serial interfaces including a CAN interface, and a 12-bit A/D converter (ADCC). They also incorporate the set of peripheral functions, which are an R/D converter (RDC3A) that converts the signal output by a resolver into digital values representing angles, and a motor control unit (EMU3) that is capable of operating in parallel with the CPU, is optimized for HEV and EV motor control.

Two products, the one for controlling two motors and the other for controlling one motor, are available. The products are provided in 252-pin BGA (for two-motor control) and 176-pin QFP (for one-motor control) packages.

Applications

Automotive (motor control for HEVs and EVs)

1.1.1 Functions of the RH850/C1M-A

Table 1.1 Overview of Products (1/2)

Item		C1M-A2	C1M-A1
CPU	CPU system	G3MH (LSDC*1) + G3MH	G3MH (LSDC*1)
	CPU frequency	320 MHz	240 MHz
	PE internal peripheral device protection function (IPG)	Provided	Provided
	System error notification control function (SEG)	Provided	Provided
	Memory protection unit (MPU)	Provided	Provided
	Floating-point unit (FPU)	Provided	Provided
	Mutual Exclusion Control Registers (MEV)	Provided	Not provided
On-chip memory	Code Flash	2 MB × 2	2 MB
	Instruction cache (Icache)	8 KB × 2	8 KB
	Local RAM	64 KB × 2	64 KB
	Data Flash	64 KB	64 KB
	Global RAM	128 KB	64 KB
External Interrupts	Maskable interrupt (IRQ)	8	8
DMA, DTS		16 channels, 128 channels	16 channels, 128 channels
Clock	Main oscillator (main OSC)	20 MHz	20 MHz
	PLL	Provided	Provided
Security	Intelligence cryptographic unit E (ICUSE)	Provided	Provided
	Secure watchdog timer A (SWDTA)	2	1
I/O ports		99	81
Timers	Timer array unit D (TAUD)	4 units	2 units
	Timer array unit J (TAUJ)	2 units	1 unit
	Motor control timer (TSG3)	3 units	2 units
	Timer option (TAPA)	6 units	4 units
	Timer pattern buffer (TPBA)	2 units	1 unit
	OS timer (OSTM)	4 units	3 units
	Encoder timer (ENCA)	2 units	2 units
	Watchdog timer (WDTA)	2 units	1 unit
Serial interface	Clocked Serial Interface H (CSIH)	3 channels	3 channels
	CAN interface (RS-CANFD)	4 channels	4 channels
	LIN interface (RLIN3)	3 channels	3 channels
	Serial Communication Interface (SCI3)	3 channels	3 channels
	RSENT (Single Edge Nibble Transmission)	4 channels	4 channels

Table 1.1 Overview of Products (2/2)

Item		C1M-A2	C1M-A1
A/D converter	12-bit A/D core	3 units	3 units
	ADCC0 : Number of input pins	16	11
	ADCC0 : Number of T&H	6	6
	ADCC1 : Number of input pins	16	14
	ADCC1 : Number of T&H	6	6
	ADCC2 : Number of input pins	16	5
	ADCC2 : Number of T&H	4	4
Motor control	R/D converter (RDC3A)	2 units	1 unit
	Enhanced motor control unit (EMU3): Number of units	1 unit (2 channels)	1 unit (2 channels)
	Enhanced motor control unit (EMU3): SubCPU frequency	320 MHz	240 MHz
Other functions	Error control module (ECM)	Provided	Provided
	Clock Monitor (CLMA)	Provided	Provided
	Data CRC (DCRA)	2 units	2 units
	Error correction coding (ECC)	Provided	Provided
	On-chip debug (OCD)	Provided	Provided
	Boundary scan	Provided	Provided
	Peripheral interconnection 1 (PIC1B)	2 units	1 unit
	Peripheral interconnection 2 (PIC2D)	1 unit	1 unit
Power supply voltage	Internal power supply	1.25 V ± 0.1 V	1.25 V ± 0.1 V
	I/O power supply	5.0 V ± 0.5 V	5.0 V ± 0.5 V
	R/D converter power supply	5.0 V ± 0.5 V	5.0 V ± 0.5 V
	A/D converter power supply	5.0 V ± 0.5 V	5.0 V ± 0.5 V
Temperature	Junction temperature (Tj)	– 40°C to 150°C	– 40°C to 150°C
Package		252-pin BGA	176-pin QFP

Note 1. LSDC (Lock Step Dual Core)

Table 1.2 List of Products

Group Name	Part Number	Package
RH850/C1M-A2	R7F701275EABG	252-pin plastic BGA (0.8-mm ball pitch) (17 × 17 mm)
RH850/C1M-A1	R7F701278EAFP	176-pin plastic QFP (0.5-mm pin pitch) (24 × 24 mm)

1.1.2 For Development and Debugging

Table 1.3 List of Development Tools

Function	Outline
On-chip debug (OCD)	On-chip debug interfaces • IEEE 1149.1 standard JTAG interface for NEXUS class 3 compliant debugging • Low pin debug (LPD) interface: 4 pins
On-chip debug emulator	E1 emulator
RAM monitor	Advanced user debugger II RAM monitor function
	NEXUS RAM monitor function
Compiler/Debugger	CubeSuite+
	MULTI environment from GreenHills
Application support hardware	RH850 evaluation platform
Flash programming	PG-FP5 flash programmer
	RFP (Renesas flash programmer) + E1 emulator
	Self-programming library
Software tool (optional)	AUTOSAR MCAL

1.1.3 Internal Block Diagram

CPU1, CPU2, and the SubCPU include their own CPU peripherals. They can access only their own CPU peripherals. The same address is assigned to all the CPU peripherals of CPU1, CPU2, and the SubCPU, but CPU1 always accesses the CPU peripheral of CPU1, CPU2 always accesses the CPU peripheral of CPU2, and the SubCPU always accesses the CPU peripheral of the SubCPU.

For the peripheral modules in each peripheral group, see **Section 3.1.2, Configuration of Peripheral Groups.**

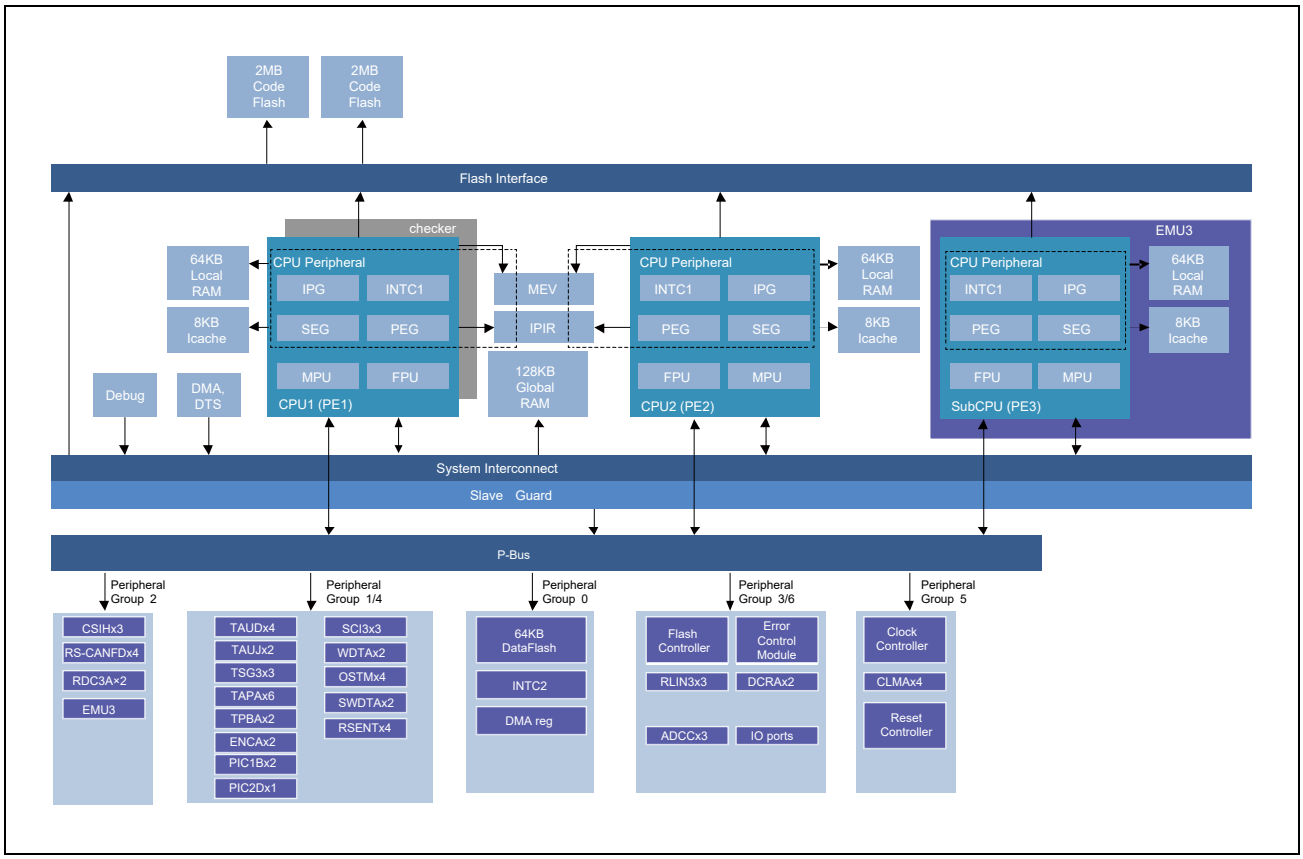


Figure 1.1 Internal Block Diagram of RH850/C1M-A2

CPU1 and the SubCPU include their own CPU peripherals. They can access only their own CPU peripherals.

For the peripheral modules in each peripheral group, see **Section 3.1.2, Configuration of Peripheral Groups**.

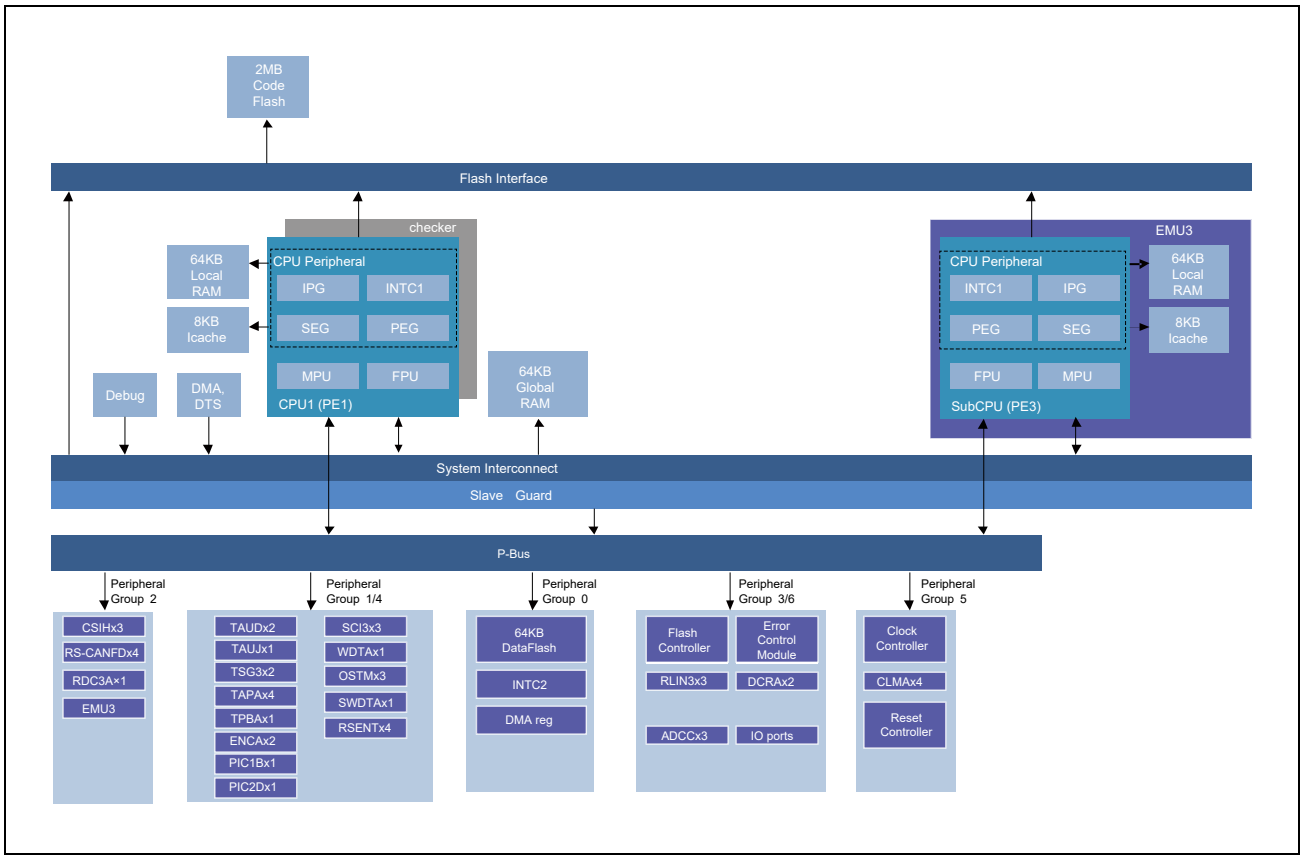


Figure 1.2 Internal Block Diagram of RH850/C1M-A1

1.2 Pin Connection Diagram (Top View)

1.2.1 RH850/C1M-A2 (252-Pin BGA)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
A	A0VSS (N.C.)	A0VSS (N.C.)	ADCC0I01	ADCC0I10	ADCC0I13	ADCC0I22	RDC3A1S2	RDC3A1RSO	RDC3A0RSO	RDC3A0S2	ADCC1I31	ADCC1I03	ADCC1I10	ADCC1I21	ADCC1I23	A2VSS	A2VCC	ADCC2I00	A2VSS (N.C.)	A2VSS (N.C.).	A
B	A0VSS (N.C.)	A0VREFH	ADCC0I00	ADCC0I03	ADCC0I12	ADCC0I21	RDC3A1S1	RDC3A1COM	RDC3A0COM	RDC3A0S1	ADCC1I30	ADCC1I01	ADCC1I11	ADCC1I22	ADCC1I33	A1VREFH	A2VREFH	ADCC2I22	ADCC2I01	A2VSS (N.C.)	B
C	ADCC0I32	ADCC0I33	ADCC0I23	ADCC0I02	ADCC0I11	ADCC0I20	RDC3A1S3	RVSS	RVCC	RDC3A0S3	ADCC1I00	ADCC1I02	ADCC1I13	ADCC1I32	A1VSS	A1VCC	ADCC2I20	ADCC2I21	ADCC2I02	ADCC2I03	C
D	ADCC0I30	ADCC0I31	A0VSS				RDC3A1S4	RVSS	RVCC	RDC3A0S4	VDD	VSS	ADCC1I12	ADCC1I20				ADCC2I31	ADCC2I23	ADCC2I30	D
E	P7_2	P7_0	A0VCC															ADCC2I33	ADCC2I32	ADCC2I10	E
F	P7_5	P7_3	P7_1															ADCC2I11	ADCC2I12	ADCC2I13	F
G	P7_4	P7_7	P7_6	VDD													VSS	P3_7	P3_6	P3_5	G
H	P5_0	P5_2	P5_1	VSS													VDD	P3_3	P3_2	P3_4	H
J	P5_3	P5_4	P5_6	P5_5					VDD	VSS	VSS	VDD					VSS	P2_7	P3_1	P3_0	J
K	P5_7	P5_8	P5_9	VCC					VDD	VSS	VSS	VDD					VCC	P2_6	P2_5	P2_4	K
L	P4_0	P4_1	P4_2	VSS					VDD	VSS	VSS	VDD					P2_1	P2_2	P2_3	P2_0	L
M	P4_3	P4_5	P4_4	VSS					VDD	VSS	VSS	VDD					P1_15	P1_12	P1_13	P1_14	M
N	P4_6	P4_7	P4_8	VDD													VSS	P1_9	P1_10	P1_11	N
P	P4_9	P4_10	P4_13	P4_12													VDD	P1_6	P1_7	P1_8	P
R	P4_11	P4_14	VDD															P1_3	P1_4	P1_5	R
T	P4_15	P6_10	VSS															P1_1	P1_0	P1_2	T
U	P6_11	$\overline{\text{AUDRST}}$	ADUCK				VDD	VSS	VSS	VCC	VSS	VDD	MD1	VSS				P6_8	P0_5	P6_9	U
V	P6_12	$\overline{\text{AUDSYNC}}$	AUDATA3	AUDATA1	P6_2	P6_3	P6_6	P0_8	P0_12	P0_15	VSS	VDD	$\overline{\text{DCUTRST}}$	VSS	SYSVCC	VCC	VSS	P0_2	P0_3	P0_4	V
W	VSS (N.C.)	P6_13	AUDATA2	AUDATA0	P6_0	P6_4	$\overline{\text{ERRROUT\_M}}$	P0_9	P0_11	P0_14	DCUTDO	DCUTDI	DCUTMS	P7_8	SYSVCC	$\overline{\text{RESET}}$	FLMODE	P0_0	P6_7	VSS (N.C.)	W
Y	VSS (N.C.)	VSS (N.C.)	P6_14	P6_15	P6_1	P6_5	P0_6	P0_7	P0_10	P0_13	$\overline{\text{DCURDY}}$	DCUTCK	VSS	X2	X1	VCC	MD0	P0_1	VSS (N.C.)	VSS (N.C.)	Y
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	

Figure 1.3 Pin Connections of RH850/C1M-A2

CAUTION

Though the pins indicated with the names of power supply pins (and as N.C.) do not affect operation of the microcontroller even if they are open-circuit, we recommend connecting the pins to the power supply of the same name as those that do not include N.C. to ensure that the power supply is stable.

In addition, be sure to mount solder balls on the substrate. The pins indicated with the names of power supply pins (and as N.C.) are internally connected with the power supply of the same name as those that do not include N.C.

Table 1.4 Pin Assignments of RH850/C1M-A2 (1/6)

Pin Number	Pin Name
1A	A0VSS(N.C.)
1B	A0VSS(N.C.)
1C	ADCC0I32
1D	ADCC0I30
1E	P7_2/TAUD0I7/TAUD0O7/TAUJ0I2/TAUJ0O2/SCI0RXD/CSIH2SO
1F	P7_5/CSIH2CSS0
1G	P7_4/TAUD0I8/TAUD0O8/TAUJ0I3/TAUJ0O3/SCI0TXD/CSIH2SI
1H	P5_0/RLIN32RX/SCI0RXD/CSIH2CSS3
1J	P5_3/RLIN31TX/SCI1SCK
1K	P5_7/SCI2SCK
1L	P4_0/CSIH1SI
1M	P4_3/CAN0RX/CSIH1CSS0
1N	P4_6/CAN1TX/CSIH0CSS3/CSIH1CSS3
1P	P4_9/CSIH0SC
1R	P4_11/CAN2TX/CSIH0CSS1
1T	P4_15/CAN3TX/ $\overline{\text{ERROROUT\_C}}$
1U	P6_11/TAUD3I11/TAUD3O11
1V	P6_12/TAUD3I12/TAUD3O12
1W	VSS(N.C.)
1Y	VSS(N.C.)
2A	A0VSS(N.C.)
2B	A0VREFH
2C	ADCC0I33
2D	ADCC0I31
2E	P7_0/TAUD0I5/TAUD0O5/TAUJ0I0/TAUJ0O0/ADCC1TRG/CSIH2SC
2F	P7_3/ENCA1TIN1/CSIH2CSS2
2G	P7_7
2H	P5_2/RLIN31RX/SCI0SCK/CSIH2RYI/CSIH2RYO
2J	P5_4/RLIN30RX/SCI1RXD
2K	P5_8/SCI2RXD
2L	P4_1/CSIH1SO
2M	P4_5/CAN1RX/CSIH0CSS2/CSIH1CSS2
2N	P4_7/CSIH0SI/CSIH1SSI
2P	P4_10/TPBA0O/CAN2RX/CSIH0CSS0
2R	P4_14/CAN3RX
2T	P6_10/TAUD3I10/TAUD3O10
2U	$\overline{\text{AUDRST}}$
2V	$\overline{\text{AUDSYNC}}$
2W	P6_13/TAUD3I13/TAUD3O13
2Y	VSS(N.C.)
3A	ADCC0I01
3B	ADCC0I00
3C	ADCC0I23
3D	A0VSS

Table 1.4 Pin Assignments of RH850/C1M-A2 (2/6)

Pin Number	Pin Name
3E	A0VCC
3F	P7_1/ENCA1TIN0/ADCC0TRG/CSIH2CSS1
3G	P7_6
3H	P5_1/RLIN32TX/SCI0TXD/CSIH2SSI
3J	P5_6/TPBA1O/TAPA0ESO
3K	P5_9/SCI2TXD
3L	P4_2/CSIH1SC
3M	P4_4/CAN0TX/CSIH1CSS1
3N	P4_8/CSIH0SO/CSIH1RYI/CSIH1RYO
3P	P4_13/RLIN30TX/CSIH0RYI/CSIH0RYO
3R	VDD
3T	VSS
3U	AUDCK
3V	AUDATA3
3W	AUDATA2
3Y	P6_14/TAUD3I14/TAUD3O14
4A	ADCC0I10
4B	ADCC0I03
4C	ADCC0I02
4G	VDD
4H	VSS
4J	P5_5/RLIN30TX/SCI1TXD/$\overline{\text{ERROROUT\_C}}$
4K	VCC
4L	VSS
4M	VSS
4N	VDD
4P	P4_12/RLIN30RX/CSIH0SSI
4V	AUDATA1
4W	AUDATA0
4Y	P6_15/TAUD3I15/TAUD3O15
5A	ADCC0I13/RDC3A1COSMNT
5B	ADCC0I12/RDC3A1SINMNT
5C	ADCC0I11
5V	P6_2/TAUD3I2/TAUD3O2/ENCA1TIN0/RDC3A0_OUT_W/ADCC0TRG
5W	P6_0/TAUD3I0/TAUD3O0/TAUJ1I2/TAUJ1O2/$\overline{\text{ERROROUT\_C}}$
5Y	P6_1/TAUD3I1/TAUD3O1/TAUJ1I3/TAUJ1O3
6A	ADCC0I22
6B	ADCC0I21
6C	ADCC0I20
6V	P6_3/TAUD3I3/TAUD3O3/RDC3A1_OUT_W
6W	P6_4/TAUD3I4/TAUD3O4/ENCA1TIN1/RDC3A0_OUT_V
6Y	P6_5/TAUD3I5/TAUD3O5/RDC3A1_OUT_V
7A	RDC3A1S2
7B	RDC3A1S1

Table 1.4 Pin Assignments of RH850/C1M-A2 (3/6)

Pin Number	Pin Name
7C	RDC3A1S3
7D	RDC3A1S4
7U	VDD
7V	P6_6/TAUD3I6/TAUD3O6/RDC3A0_OUT_U/TAPA0ESO
7W	$\overline{\text{ERROROUT\_M}}$
7Y	P0_6/TAUD0I6/TAUD0O6/TAUJ0I1/TAUJ0O1/ENCA0E0/RDC3A0_OUT_U/INTP3
8A	RDC3A1RSO
8B	RDC3A1COM
8C	RVSS
8D	RVSS
8U	VSS
8V	P0_8/TAUD0I8/TAUD0O8/TAUJ0I3/TAUJ0O3/ENCA0EC/RDC3A0_OUT_W/INTP5
8W	P0_9/TAUD0I9/TAUD0O9/TAUD3I3/TAUD3O3/TAPA5ESO/INTP6
8Y	P0_7/TAUD0I7/TAUD0O7/TAUJ0I2/TAUJ0O2/ENCA0E1/RDC3A0_OUT_V/INTP4
9A	RDC3A0RSO
9B	RDC3A0COM
9C	RVCC
9D	RVCC
9J	VDD
9K	VDD
9L	VDD
9M	VDD
9U	VSS
9V	P0_12/TAUD0I12/TAUD0O12/TAPA0VP/TAUD3I9/TAUD3O9/RSENT0RX/RSENT0SPCO
9W	P0_11/TAUD0I11/TAUD0O11/TAPA0UN/TAUD3I7/TAUD3O7/TSGTRG
9Y	P0_10/TAUD0I10/TAUD0O10/TAPA0UP/TAUD3I5/TAUD3O5/INTP7
10A	RDC3A0S2
10B	RDC3A0S1
10C	RDC3A0S3
10D	RDC3A0S4
10J	VSS
10K	VSS
10L	VSS
10M	VSS
10U	VCC
10V	P0_15/TAUD0I15/TAUD0O15/TAPA0WN/TAUD3I15/TAUD3O15/RSENT1SPCO
10W	P0_14/TAUD0I14/TAUD0O14/TAPA0WP/TAUD3I13/TAUD3O13/RSENT1RX/RSENT1SPCO
10Y	P0_13/TAUD0I13/TAUD0O13/TAPA0VN/TAUD3I11/TAUD3O11/RSENT0SPCO
11A	ADCC1I31
11B	ADCC1I30
11C	ADCC1I00/RDC3A0SINMNT
11D	VDD
11J	VSS
11K	VSS

Table 1.4 Pin Assignments of RH850/C1M-A2 (4/6)

Pin Number	Pin Name
11L	VSS
11M	VSS
11U	VSS
11V	VSS
11W	DCUTDO
11Y	$\overline{\text{DCURDY}}$
12A	ADCC1I03
12B	ADCC1I01/RDC3A0COSMNT
12C	ADCC1I02
12D	VSS
12J	VDD
12K	VDD
12L	VDD
12M	VDD
12U	VDD
12V	VDD
12W	DCUTDI
12Y	DCUTCK
13A	ADCC1I10
13B	ADCC1I11
13C	ADCC1I13
13D	ADCC1I12
13U	MD1
13V	$\overline{\text{DCUTRST}}$
13W	DCUTMS
13Y	VSS
14A	ADCC1I21
14B	ADCC1I22
14C	ADCC1I32
14D	ADCC1I20
14U	VSS
14V	VSS
14W	P7_8
14Y	X2
15A	ADCC1I23
15B	ADCC1I33
15C	A1VSS
15V	SYSVCC
15W	SYSVCC
15Y	X1
16A	A2VSS
16B	A1VREFH
16C	A1VCC
16V	VCC

Table 1.4 Pin Assignments of RH850/C1M-A2 (5/6)

Pin Number	Pin Name
16W	$\overline{\text{RESET}}$
16Y	VCC
17A	A2VCC
17B	A2VREFH
17C	ADCC2I20
17G	VSS
17H	VDD
17J	VSS
17K	VCC
17L	P2_1/TAUD2I1/TAUD2O1/TSG31O7/INTP1
17M	P1_15/TAUD1I15/TAUD1O15/TAPA1WN/TSG32O6
17N	VSS
17P	VDD
17V	VSS
17W	FLMODE
17Y	MD0
18A	ADCC2I00
18B	ADCC2I22
18C	ADCC2I21
18D	ADCC2I31
18E	ADCC2I33
18F	ADCC2I11
18G	P3_7/TAUD2I15/TAUD2O15/TAPA2WN/ADCC2TRG/RSENT3SPCO
18H	P3_3/TAUD2I11/TAUD2O11/TAPA2UN/ENCA1E0/RDC3A1_OUT_U
18J	P2_7/TAUD2I7/TAUD2O7/TSG31O6/INTP7
18K	P2_6/TAUD2I6/TAUD2O6/TSG31O4/INTP6
18L	P2_2/TAUD2I2/TAUD2O2/TSG31O1/INTP2
18M	P1_12/TAUD1I12/TAUD1O12/TAPA1VP/TAUD1O13/TSG32O5
18N	P1_9/TAUD1I9/TAUD1O9/TSG32O7/TAPA4ESO
18P	P1_6/TAUD1I6/TAUD1O6/TAUD1O7/TSG30O4
18R	P1_3/TAUD1I3/TAUD1O3/TSG30O3
18T	P1_1/TAUD1I1/TAUD1O1/ENCA0TIN1/TSG30O7
18U	P6_8/TAUD3I8/TAUD3O8/TAUJ1I0/TAUJ1O0
18V	P0_2/TAUD0I2/TAUD0O2/TAUJ0I2/TAUJ0O2/TAPA3ESO
18W	P0_0/TAUD0I0/TAUD0O0/TAUJ0I0/TAUJ0O0
18Y	P0_1/TAUD0I1/TAUD0O1/TAUJ0I1/TAUJ0O1/TAPA5ESO
19A	A2VSS(N.C.)
19B	ADCC2I01
19C	ADCC2I02
19D	ADCC2I23
19E	ADCC2I32
19F	ADCC2I12
19G	P3_6/TAUD2I14/TAUD2O14/TAPA2WP/RSENT3RX/RSENT3SPCO
19H	P3_2/TAUD2I10/TAUD2O10/TAPA2UP/ADCC1TRG

Table 1.4 Pin Assignments of RH850/C1M-A2 (6/6)

Pin Number	Pin Name
19J	P3_1/TAUD2I9/TAUD2O9/ADCC0TRG/RSENT2SPCO/TAPA2ESO
19K	P2_5/TAUD2I5/TAUD2O5/TSG31O2/INTP5
19L	P2_3/TAUD2I3/TAUD2O3/TSG31O3/INTP3
19M	P1_13/TAUD1I13/TAUD1O13/TAPA1VN/TSG32O2
19N	P1_10/TAUD1I10/TAUD1O10/TAPA1UP/TAUD1O11/TSG32O1
19P	P1_7/TAUD1I7/TAUD1O7/TSG30O6
19R	P1_4/TAUD1I4/TAUD1O4/TAUD1O5/TSG30O5
19T	P1_0/TAUD1I0/TAUD1O0/ENCA0TIN0/TAUD1O1/TSG30O0
19U	P0_5/TAUD0I5/TAUD0O5/TAUJ0I0/TAUJ0O0/TAPA4ESO/INTP2
19V	P0_3/TAUD0I3/TAUD0O3/TAUJ0I3/TAUJ0O3/CAN2RX/INTP0
19W	P6_7/TAUD3I7/TAUD3O7/RDC3A1_OUT_U/TAPA1ESO
19Y	VSS(N.C.)
20A	A2VSS(N.C.)
20B	A2VSS(N.C.)
20C	ADCC2I03
20D	ADCC2I30
20E	ADCC2I10
20F	ADCC2I13
20G	P3_5/TAUD2I13/TAUD2O13/TAPA2VN/ENCA1EC/RDC3A1_OUT_W/TAPA0ESO
20H	P3_4/TAUD2I12/TAUD2O12/TAPA2VP/ENCA1E1/RDC3A1_OUT_V/ADCC0TRG/TAPA3ESO
20J	P3_0/TAUD2I8/TAUD2O8/RSENT2RX/RSENT2SPCO/TAPA1ESO
20K	P2_4/TAUD2I4/TAUD2O4/TSG31O5/INTP4
20L	P2_0/TAUD2I0/TAUD2O0/TSG31O0/INTP0
20M	P1_14/TAUD1I14/TAUD1O14/TAPA1WP/TAUD1O15/TSG32O4
20N	P1_11/TAUD1I11/TAUD1O11/TAPA1UN/TSG32O3
20P	P1_8/TAUD1I8/TAUD1O8/TAUD1O9/TSG32O0/TAPA2ESO
20R	P1_5/TAUD1I5/TAUD1O5/TSG30O2
20T	P1_2/TAUD1I2/TAUD1O2/TAUD1O3/TSG30O1
20U	P6_9/TAUD3I9/TAUD3O9/TAUJ1I1/TAUJ1O1
20V	P0_4/TAUD0I4/TAUD0O4/TAUD3I1/TAUD3O1/CAN2TX/INTP1
20W	VSS(N.C.)
20Y	VSS(N.C.)

1.2.2 RH850/C1M-A1 (176-Pin QFP)

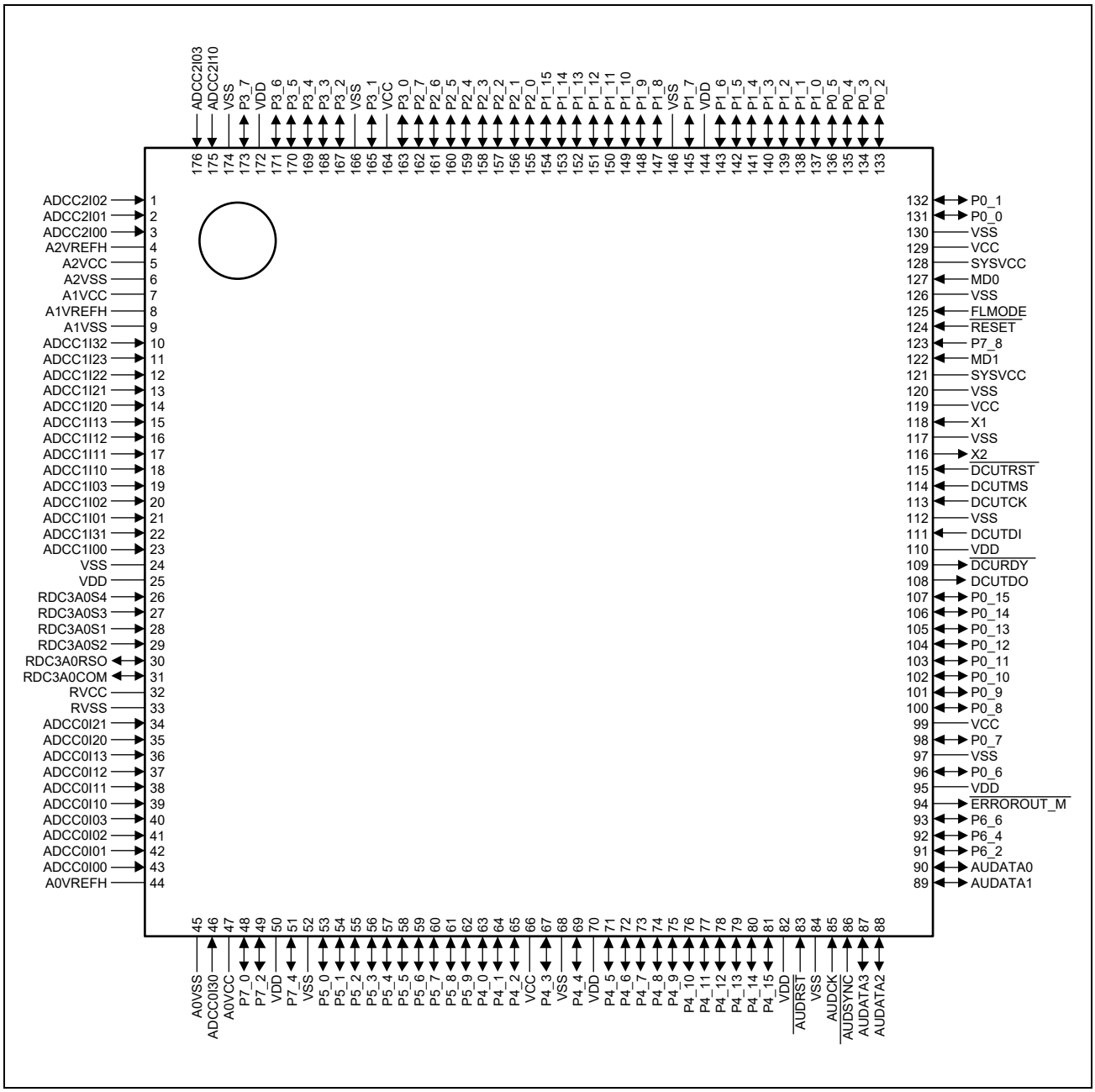


Figure 1.4 Pin Connections of RH850/C1M-A1

Table 1.5 Pin Assignments of RH850/C1M-A1 (1/4)

Pin Number	Pin Name
1	ADCC2I02
2	ADCC2I01
3	ADCC2I00
4	A2VREFH
5	A2VCC
6	A2VSS
7	A1VCC
8	A1VREFH
9	A1VSS
10	ADCC1I32
11	ADCC1I23
12	ADCC1I22
13	ADCC1I21
14	ADCC1I20
15	ADCC1I13
16	ADCC1I12
17	ADCC1I11
18	ADCC1I10
19	ADCC1I03
20	ADCC1I02
21	ADCC1I01/RDC3A0COSMNT
22	ADCC1I31
23	ADCC1I00/RDC3A0SINMNT
24	VSS
25	VDD
26	RDC3A0S4
27	RDC3A0S3
28	RDC3A0S1
29	RDC3A0S2
30	RDC3A0RSO
31	RDC3A0COM
32	RVCC
33	RVSS
34	ADCC0I21
35	ADCC0I20
36	ADCC0I13
37	ADCC0I12
38	ADCC0I11
39	ADCC0I10
40	ADCC0I03
41	ADCC0I02
42	ADCC0I01
43	ADCC0I00
44	A0VREFH

Table 1.5 Pin Assignments of RH850/C1M-A1 (2/4)

Pin Number	Pin Name
45	A0VSS
46	ADCC0I30
47	A0VCC
48	P7_0/TAUD0I5/TAUD0O5/TAUJ0I0/TAUJ0O0/ADCC1TRG/CSIH2SC
49	P7_2/TAUD0I7/TAUD0O7/TAUJ0I2/TAUJ0O2/SCI0RXD/CSIH2SO
50	VDD
51	P7_4/TAUD0I8/TAUD0O8/TAUJ0I3/TAUJ0O3/SCI0TXD/CSIH2SI
52	VSS
53	P5_0/RLIN32RX/SCI0RXD/CSIH2CSS3
54	P5_1/RLIN32TX/SCI0TXD/CSIH2SSI
55	P5_2/RLIN31RX/SCI0SCK/SCI0SCK/CSIH2RYI/CSIH2RYO
56	P5_3/RLIN31TX/SCI1SCK/SCI1SCK
57	P5_4/RLIN30RX/SCI1RXD
58	P5_5/RLIN30TX/SCI1TXD/$\overline{\text{ERROROUT\_C}}$
59	P5_6/TAPA0ESO
60	P5_7/SCI2SCK
61	P5_8/SCI2RXD
62	P5_9/SCI2TXD
63	P4_0/CSIH1SI
64	P4_1/CSIH1SO
65	P4_2/CSIH1SC
66	VCC
67	P4_3/CAN0RX/CSIH1CSS0
68	VSS
69	P4_4/CAN0TX/CSIH1CSS1
70	VDD
71	P4_5/CAN1RX/CSIH0CSS2/CSIH1CSS2
72	P4_6/CAN1TX/CSIH0CSS3/CSIH1CSS3
73	P4_7/CSIH0SI/CSIH1SSI
74	P4_8/CSIH0SO/CSIH1RYI/CSIH1RYO
75	P4_9/CSIH0SC
76	P4_10/TPBA0O/CAN2RX/CSIH0CSS0
77	P4_11/CAN2TX/CSIH0CSS1
78	P4_12/RLIN30RX/CSIH0SSI
79	P4_13/RLIN30TX/CSIH0RYI/CSIH0RYO
80	P4_14/CAN3RX
81	P4_15/CAN3TX/$\overline{\text{ERROROUT\_C}}$
82	VDD
83	$\overline{\text{AUDRST}}$
84	VSS
85	AUDCK
86	$\overline{\text{AUDSYNC}}$
87	AUDATA3
88	AUDATA2

Table 1.5 Pin Assignments of RH850/C1M-A1 (3/4)

Pin Number	Pin Name
89	AUDATA1
90	AUDATA0
91	P6_2/ENCA1TIN0/RDC3A0_OUT_W/ADCC0TRG
92	P6_4/ENCA1TIN1/RDC3A0_OUT_V
93	P6_6/RDC3A0_OUT_U/TAPA0ESO
94	$\overline{\text{ERROROUT\_M}}$
95	VDD
96	P0_6/TAUD0I6/TAUD0O6/TAUJ0I1/TAUJ0O1/ENCA0E0/RDC3A0_OUT_U/INTP3
97	VSS
98	P0_7/TAUD0I7/TAUD0O7/TAUJ0I2/TAUJ0O2/ENCA0E1/RDC3A0_OUT_V/INTP4
99	VCC
100	P0_8/TAUD0I8/TAUD0O8/TAUJ0I3/TAUJ0O3/ENCA0EC/RDC3A0_OUT_W/INTP5
101	P0_9/TAUD0I9/TAUD0O9/INTP6
102	P0_10/TAUD0I10/TAUD0O10/TAPA0UP/INTP7
103	P0_11/TAUD0I11/TAUD0O11/TAPA0UN/TSGTRG
104	P0_12/TAUD0I12/TAUD0O12/TAPA0VP/RSENT0RX/RSENT0SPCO
105	P0_13/TAUD0I13/TAUD0O13/TAPA0VN/RSENT0SPCO
106	P0_14/TAUD0I14/TAUD0O14/TAPA0WP/RSENT1RX/RSENT1SPCO
107	P0_15/TAUD0I15/TAUD0O15/TAPA0WN/RSENT1SPCO
108	DCUTDO
109	$\overline{\text{DCURDY}}$
110	VDD
111	DCUTDI
112	VSS
113	DCUTCK
114	DCUTMS
115	$\overline{\text{DCUTRST}}$
116	X2
117	VSS
118	X1
119	VCC
120	VSS
121	SYSVCC
122	MD1
123	P7_8
124	$\overline{\text{RESET}}$
125	FLMODE
126	VSS
127	MD0
128	SYSVCC
129	VCC
130	VSS
131	P0_0/TAUD0I0/TAUD0O0/TAUJ0I0/TAUJ0O0
132	P0_1/TAUD0I1/TAUD0O1/TAUJ0I1/TAUJ0O1

Table 1.5 Pin Assignments of RH850/C1M-A1 (4/4)

Pin Number	Pin Name
133	P0_2/TAUD0I2/TAUD0O2/TAUJ0I2/TAUJ0O2/TAPA3ESO
134	P0_3/TAUD0I3/TAUD0O3/TAUJ0I3/TAUJ0O3/CAN2RX/INTP0
135	P0_4/TAUD0I4/TAUD0O4/CAN2TX/INTP1
136	P0_5/TAUD0I5/TAUD0O5/TAUJ0I0/TAUJ0O0/TAPA4ESO/INTP2
137	P1_0/TAUD1I0/TAUD1O0/ENCA0TIN0/TAUD1O1/TSG30O0
138	P1_1/TAUD1I1/TAUD1O1/ENCA0TIN1/TSG30O7
139	P1_2/TAUD1I2/TAUD1O2/TAUD1O3/TSG30O1
140	P1_3/TAUD1I3/TAUD1O3/TSG30O3
141	P1_4/TAUD1I4/TAUD1O4/TAUD1O5/TSG30O5
142	P1_5/TAUD1I5/TAUD1O5/TSG30O2
143	P1_6/TAUD1I6/TAUD1O6/TAUD1O7/TSG30O4
144	VDD
145	P1_7/TAUD1I7/TAUD1O7/TSG30O6
146	VSS
147	P1_8/TAUD1I8/TAUD1O8/TAUD1O9
148	P1_9/TAUD1I9/TAUD1O9/TAPA4ESO
149	P1_10/TAUD1I10/TAUD1O10/TAPA1UP/TAUD1O11
150	P1_11/TAUD1I11/TAUD1O11/TAPA1UN
151	P1_12/TAUD1I12/TAUD1O12/TAPA1VP/TAUD1O13
152	P1_13/TAUD1I13/TAUD1O13/TAPA1VN
153	P1_14/TAUD1I14/TAUD1O14/TAPA1WP/TAUD1O15
154	P1_15/TAUD1I15/TAUD1O15/TAPA1WN
155	P2_0/TSG31O0/INTP0
156	P2_1/TSG31O7/INTP1
157	P2_2/TSG31O1/INTP2
158	P2_3/TSG31O3/INTP3
159	P2_4/TSG31O5/INTP4
160	P2_5/TSG31O2/INTP5
161	P2_6/TSG31O4/INTP6
162	P2_7/TSG31O6/INTP7
163	P3_0/RSENT2RX/RSENT2SPCO/TAPA1ESO
164	VCC
165	P3_1/ADCC0TRG/RSENT2SPCO
166	VSS
167	P3_2/ADCC1TRG
168	P3_3/ENCA1E0
169	P3_4/ENCA1E1/ADCC0TRG/TAPA3ESO
170	P3_5/ENCA1EC/TAPA0ESO
171	P3_6/RSENT3RX/RSENT3SPCO
172	VDD
173	P3_7/ADCC2TRG/RSENT3SPCO
174	VSS
175	ADCC2I10
176	ADCC2I03

Section 2 Pins

This section describes the pin and port functions.

The first part of the section describes the port functions.

The remainder describes the port groups, digital noise filter, and respective pin functions.

2.1 Port Functions

2.1.1 Features

Port Group

The RH850/C1M-A provides the following port groups, indicated by the numbers in the table below.

Table 2.1 Port Groups in RH850/C1M-A

Product		Number of Groups	Name of Group
RH850/C1M-A	BGA252/QFP176	8	P0 to P7

Port Group Index n

Each port group is identified by its own index “n” (n = 0 to 7) throughout this section; e.g. PMCn for the port mode control register of the Pn pin.

Base addresses of register base address ports are listed in the following table. Port register addresses are given as offsets from the base addresses in general.

Table 2.2 Register Base Address

Base Address Name	Base Address
<PORT_base>	FFC1 0000_H

2.1.2 Overview

This product has various pins for input/output (I/O) functions, known as ports. The ports are organized in port groups.

This product also has several control registers to allocate the functions other than general-purpose I/O to the corresponding pins.

For definitions of pin, port, and port group, see **Section 2.1.2.1, Terms.**

2.1.2.1 Terms

The terms descried in this section are defined as follows.

- Port group
  A port group consists of a maximum of 16 pins. The number of pins differs depending on the port group. All the pins of a specific port group are controlled by the same port control register.
- Port mode/Port
  A pin in port mode functions as a general-purpose I/O pin. This general-purpose I/O pin is called “port”, and is named as Pn_m. For example, P0_7 indicates port 7 of port group 0.
- Alternative mode
  A pin in alternative mode functions as an I/O pin for peripheral functions. A pin has multiple peripheral functions, and the functions to be used can be selected by control registers.

2.1.2.2 Overview of Pin Functions

Pins can operate in three modes.

- Port mode (PMCn.PMCn_m = 0)
  A pin in port mode operates as a general-purpose input/output pin. The I/O mode is selected by setting the PMn.PMn_m bit.
- Software I/O control alternative mode (PMCn.PMCn_m = 1, PIPCn.PIPCn_m = 0)
  In this mode, the pins operate as alternative functions. The I/O mode is selected by setting the PMn.PMn_m control bit by using software.
- Direct I/O control alternative mode (PMCn.PMCn_m = 1, PIPCn.PIPCn_m = 1)
  In this mode, the pins operate as alternative functions. Unlike the software I/O alternative mode, however, the I/O mode is selected by the alternative function.

Table 2.3 shows the outline of the register settings.

Table 2.3 Pin Function Configuration (Outline)

Mode	Bit			I/O
	PMCn_m	PMn_m	PIPCn_m	
Port mode	0	0	0/1	Output mode
		1	0/1	Input mode*1
Software I/O control alternative mode	1	0	0	Output mode
		1	0	Input mode*2
Direct I/O control alternative mode		0/1	1	Controlled by the alternative function*2

Note 1. The input buffer should be enabled (PIBCn_m = 1).

Note 2. When the pin is used as an input pin in alternative mode, set PIBCn_m to 0.

If a pin is in alternative mode (PMCn.PMCn_m = 1), one of up to seven alternative functions can be selected for that pin by using the PFCn, PFCEn, and PFCAEn registers.

- Software I/O control alternative mode (PIPCn.PIPCn_m = 0):
  - Output (PMn_m = 0): ALT-OUT1 to ALT-OUT7
  - Input (PMn_m = 1): ALT-IN1 to ALT-IN7
- Direct I/O control alternative mode (PIPCn.PIPCn_m = 1):
  - The I/O mode for ALT-OUT1 to ALT-OUT7 and ALT-IN1 to ALT-IN7 is directly selected by the alternative function.

Table 2.4 Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)

Function	Register				I/O
	PFCAE	PFCE	PFC	PM*1	
Alternative output mode 1 (ALT-OUT1)	0	0	0	0	O
Alternative input mode 1 (ALT-IN1)	0	0	0	1	I
Alternative output mode 2 (ALT-OUT2)	0	0	1	0	O
Alternative input mode 2 (ALT-IN2)	0	0	1	1	I
Alternative output mode 3 (ALT-OUT3)	0	1	0	0	O
Alternative input mode 3 (ALT-IN3)	0	1	0	1	I
Alternative output mode 4 (ALT-OUT4)	0	1	1	0	O
Alternative input mode 4 (ALT-IN4)	0	1	1	1	I
Alternative output mode 5 (ALT-OUT5)	1	0	0	0	O
Alternative input mode 5 (ALT-IN5)	1	0	0	1	I
Alternative output mode 6 (ALT-OUT6)	1	0	1	0	O
Alternative input mode 6 (ALT-IN6)	1	0	1	1	I
Alternative output mode 7 (ALT-OUT7)	1	1	0	0	O
Alternative input mode 7 (ALT-IN7)	1	1	0	1	I

Note 1. When PIPCn.PIPCn_m = 1, the I/O direction is directly controlled by the peripheral (alternative) function and PM is ignored.

If a pin is in alternative mode (PMCn.PMCn_m = 1), one of several alternative functions can be selected for that pin by using the PFCn, PFCEn, and PFCAEn registers.

2.1.2.3 Pin Data Input/Output

The registers used for data input and output are described below.

The source of the data to be read via the PPRn register depends on pin mode.

Output data

In port mode (PMCn.PMCn_m = 0), the value of Pn.Pn_m is output from the Pn_m pin.

Input data

A read operation of the PPRn register returns either the value of the Pn_m pin, the associated bit of the port register Pn.Pn_m, or the data output by an alternative function.

The source of the data read via PPRn depends on pin mode and setting of several control bits.

Table 2.5 summarizes the differences of PPRn read modes.

Table 2.5 PPRn_m Read Values

PMCn_m	PMn_m	PIBCn_m	PIPCn_m	Mode	PPRn_m Read Value
0	1	0	X	Port input, input buffer disabled	Pn.Pn_m bit
		1		Port input, input buffer enabled	Pn_m pin
	0	X		Port push-pull output	Pn.Pn_m bit*1
1	1	X	0	Software I/O control alternative input	Pn_m pin
	0			Software I/O control alternative output	Alternative function internal output signal*1
	X		1	Direct I/O control alternative mode	I/O port in alternative mode: • Input: Pn_m pin • Output: Output signal from the alternative function*1

Note 1. When PBDCn_m = 1, the level of the Pn_m pin is returned by the PPRn_m bit.

The control registers in **Table 2.5** have the following effects:

- PMCn.PMCn_m
  This bit selects port mode (PMCn_m = 0) or alternative mode (PMCn_m = 1).
- PMn.PMn_m
  This bit selects input (PMn_m = 1) or output (PMn_m = 0) in port mode (PMCn_m = 0) and software I/O control alternative mode (PMCn_m = 1, PIPCn_m = 0).
- PIBCn.PIBCn_m
  This bit disables (PIBCn_m = 0) or enables (PIBCn_m = 1) the input buffer in input port mode (PMCn_m = 0 and PMn_m = 1). When the input buffer is disabled, PPRn_m reads the Pn.Pn_m bit, otherwise the Pn_m pin level is returned.
- PIPCn.PIPCn_m
  This bit selects the software or direct I/O control alternative mode.
- PBDCn.PBDCn_m
  In output mode, when this bit is set to 1, the pin enters the bidirectional mode. In bidirectional mode, the level of the signal on a Pn_m pin can be read from PPRn.PPRn_m.

Writing to the Pn Register

The data to be output via port Pn_m in port mode (PMCn.PMCn_m = 0) is held in the port register Pn.

Pn data can be overwritten in two ways:

- By writing data directly to the Pn register.
  In this case, new data can be written directly to the Pn register.
- By performing an indirect bitwise operation (a “set”, “reset”, or “not” operation) on the Pn register.
  An indirect bitwise operation (“set”, “reset”, or “not”), can be performed on the Pn register by using the following two registers:
  - Port set/reset register: PSRn
    If the PSRn.PSRn_(m + 16) bit = 1, the value of the Pn.Pn_m bit is determined by the value of the PSRn.PSRn_m bit.
    In other words, the Pn_m bit can be set or reset without writing directly to the Pn register.
  - Port NOT register: PNOTn
    By setting PNOTn.PNOTn_m to 1, the Pn.Pn_m bit can be inverted without writing directly to the Pn register.

An indirect bitwise operation on the Pn register (“set”, “reset”, or “not”), has no effect on the bits that do not need to be updated, allowing you to overwrite only the bit or bits that need to be overwritten.

2.1.3 Port Type

Figure 2.1 shows the overall configuration of the pins. For the details of port blocks, see **Figure 2.2**.

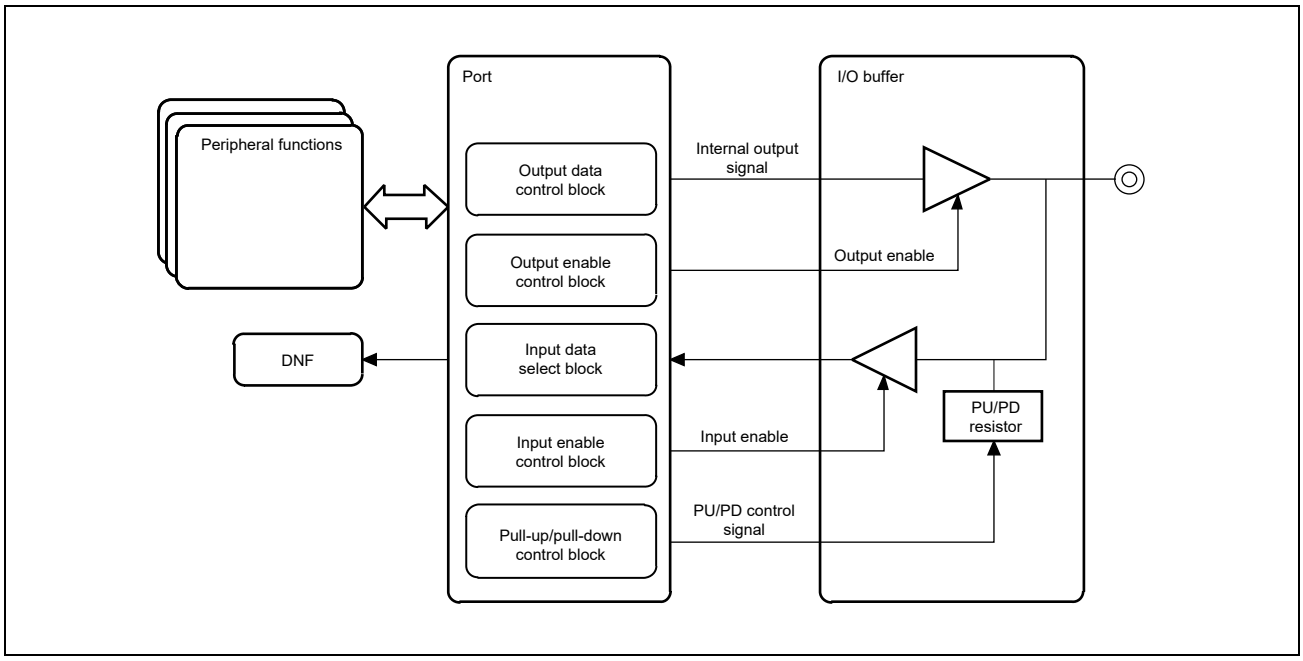
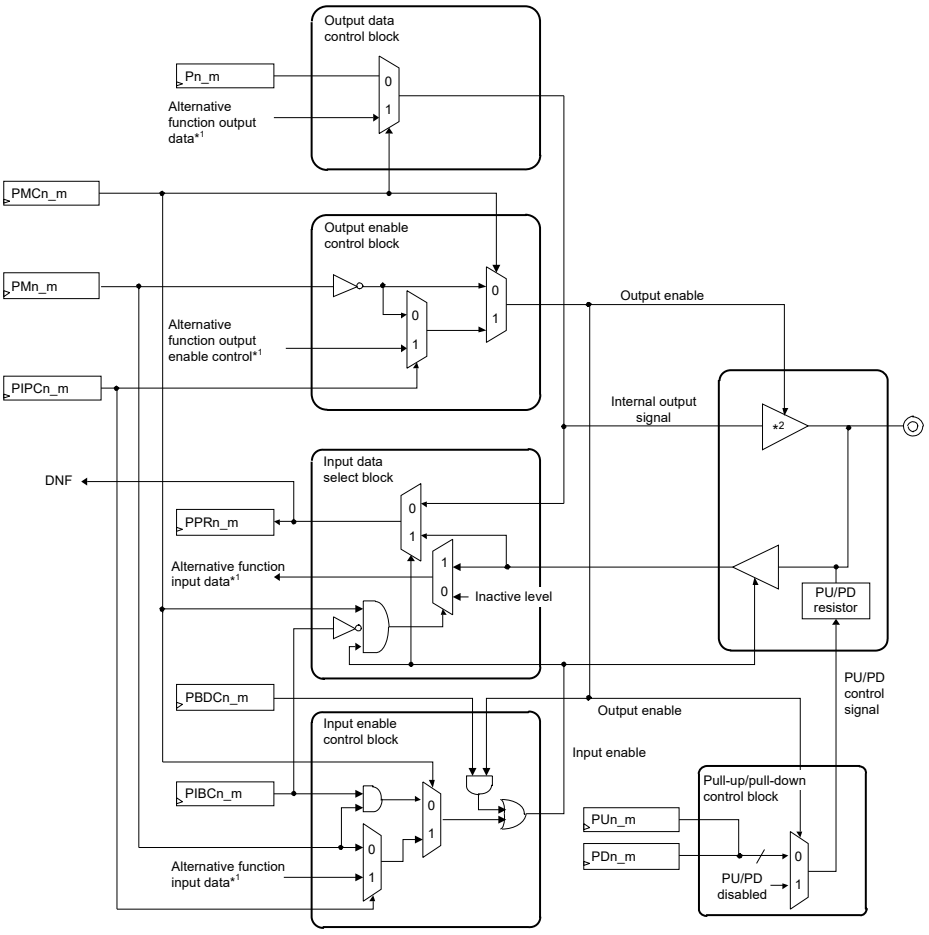


Figure 2.1 Block Diagram of Pin Configuration

Figure 2.2 shows the logical circuitry of the port control functions. The diagram is only a logical reference and does not show the real circuitry.



Note 1. Alternative function signal selected by PFCn_m, PFCEn_m, or PFCAEn_m.

Note 2. Output drive strength can be set by the PDSC and PUCC bits in PCRn_m.

Figure 2.2 Port Control Logic Diagram

2.1.4 Port Group Configuration Register

This section starts with an overview of all configuration registers and then presents all registers in detail.

The configuration registers are classified as follows:

- **2.1.4.2, Configuration of Pin Function**
- **2.1.4.3, Pin Data Input/Output**
- **2.1.4.4, Configuration of Electrical Characteristics Registers**
- **2.1.4.5, Pin-Unit Register**

2.1.4.1 Outline

The following registers are used for the configuration of the individual pins of the port groups:

Table 2.6 Registers for Port Group Configuration

Register Name	Symbol	Register Configuration Unit	Address
Port register	Pn	Port group	<PORT_base> + 0000_H + n × 40_H
Port set/reset register	PSRn	Port group	<PORT_base> + 0004_H + n × 40_H
Port NOT register	PNOTn	Port group	<PORT_base> + 0008_H + n × 40_H
Port pin read register	PPRn	Port group	<PORT_base> + $000C_H$ + n × 40_H
Port mode register	PMn	Port group	<PORT_base> + 0010_H + n × 40_H
Port mode control register	PMCn	Port group	<PORT_base> + 0014_H + n × 40_H
Port function control register	PFCn	Port group	<PORT_base> + 0018_H + n × 40_H
Port function control expansion register	PFCEn	Port group	<PORT_base> + $001C_H$ + n × 40_H
Port mode set/reset register	PMSRn	Port group	<PORT_base> + 0020_H + n × 40_H
Port mode control set/reset register	PMCSRn	Port group	<PORT_base> + 0024_H + n × 40_H
Port function control additional expansion register	PFCAEn	Port group	<PORT_base> + 0028_H + n × 40_H
Port input buffer control register	PIBCn	Port group	<PORT_base> + 4000_H + n × 40_H
Port bidirectional control register	PBDCn	Port group	<PORT_base> + 4004_H + n × 40_H
Port IP control register	PIPCn	Port group	<PORT_base> + 4008_H + n × 40_H
Pull-up option register	PUn	Port group	<PORT_base> + $400C_H$ + n × 40_H
Pull-down option register	PDn	Port group	<PORT_base> + 4010_H + n × 40_H
Port control register	PCRn_m	Pin	<PORT_base> + 2000_H + n × 40_H + m × 4_H

Note: n: Port group number
m: Bit number in a port group

Base address

The base address of PORTn <PORT_base> is defined in Register Base Address in **Section 2.1.1, Features**.

Register value after reset

The value after resets of the registers after reset release depend on the port, and are not described in the following register descriptions, but are given in **Section 2.2.1.1, List of the C1M-A2 Port Registers (BGA), Section 2.2.2.1, List of the C1M-A1 Port Registers (QFP)**.

2.1.4.2 Configuration of Pin Function

(1) PMCn — Port Mode Control Register

This register specifies whether the individual pins of port group n are in port mode or in alternative mode.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 0014_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMCn_15	PMCn_14	PMCn_13	PMCn_12	PMCn_11	PMCn_10	PMCn_9	PMCn_8	PMCn_7	PMCn_6	PMCn_5	PMCn_4	PMCn_3	PMCn_2	PMCn_1	PMCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.7 PMCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PMCn_[15:0]	Specify the operation mode of the corresponding pin: 0: Port mode 1: Alternative mode

(2) PMCSRn — Port Mode Control Set/Reset Register

This register provides an alternative method to write data to a bit in the PMCn register.

The 16 higher-order bits of PMCSRn (PMCSRn_[31:16]) select or deselect writing of values from the 16 lower-order bits (PMCSRn_[15:0]) to the corresponding PMCn.PMCn_m bits.

Even when pins being used by multiple programs belong to the same port group, the PMCSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.

Bits 31 to 16 are always read as 0000_H. Reading bits 15 to 0 returns the value of the PMCn register.

Address: <PORT_base> + 0024_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMCSRn_31	PMCSRn_30	PMCSRn_29	PMCSRn_28	PMCSRn_27	PMCSRn_26	PMCSRn_25	PMCSRn_24	PMCSRn_23	PMCSRn_22	PMCSRn_21	PMCSRn_20	PMCSRn_19	PMCSRn_18	PMCSRn_17	PMCSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMCSRn_15	PMCSRn_14	PMCSRn_13	PMCSRn_12	PMCSRn_11	PMCSRn_10	PMCSRn_9	PMCSRn_8	PMCSRn_7	PMCSRn_6	PMCSRn_5	PMCSRn_4	PMCSRn_3	PMCSRn_2	PMCSRn_1	PMCSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.8 PMCSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PMCSRn_[31:16]	Enable bits that specify whether the value of the corresponding lower-order bit of PMCSRn_m is to be written to PMCn_m: 0: PMCn_m does not depend on PMCSRn_m. 1: The value of PMCn_m is the same as that of PMCSRn_m. Example: If PMCSRn.PMCSRn_31 = 1, the value of bit PMCSRn.PMCSRn_15 is written to bit PMCn.PMCn_15.
15 to 0	PMCSRn_[15:0]	Data bits that specify the PMCn_m value when the corresponding higher-order bit PMCSRn_(m+16) is 1: 0: PMCn_m = 0 1: PMCn_m = 1

(3) PIPCn — Port IP Control Register

This register specifies whether the I/O direction of pin Pn_m is controlled by the port mode register PMn.PMn_m or by an alternative function.

When the Pn_m pin is operated in alternative mode (PMCn.PMCn_m = 1) and the alternative function directly controls the I/O direction of Pn_m, PIPCn.PIPCn_m must be set to 1 as well. This hands over I/O control to the alternative function and overrules the PMn.PMn_m setting.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 4008_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIPCn_15	PIPCn_14	PIPCn_13	PIPCn_12	PIPCn_11	PIPCn_10	PIPCn_9	PIPCn_8	PIPCn_7	PIPCn_6	PIPCn_5	PIPCn_4	PIPCn_3	PIPCn_2	PIPCn_1	PIPCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.9 PIPCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PIPCn_[15:0]	Specify the I/O mode: 0: I/O mode is selected by PMn.PMn_m (software I/O control). 1: I/O mode is selected by the peripheral function (direct I/O control).

(4) PMn — Port Mode Register

The PMn register specifies whether the individual pins of port group n are in input mode or in output mode.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 0010_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMn_15	PMn_14	PMn_13	PMn_12	PMn_11	PMn_10	PMn_9	PMn_8	PMn_7	PMn_6	PMn_5	PMn_4	PMn_3	PMn_2	PMn_1	PMn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.10 PMn Register Contents

Bit Position	Bit Name	Function
15 to 0	PMn_[15:0]	Specify input/output mode of the corresponding pin: 0: Output mode (output enabled) 1: Input mode (output disabled)

NOTES

1. To use a pin in input port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 1), the input buffer must be enabled (PIBCn.PIBCn_m = 1).
2. PMn_m specifies the I/O direction in port mode (PMCn.PMCn_m = 0) and alternative mode (PMCn.PMCn_m = 1) because PIPCn.PIPCn_m = 0 after reset.

(5) PMSRn — Port Mode Set/Reset Register

This register provides an alternative method to write data to a bit in the PMn register.

The 16 higher-order bits of PMSRn (PMSRn_[31:16]) select or deselect writing of values from the 16 lower-order bits (PMSRn_[15:0]) to the corresponding PMn.PMn_m bits.

Even when pins being used by multiple programs belong to the same port group, the PMSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.

Bits 31 to 16 are always read as 0000_H. Reading bits 15 to 0 returns the value of the PMn register.

Address: <PORT_base> + 0020_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMSRn_31	PMSRn_30	PMSRn_29	PMSRn_28	PMSRn_27	PMSRn_26	PMSRn_25	PMSRn_24	PMSRn_23	PMSRn_22	PMSRn_21	PMSRn_20	PMSRn_19	PMSRn_18	PMSRn_17	PMSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMSRn_15	PMSRn_14	PMSRn_13	PMSRn_12	PMSRn_11	PMSRn_10	PMSRn_9	PMSRn_8	PMSRn_7	PMSRn_6	PMSRn_5	PMSRn_4	PMSRn_3	PMSRn_2	PMSRn_1	PMSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.11 PMSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PMSRn_[31:16]	Enable bits that specify whether the value of the corresponding lower-order bit of PMSRn_m is written to PMn_m: 0: PMn_m does not depend on PMSRn_m. 1: The value of PMn_m is the same as that of PMSRn_m. Example: If PMSRn.PMSRn_31 = 1, the value of bit PMSRn.PMSRn_15 is written to bit PMn.PMn_15.
15 to 0	PMSRn_[15:0]	Data bits that specify the PMn_m value when the corresponding higher-order bit PMSRn_(m+16) is 1: 0: PMn_m = 0 1: PMn_m = 1

(6) PIBCn — Port Input Buffer Control Register

This register enables and disables the input buffer when a pin is used in input port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 1). However, when the pin is used as an input pin in software I/O control alternative mode (PMCn.PMCn_m = 1 and PIPCn.PIPCn_m = 0) or in direct I/O control alternative mode (PMCn.PMCn_m = 1 and PIPCn.PIPCn_m = 1), set PIBCn.PIBCn_m to 0.

And when pins are in bidirectional mode (PBDCn.PBDCn_m = 1), alternative output level loopback function and pin output level read function can be selected by the setting of PIBCn.PIBCn_m.

Refer to **2.1.4.3(1), PBDCn — Port Bidirectional Control Register**.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 4000_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIBCn_15	PIBCn_14	PIBCn_13	PIBCn_12	PIBCn_11	PIBCn_10	PIBCn_9	PIBCn_8	PIBCn_7	PIBCn_6	PIBCn_5	PIBCn_4	PIBCn_3	PIBCn_2	PIBCn_1	PIBCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.12 PIBCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PIBCn_[15:0]	Enable and disable the input buffer in input port mode. 0: Input buffer is disabled. 1: Input buffer is enabled.

NOTE

When the input buffer is disabled, through current does not flow even when the pin level is Hi-Z. Thus the pin does not need to be fixed to a high or low level externally.

(7) PFCn — Port Function Control Register

This register, together with the PFCEn and PFCAEn registers, specifies an alternative function of the pins.

Some alternative functions directly control input/output of pin Pn_m. For such alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 0018_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCn_15	PFCn_14	PFCn_13	PFCn_12	PFCn_11	PFCn_10	PFCn_9	PFCn_8	PFCn_7	PFCn_6	PFCn_5	PFCn_4	PFCn_3	PFCn_2	PFCn_1	PFCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.13 PFCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

(8) PFCEn — Port Function Control Expansion Register

This register, together with the PFCn and PFCAEn registers, specifies an alternative function of the pins.

Some alternative functions directly control input/output of pin Pn_m. For such alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + $001C_H$ + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCEn_15	PFCEn_14	PFCEn_13	PFCEn_12	PFCEn_11	PFCEn_10	PFCEn_9	PFCEn_8	PFCEn_7	PFCEn_6	PFCEn_5	PFCEn_4	PFCEn_3	PFCEn_2	PFCEn_1	PFCEn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.14 PFCEn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCEn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

(9) PFCAEn — Port Function Control Additional Expansion Register

This register, together with the PFCn and PFCEn registers, specifies an alternative function of the pins.

Some alternative functions directly control input/output of pin Pn_m. For such alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 0028_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCAEn_15	PFCAEn_14	PFCAEn_13	PFCAEn_12	PFCAEn_11	PFCAEn_10	PFCAEn_9	PFCAEn_8	PFCAEn_7	PFCAEn_6	PFCAEn_5	PFCAEn_4	PFCAEn_3	PFCAEn_2	PFCAEn_1	PFCAEn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.15 PFCAEn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCAEn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

2.1.4.3 Pin Data Input/Output

(1) PBDCn — Port Bidirectional Control Register

When a pin is used in output mode, this register enables the input buffer and sets the port to bidirectional mode. The Pn_m pin level is read via PPRn.PPRn_m in bidirectional mode.

- Alternative output level loopback function
  When the Pn_m pin is used as the alternative output function, the actual pin output level based on the alternative output function can be looped back to the alternative input side by setting PBDCn.PBDCn_m = 1 and PIBCn.PIBCn_m = 0. For example, the pin output level based on the first alternative function can be looped back to the same alternative input side. Also the pin output level can be read via PPRn.PPRn_m.
- Pin output level read function
  When the Pn_m pin is used as the general-purpose output port function or the alternative output function, the actual pin output level can be read via PPRn.PPRn_m by setting PBDCn.PBDCn_m = 1 and PIBCn.PIBCn_m = 1. Under this setting, the pin output level will never be looped back to the alternative input side even in alternative output mode.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 4004_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PBDCn_15	PBDCn_14	PBDCn_13	PBDCn_12	PBDCn_11	PBDCn_10	PBDCn_9	PBDCn_8	PBDCn_7	PBDCn_6	PBDCn_5	PBDCn_4	PBDCn_3	PBDCn_2	PBDCn_1	PBDCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.16 PBDCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PBDCn_[15:0]	Enable or disable bidirectional mode of the corresponding pin: 0: Bidirectional mode is disabled. 1: Bidirectional mode is enabled.

(2) PPRn — Port Pin Read Register

This register reflects an actual Pn_m pin level, a Pn.Pn_m bit value, or an output level of the alternative function. The value to be read depends on various control settings as described in **Table 2.5, PPRn_m Read Values**.

Access: This register is only readable in 16-bit units.

Address: <PORT_base> + $000C_H$ + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PPRn_15	PPRn_14	PPRn_13	PPRn_12	PPRn_11	PPRn_10	PPRn_9	PPRn_8	PPRn_7	PPRn_6	PPRn_5	PPRn_4	PPRn_3	PPRn_2	PPRn_1	PPRn_0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 2.17 PPRn Register Contents

Bit Position	Bit Name	Function
15 to 0	PPRn_[15:0]	Indicate a Pn_m pin level, a Pn.Pn_m value, or alternative function output level.

(3) Pn — Port Register

This register sets and holds the Pn.Pn_m data to be output via the related Pn_m port in output port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 0).

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 0000_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Pn_15	Pn_14	Pn_13	Pn_12	Pn_11	Pn_10	Pn_9	Pn_8	Pn_7	Pn_6	Pn_5	Pn_4	Pn_3	Pn_2	Pn_1	Pn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.18 Pn Register Contents

Bit Position	Bit Name	Function
15 to 0	Pn_[15:0]	Set the output level of pin m (m = 0 to 15): 0: Low level output 1: High level output

NOTE

The bits of this register can be manipulated by various means; refer to the subsection, Write to the Pn Register in **Section 2.1.2.3, Pin Data Input/Output**.

(4) PNOTn — Port NOT Register

This register allows bit Pn_m of the port register Pn to be inverted without directly writing to Pn.

Access: This register is writable in 16-bit units. The read value is always 0000_H.

Address: <PORT_base> + 0008_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PNOTn_15	PNOTn_14	PNOTn_13	PNOTn_12	PNOTn_11	PNOTn_10	PNOTn_9	PNOTn_8	PNOTn_7	PNOTn_6	PNOTn_5	PNOTn_4	PNOTn_3	PNOTn_2	PNOTn_1	PNOTn_0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 2.19 PNOTn Register Contents

Bit Position	Bit Name	Function
15 to 0	PNOTn_[15:0]	Specify if Pn.Pn_m is inverted or not: 0: Pn.Pn_m is not inverted (Pn_m → Pn_m). 1: Pn.Pn_m is inverted ($\overline{\text{Pn\_m}}$ → Pn_m)

(5) PSRn — Port Set/Reset Register

This register provides an alternative method to write data to a bit in the Pn register.

The 16 higher-order bits of PSRn (PSRn_[31:16]) select or deselect writing of values from the 16 lower-order bits (PSRn_[15:0]) to the corresponding Pn.Pn_m bits.

Even when pins being used by multiple programs belong to the same port group, the PSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.

Bits 31 to 16 are always read as 0000_H. Reading bits 15 to 0 returns the value of the Pn register.

Address: <PORT_base> + 0004_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PSRn_31	PSRn_30	PSRn_29	PSRn_28	PSRn_27	PSRn_26	PSRn_25	PSRn_24	PSRn_23	PSRn_22	PSRn_21	PSRn_20	PSRn_19	PSRn_18	PSRn_17	PSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSRn_15	PSRn_14	PSRn_13	PSRn_12	PSRn_11	PSRn_10	PSRn_9	PSRn_8	PSRn_7	PSRn_6	PSRn_5	PSRn_4	PSRn_3	PSRn_2	PSRn_1	PSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.20 PSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PSRn_[31:16]	Enable bits that specify whether the value of the corresponding lower-order bit of PSRn_m is written to Pn_m: 0: Pn_m does not depend on PSRn_m. 1: The value of Pn_m is the same as that of PSRn_m. Example: When PSRn.PSRn_31 = 1, the value of bit PSRn.PSRn_15 is written to bit Pn.Pn_15 and output.
15 to 0	PSRn_[15:0]	Specify the Pn_m value when the corresponding higher-order bit PSRn_(m+16) is 1: 0: Pn_m = 0 1: Pn_m = 1

2.1.4.4 Configuration of Electrical Characteristics Registers

(1) PUn — Pull-Up Option Register

This register specifies whether an on-chip pull-up resistor is connected to an input pin.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + $400C_H$ + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PUn_15	PUn_14	PUn_13	PUn_12	PUn_11	PUn_10	PUn_9	PUn_8	PUn_7	PUn_6	PUn_5	PUn_4	PUn_3	PUn_2	PUn_1	PUn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.21 PUn Register Contents

Bit Position	Bit Name	Function
15 to 0	PUn_[15:0]	Specify whether an on-chip pull-up resistor is connected to the corresponding pin: 0: No on-chip pull-up resistor is connected. 1: On-chip pull-up resistor is connected.

NOTES

1. Do not set PUn.PUn_m = 1 and PDn.PDn_m = 1 to a single pin.
2. The on-chip pull-up resistor has no effect when the pin is operated in output mode.

(2) PDn — Pull-Down Option Register

This register specifies whether an on-chip pull-down resistor is connected to an input pin.

Access: This register is readable/writable in 16-bit units.

Address: <PORT_base> + 4010_H + n × 40_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDn_15	PDn_14	PDn_13	PDn_12	PDn_11	PDn_10	PDn_9	PDn_8	PDn_7	PDn_6	PDn_5	PDn_4	PDn_3	PDn_2	PDn_1	PDn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.22 PDn Register Contents

Bit Position	Bit Name	Function
15 to 0	PDn_[15:0]	Specify whether an on-chip pull-down resistor is connected to the corresponding pin: 0: No on-chip pull-down resistor is connected. 1: On-chip pull-down resistor is connected.

NOTES

1. Do not set PUn.PUn_m = 1 and PDn.PDn_m = 1 to a single pin.
2. The on-chip pull-down resistor has no effect when the pin is operated in output mode.

2.1.4.5 Pin-Unit Register

(1) PCRn_m — Port Control Register

Each register of a port group can be accessed via this register and a PCRn_m register can set all functions of a single pin. For example, setting bit 6 of the PCRn_m register to 1 sets bit m of the PMCn register to 1 also.

Access: This register is readable/writable in 32-bit units.

Address: <PORT_base> + 2000_H + n × 40_H + m × 4_H

Value after reset: See **Section 2.2.1.1 List of the C1M-A2 Port Registers (BGA)** and **Section 2.2.2.1 List of the C1M-A1 Port Registers (QFP)**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PUCC	PDSC	—	—	—	—	PU	PD	PBDC	PIBC
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	P	—	—	—	PPR	—	PMC	PIPC	PM	—	PFCAE	PFCE	PFC
R/W	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 2.23 PCRn_m Register Contents

Bit Position	Bit Name	Function
31 to 26	Reserved	When read, the value after reset is read. When writing, write the value after reset.
25	PUCC	PUCC PDSC Selects the output drive strength. 0 0: Low 0 1: High 1 0: Mid Settings other than the above are prohibited.
24	PDSC	
23 to 20	Reserved	When read, the value after reset is read. When writing, write the value after reset.
19	PU	Same function as bit m of the PUn register
18	PD	Same function as bit m of the PDn register
17	PBDC	Same function as bit m of the PBDCn register
16	PIBC	Same function as bit m of the PIBCn register
15 to 13	Reserved	When read, the value after reset is read. When writing, write the value after reset.
12	P	Same function as bit m of the Pn register
11 to 9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	PPR	Same function as bit m of the PPRn register
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6	PMC	Same function as bit m of the PMCn register
5	PIPC	Same function as bit m of the PIPCn register
4	PM	Same function as bit m of the PMn register
3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2	PFCAE	Same function as bit m of the PFCAEn register
1	PFCE	Same function as bit m of the PFCEn register
0	PFC	Same function as bit m of the PFCn register

2.1.4.6 Example Port Settings

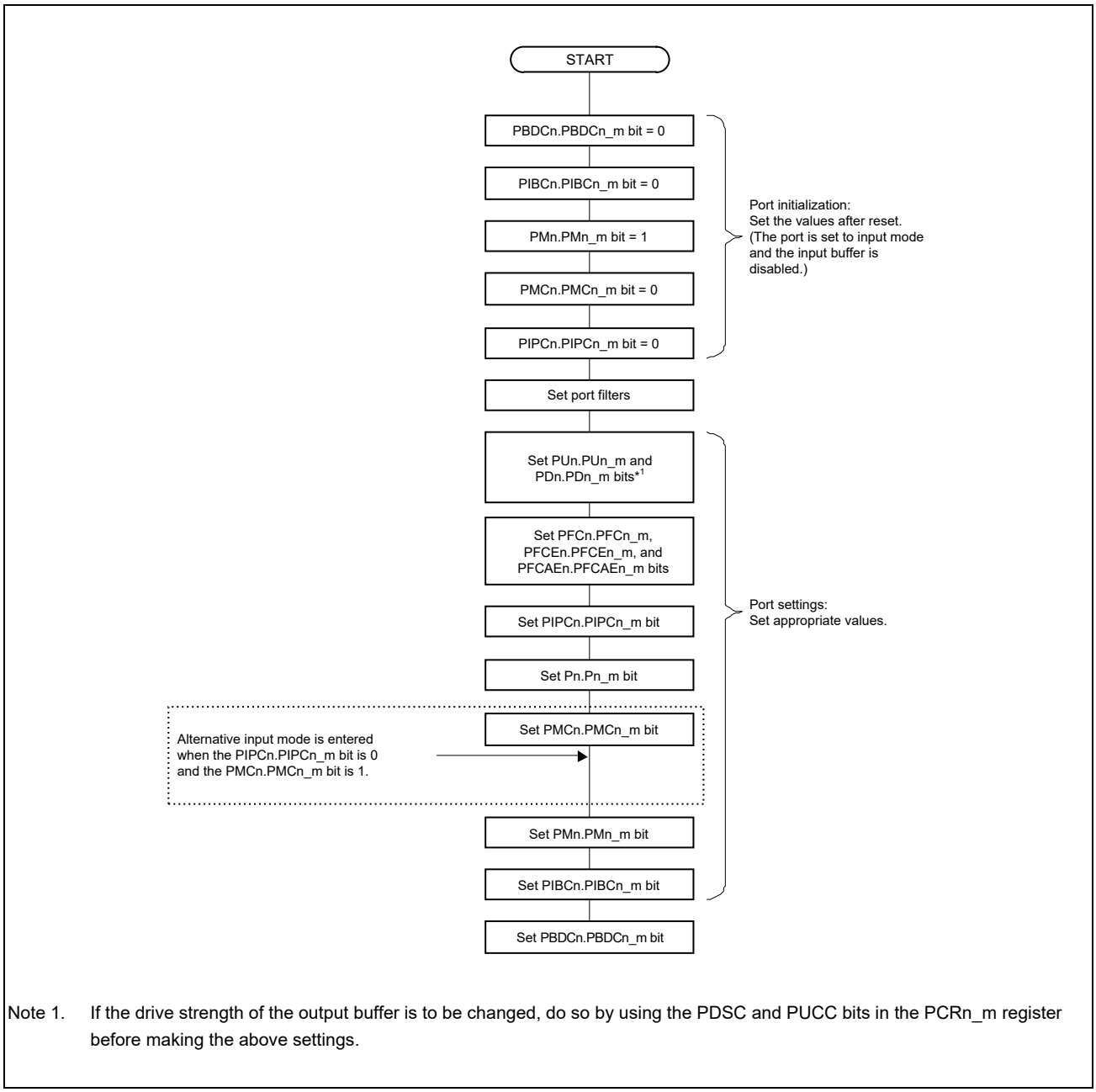
Examples of the port settings are shown in the flowchart below. For port filter setting of each flow chart, see **Section 2.3.2.4, Setting Procedures of Peripheral Function DNF**.

CAUTION

If a port is used in software I/O control alternative mode, it may be temporarily switched to alternative input mode in the following example. This may occur between when PMCn_m is set to 1 and when PMn_m is set to 0.

(1) Batch Setting

An example of specifying batch port settings is shown in the flowchart below.



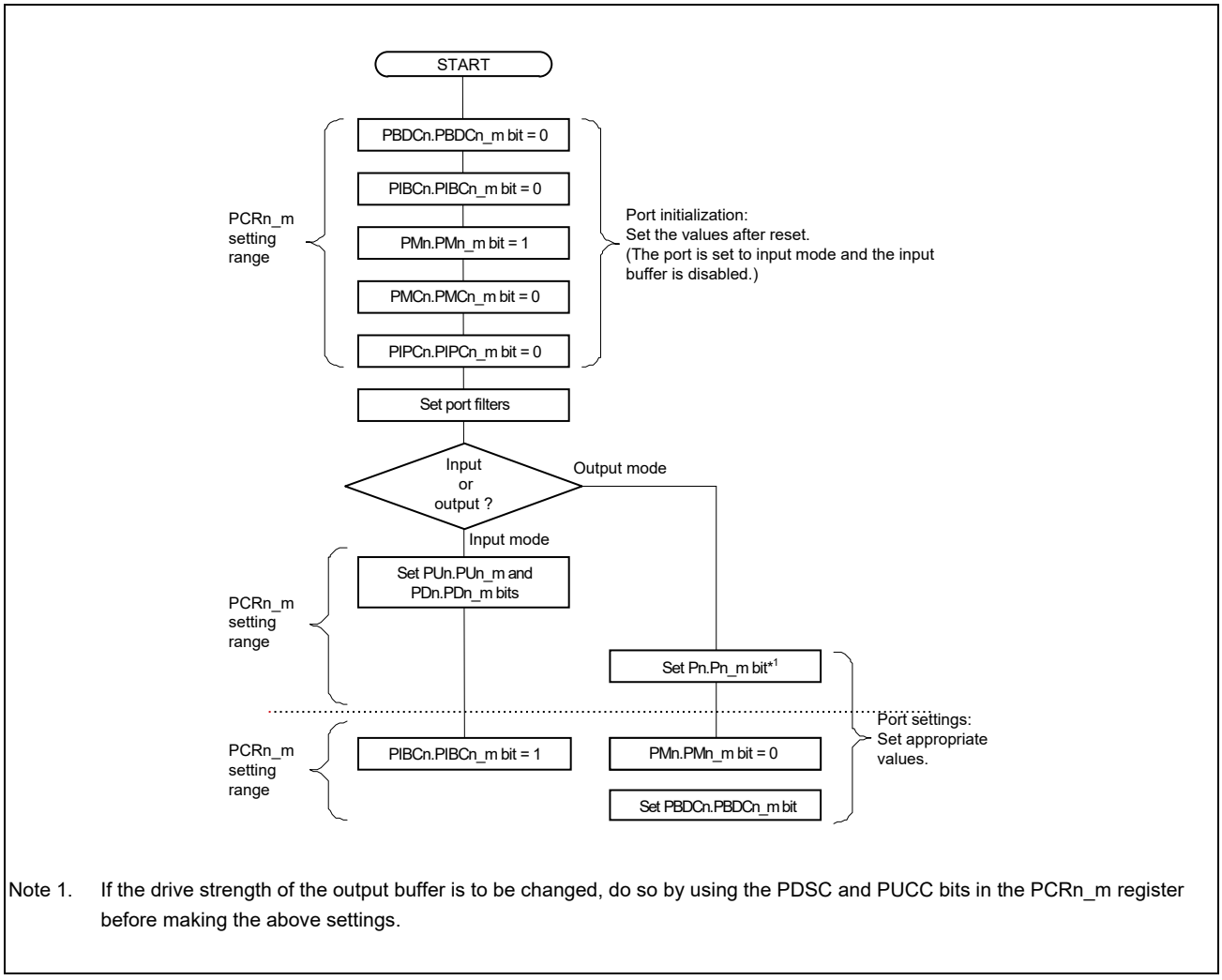
Note 1. If the drive strength of the output buffer is to be changed, do so by using the PDSC and PUCC bits in the PCRn_m register before making the above settings.

Figure 2.3 Example of Port Settings (When Specified in Batch)

(2) Individual Settings

An example of specifying individual port settings is shown in the flowchart below.

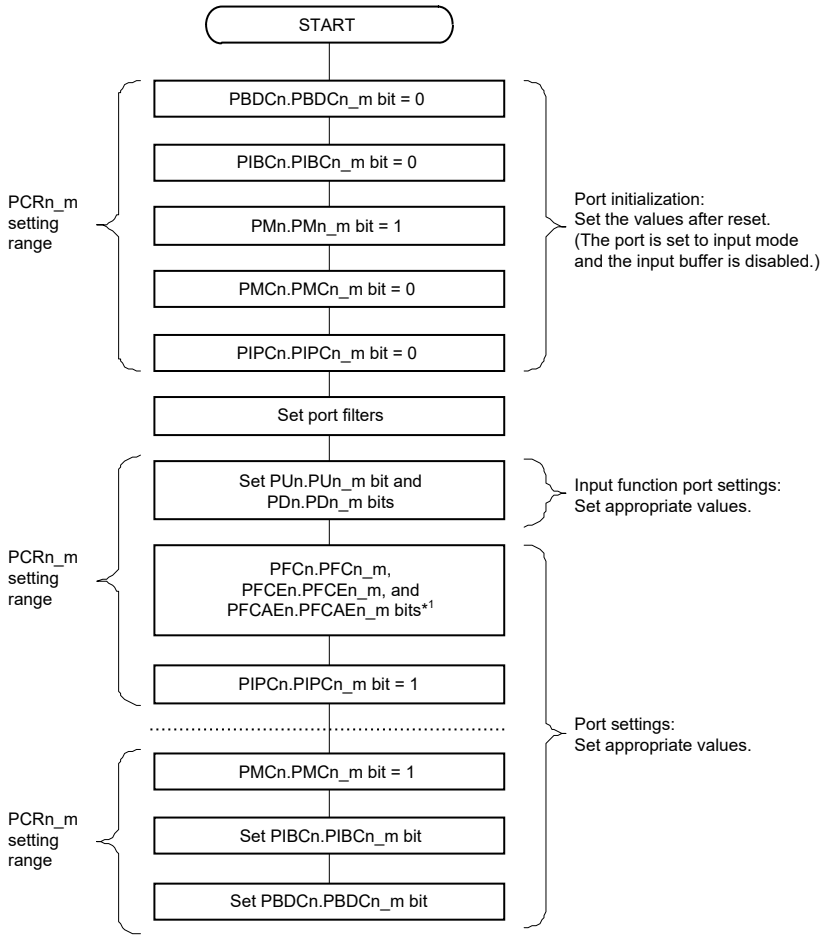
Furthermore, setting of multiple bits within the range for setting of the PCRn_m register shown in the figure below is possible.



Note 1. If the drive strength of the output buffer is to be changed, do so by using the PDSC and PUCC bits in the PCRn_m register before making the above settings.

Figure 2.4 Example of Port Settings (in Port Mode)

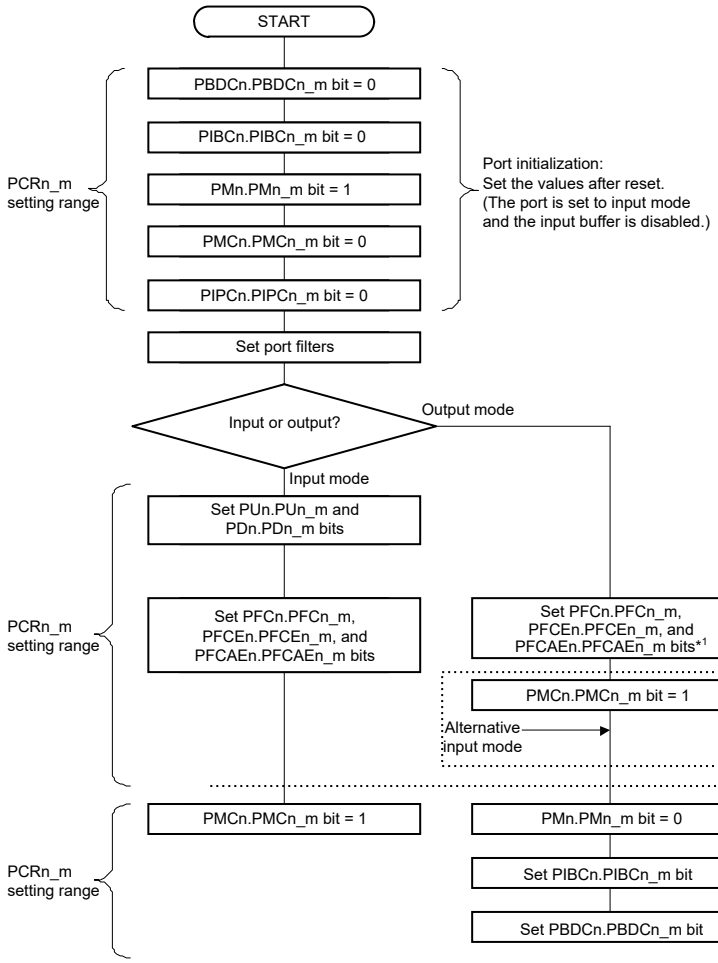
(a) With IP Control (PIPC = 1)



Note 1. If the drive strength of the output buffer is to be changed, do so by using the PDSC and PUCC bits in the PCRn_m register before making the above settings.

Figure 2.5 Example of Port Settings (in Alternative Mode) (1/2)

(b) Without IP Control (PIPC = 0)



Note 1. If the drive strength of the output buffer is to be changed, do so by using the PDSC and PUCC bits in the PCRn_m register

Figure 2.6 Example of Port Settings (in Alternative Mode) (2/2)

2.1.5 Functional Selection

2.1.5.1 Register Configuration in Use of the Alternative Function

When the pin alternative function is used, set PMCn_m = 1 and select the alternative numbers of PFCn_m, PFCEn_m, and PFCAEn_m. In several peripheral functions, a single alternative I/O function is allocated to multiple pins. However, such an alternative function should not be enabled in multiple pins at the same time.

2.1.5.2 Alternative Function to be Used in Direct I/O Control Alternative Mode

When the alternative functions described in **Table 2.24** are used, switch to direct I/O control alternative mode. When setting PIPCn_m = 1, the PMn_m value which has been set is ignored because the peripheral function enables or disables inputs and outputs of the buffer.

If you are using an alternative function not listed in **Table 2.24**, set PIPCn_m = 0.

Table 2.24 List of the Pins which Require PIPC Register Setting

Category	Pin Name	I/O	Function
SCIn (n = 0 to 2)	SCInRXD	I	Receive data input
	SCInTXD	O	Transmit data output
	SCInSCK	I/O	Serial clock input/output
CSIHn (n = 0 to 2)	CSIHnSO	O	Transmit data output
	CSIHnSC	I/O	Serial clock input/output
	CSIHnRYI/CSIHnRYO	I/O	Handshake signal input/output
TSG3n*1	TSG3nO1	O	Three-phase PWM output (Hi-Z control)
	TSG3nO2	O	
	TSG3nO3	O	
	TSG3nO4	O	
	TSG3nO5	O	
	TSG3nO6	O	
TAPAn*2	TAPAn UN	O	Three-phase PWM output (Hi-Z control)
	TAPAn UP	O	
	TAPAn VN	O	
	TAPAn VP	O	
	TAPAn WN	O	
	TAPAn WP	O	
RSENT (n = 0 to 3)	RSENTnRX	I	RSENTn receive data input
	RSENTnSPCO	O	RSENTnSPC extended output

Note 1. C1M-A2: n = 0 to 2, C1M-A1: n = 0 and 1

Note 2. C1M-A2: n = 0 to 5, C1M-A1: n = 0, 1, 3, and 4

2.1.5.3 Setting of the $\overline{\text{ERROROUT\_C}}$ Pin

When an error output function of the $\overline{\text{ERROROUT\_C}}$ pin is used, enable the alternative output level loopback function for fault diagnosis.

2.1.5.4 Selecting Debugging Function of the JTAG Port

For the connection of multiple tools to the JTAG port, the interface is multiplexed on multiple sets of pins. The interface is selected by the combination of the settings of mode pins and an option byte. When user boot mode is selected as the operating mode, the interface is selected by the setting of option byte OPBT2.

When any other operating mode is selected, the setting of OPBT2 has no effect and the I/F that corresponds to the operating mode is selected. For details, refer to **Section 5.2, Operating Mode.**

2.2 Organization of Port Groups

2.2.1 C1M-A2 Port Function (BGA)

2.2.1.1 List of the C1M-A2 Port Registers (BGA)

Table 2.25 to **Table 2.32** show detailed bitmaps of control registers in each port group. In the bitmap field, "√" means an effective bit and "—" means a reserved one. Reserved bits are always read as value after resets. The write value also should be a value after reset.

Table 2.25 List of Registers in Port Group 0 of C1M-A2 (BGA)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	0	P0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR0	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR0	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM0	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT0	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR0	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR0	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC0	R/W	0000_H	16	√	√	√	√	√	√	—	—	—	—	—	—	—	—	—	—	
PORT		PU0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
PORT	0	PCR0_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.26 List of Registers in Port Group 1 of C1M-A2 (BGA)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	1	P1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR1	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM1	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT1	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR1	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC1	R/W	0000_H	16	√	√	√	√	√	√	—	—	√	√	√	√	√	√	—	—	
PORT		PU1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	1	PCR1_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR1_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR1_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR1_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR1_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.27 List of Registers in Port Group 2 of C1M-A2 (BGA)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	2	P2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR2	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PM2	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCAE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PNOT2	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR2	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PBDC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	—	—	
PORT		PU2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	2	PCR2_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR2_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR2_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.28 List of Registers in Port Group 3 of C1M-A2 (BGA)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	3	P3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR3	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PM3	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCAE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PNOT3	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR3	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PBDC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	—	√	
PORT		PU3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	3	PCR3_0	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_1	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR3_2	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_3	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_4	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_5	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_6	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_7	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.29 List of Registers in Port Group 4 of C1M-A2 (BGA)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	4	P4	R/W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR4	R/W	0000 0000$_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR4	R	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM4	R/W	FFFF$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC4	R/W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC4	R/W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	—	—	—	
PORT		PFCE4	R/W	0000$_H$	16	—	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE4	R/W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT4	W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR4	R/W	0000 FFFF$_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR4	R/W	0000 0000$_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC4	R/W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC4	R/W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC4	R/W	0000$_H$	16	—	—	√	—	—	—	√	√	—	—	—	—	—	√	√	—	
PORT		PU4	R/W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD4	R/W	0000$_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	4	PCR4_0	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	—
PORT		PCR4_1	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—
PORT		PCR4_2	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—
PORT		PCR4_3	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_4	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_5	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_6	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_7	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_8	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_9	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_10	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_11	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_12	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_13	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_14	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√
PORT		PCR4_15	R/W	0000 0010$_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√

Table 2.30 List of Registers in Port Group 5 of C1M-A2 (BGA)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	5	P5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PSR5	R/W	0000 0000H	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR5	R	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PM5	R/W	FFFFH	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PMC5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFC5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT5	W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR5	R/W	0000 FFFFH	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR5	R/W	0000 0000H	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	—	√	√	√	√	√	√	
PORT		PU5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PD5	R/W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	5	PCR5_0	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_1	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_2	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_3	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_4	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_5	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_6	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR5_7	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_8	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_9	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.31 List of Registers in Port Group 6 of C1M-A2 (BGA)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	6	P6	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR6	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR6	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM6	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC6	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC6	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	—	√	
PORT		PFCAE6	R/W	0000_H	16	√	√	√	√	√	√	—	—	√	√	—	√	—	√	—	√	
PORT		PNOT6	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR6	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR6	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC6	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC6	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC6	R/W	0000_H	16	√	√	√	√	√	√	—	—	—	—	—	—	—	—	—	—	
PORT		PU6	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD6	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
PORT	6	PCR6_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR6_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	—	√
PORT		PCR6_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR6_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	√	√
PORT		PCR6_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR6_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	√	√
PORT		PCR6_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR6_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR6_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	—	√
PORT		PCR6_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	—	√
PORT		PCR6_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√
PORT		PCR6_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√
PORT		PCR6_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√
PORT		PCR6_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√
PORT		PCR6_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√
PORT		PCR6_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√

Table 2.32 List of Registers in Port Group 7 of C1M-A2 (BGA)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	7	P7	R/W	0000H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR7	R/W	0000 0000H	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR7	R	0000H	16	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	
PORT		PM7	R/W	FFFFH	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC7	R/W	0000H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
PORT		PFC7	R/W	0000H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
PORT		PFCE7	R/W	0000H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
PORT		PFCAE7	R/W	0000H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
PORT		PNOT7	W	0000H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR7	R/W	0000 FFFFH	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR7	R/W	0000 0000H	32	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC7	R/W	0000H	16	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	
PORT		PBDC7	R/W	0000H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC7	R/W	0000H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PU7	R/W	0000H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD7	R/W	0000H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	7	PCR7_0	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR7_1	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR7_2	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR7_3	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR7_4	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR7_5	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR7_6	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	—	—	√	—	—	—
PORT		PCR7_7	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	—	—	√	—	—	—
PORT		PCR7_8	R/W	0000 0000H	32	—	—	—	—	—	√	—	√	—	—	—	—	—	—

2.2.1.2 List of Pin Alternative Functions of C1M-A2 (BGA)

Table 2.33 to **Table 2.40** show the list of alternative functions of each port pin. In the tables, "—" means a reserved bit which cannot be selected.

Table 2.33 List of Pin Alternative Functions in C1M-A2 (BGA) Port Group 0

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P0_0	IO	TAUD0I0	TAUD0O0	TAUJ0I0	TAUJ0O0	—	—	—	—	—	—	—	—	—	—	—	—
P0_1	IO	TAUD0I1	TAUD0O1	TAUJ0I1	TAUJ0O1	—	—	—	—	—	—	TAPA5ESO	—	—	—	—	—
P0_2	IO	TAUD0I2	TAUD0O2	TAUJ0I2	TAUJ0O2	—	—	—	—	—	—	TAPA3ESO	—	—	—	—	—
P0_3	IO	TAUD0I3	TAUD0O3	TAUJ0I3	TAUJ0O3	—	—	—	—	CAN2RX	—	—	—	INTP0	—	—	—
P0_4	IO	TAUD0I4	TAUD0O4	—	—	—	—	TAUD3I1	TAUD3O1	—	CAN2TX	—	—	INTP1	—	—	—
P0_5	IO	TAUD0I5	TAUD0O5	TAUJ0I0	TAUJ0O0	—	—	—	—	—	—	TAPA4ESO	—	INTP2	—	—	—
P0_6	IO	TAUD0I6	TAUD0O6	TAUJ0I1	TAUJ0O1	ENCA0E0	RDC3A0_OUT_U	—	—	—	—	—	—	INTP3	—	—	—
P0_7	IO	TAUD0I7	TAUD0O7	TAUJ0I2	TAUJ0O2	ENCA0E1	RDC3A0_OUT_V	—	—	—	—	—	—	INTP4	—	—	—
P0_8	IO	TAUD0I8	TAUD0O8	TAUJ0I3	TAUJ0O3	ENCA0EC	RDC3A0_OUT_W	—	—	—	—	—	—	INTP5	—	—	—
P0_9	IO	TAUD0I9	TAUD0O9	—	—	—	—	TAUD3I3	TAUD3O3	—	—	TAPA5ESO	—	INTP6	—	—	—
P0_10	IO	TAUD0I10	TAUD0O10	—	TAPA0UP	—	—	TAUD3I5	TAUD3O5	—	—	—	—	INTP7	—	—	—
P0_11	IO	TAUD0I11	TAUD0O11	—	TAPA0UN	—	—	TAUD3I7	TAUD3O7	—	—	TSGTRG	—	—	—	—	—
P0_12	IO	TAUD0I12	TAUD0O12	—	TAPA0VP	—	—	TAUD3I9	TAUD3O9	RSENT0RX	RSENT0SPCO	—	—	—	—	—	—
P0_13	IO	TAUD0I13	TAUD0O13	—	TAPA0VN	—	—	TAUD3I11	TAUD3O11	—	RSENT0SPCO	—	—	—	—	—	—
P0_14	IO	TAUD0I14	TAUD0O14	—	TAPA0WP	—	—	TAUD3I13	TAUD3O13	RSENT1RX	RSENT1SPCO	—	—	—	—	—	—
P0_15	IO	TAUD0I15	TAUD0O15	—	TAPA0WN	—	—	TAUD3I15	TAUD3O15	—	RSENT1SPCO	—	—	—	—	—	—

Table 2.34 List of Pin Alternative Functions in C1M-A2 (BGA) Port Group 1

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P1_0	IO	TAUD1I0	TAUD1O0	—	—	ENCA0TIN0	—	—	TAUD1O1	—	TSG30O0	—	—	—	—	—	—
P1_1	IO	TAUD1I1	TAUD1O1	—	—	ENCA0TIN1	—	—	—	—	TSG30O7	—	—	—	—	—	—
P1_2	IO	TAUD1I2	TAUD1O2	—	—	—	—	—	TAUD1O3	—	TSG30O1	—	—	—	—	—	—
P1_3	IO	TAUD1I3	TAUD1O3	—	—	—	—	—	—	—	TSG30O3	—	—	—	—	—	—
P1_4	IO	TAUD1I4	TAUD1O4	—	—	—	—	—	TAUD1O5	—	TSG30O5	—	—	—	—	—	—
P1_5	IO	TAUD1I5	TAUD1O5	—	—	—	—	—	—	—	TSG30O2	—	—	—	—	—	—
P1_6	IO	TAUD1I6	TAUD1O6	—	—	—	—	—	TAUD1O7	—	TSG30O4	—	—	—	—	—	—
P1_7	IO	TAUD1I7	TAUD1O7	—	—	—	—	—	—	—	TSG30O6	—	—	—	—	—	—
P1_8	IO	TAUD1I8	TAUD1O8	—	—	—	—	—	TAUD1O9	—	TSG32O0	TAPA2ESO	—	—	—	—	—
P1_9	IO	TAUD1I9	TAUD1O9	—	—	—	—	—	—	—	TSG32O7	TAPA4ESO	—	—	—	—	—
P1_10	IO	TAUD1I10	TAUD1O10	—	TAPA1UP	—	—	—	TAUD1O11	—	TSG32O1	—	—	—	—	—	—
P1_11	IO	TAUD1I11	TAUD1O11	—	TAPA1UN	—	—	—	—	—	TSG32O3	—	—	—	—	—	—
P1_12	IO	TAUD1I12	TAUD1O12	—	TAPA1VP	—	—	—	TAUD1O13	—	TSG32O5	—	—	—	—	—	—
P1_13	IO	TAUD1I13	TAUD1O13	—	TAPA1VN	—	—	—	—	—	TSG32O2	—	—	—	—	—	—
P1_14	IO	TAUD1I14	TAUD1O14	—	TAPA1WP	—	—	—	TAUD1O15	—	TSG32O4	—	—	—	—	—	—
P1_15	IO	TAUD1I15	TAUD1O15	—	TAPA1WN	—	—	—	—	—	TSG32O6	—	—	—	—	—	—

Table 2.35 List of Pin Alternative Functions in C1M-A2 (BGA) Port Group 2

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P2_0	IO	TAUD2I0	TAUD2O0	—	—	—	—	—	—	—	TSG31O0	—	—	INTP0	—	—	—
P2_1	IO	TAUD2I1	TAUD2O1	—	—	—	—	—	—	—	TSG31O7	—	—	INTP1	—	—	—
P2_2	IO	TAUD2I2	TAUD2O2	—	—	—	—	—	—	—	TSG31O1	—	—	INTP2	—	—	—
P2_3	IO	TAUD2I3	TAUD2O3	—	—	—	—	—	—	—	TSG31O3	—	—	INTP3	—	—	—
P2_4	IO	TAUD2I4	TAUD2O4	—	—	—	—	—	—	—	TSG31O5	—	—	INTP4	—	—	—
P2_5	IO	TAUD2I5	TAUD2O5	—	—	—	—	—	—	—	TSG31O2	—	—	INTP5	—	—	—
P2_6	IO	TAUD2I6	TAUD2O6	—	—	—	—	—	—	—	TSG31O4	—	—	INTP6	—	—	—
P2_7	IO	TAUD2I7	TAUD2O7	—	—	—	—	—	—	—	TSG31O6	—	—	INTP7	—	—	—

Table 2.36 List of Pin Alternative Functions in C1M-A2 (BGA) Port Group 3

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P3_0	IO	TAUD2I8	TAUD2O8	—	—	—	—	—	—	RSENT2RX	RSENT2SPCO	TAPA1ESO	—	—	—	—	—
P3_1	IO	TAUD2I9	TAUD2O9	—	—	—	—	ADCC0TRG	—	—	RSENT2SPCO	TAPA2ESO	—	—	—	—	—
P3_2	IO	TAUD2I10	TAUD2O10	—	TAPA2UP	—	—	ADCC1TRG	—	—	—	—	—	—	—	—	—
P3_3	IO	TAUD2I11	TAUD2O11	—	TAPA2UN	ENCA1E0	RDC3A1_OUT_U	—	—	—	—	—	—	—	—	—	—
P3_4	IO	TAUD2I12	TAUD2O12	—	TAPA2VP	ENCA1E1	RDC3A1_OUT_V	ADCC0TRG	—	—	—	TAPA3ESO	—	—	—	—	—
P3_5	IO	TAUD2I13	TAUD2O13	—	TAPA2VN	ENCA1EC	RDC3A1_OUT_W	—	—	—	—	TAPA0ESO	—	—	—	—	—
P3_6	IO	TAUD2I14	TAUD2O14	—	TAPA2WP	—	—	—	—	RSENT3RX	RSENT3SPCO	—	—	—	—	—	—
P3_7	IO	TAUD2I15	TAUD2O15	—	TAPA2WN	—	—	ADCC2TRG	—	—	RSENT3SPCO	—	—	—	—	—	—

Table 2.37 List of Pin Alternative Functions in C1M-A2 (BGA) Port Group 4

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P4_0	IO	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SI	—	—	—
P4_1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SO	—	—
P4_2	IO	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SC	CSIH1SC	—	—
P4_3	IO	—	—	—	—	—	—	—	—	CAN0RX	—	—	—	—	CSIH1CSS0	—	—
P4_4	IO	—	—	—	—	—	—	—	—	—	CAN0TX	—	—	—	CSIH1CSS1	—	—
P4_5	IO	—	—	—	—	—	—	—	—	CAN1RX	—	—	CSIH0CSS2	—	CSIH1CSS2	—	—
P4_6	IO	—	—	—	—	—	—	—	—	—	CAN1TX	—	CSIH0CSS3	—	CSIH1CSS3	—	—
P4_7	IO	—	—	—	—	—	—	—	—	—	—	CSIH0SI	—	$\overline{\text{CSIH1SSI}}$	—	—	—
P4_8	IO	—	—	—	—	—	—	—	—	—	—	—	CSIH0SO	CSIH1RYI	CSIH1RYO	—	—
P4_9	IO	—	—	—	—	—	—	—	—	—	—	CSIH0SC	CSIH0SC	—	—	—	—
P4_10	IO	—	—	—	—	—	TPBA0O	—	—	CAN2RX	—	—	CSIH0CSS0	—	—	—	—
P4_11	IO	—	—	—	—	—	—	—	—	—	CAN2TX	—	CSIH0CSS1	—	—	—	—
P4_12	IO	—	—	—	—	—	—	RLIN30RX	—	—	—	$\overline{\text{CSIH0SSI}}$	—	—	—	—	—
P4_13	IO	—	—	—	—	—	—	—	RLIN30TX	—	—	CSIH0RYI	CSIH0RYO	—	—	—	—
P4_14	IO	—	—	—	—	—	—	—	—	CAN3RX	—	—	—	—	—	—	—
P4_15	IO	—	—	—	—	—	—	—	—	—	CAN3TX	—	$\overline{\text{ERROROUT\_C}}$	—	—	—	—

Table 2.38 List of Pin Alternative Functions in C1M-A2 (BGA) Port Group 5

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P5_0	IO	—	—	—	—	—	—	RLIN32RX	—	—	—	SCI0RXD	—	—	CSIH2CSS3	—	—
P5_1	IO	—	—	—	—	—	—	—	RLIN32TX	—	—	—	SCI0TXD	$\overline{\text{CSIH2SSI}}$	—	—	—
P5_2	IO	—	—	—	—	—	—	RLIN31RX	—	—	—	SCI0SCK	SCI0SCK	CSIH2RYI	CSIH2RYO	—	—
P5_3	IO	—	—	—	—	—	—	—	RLIN31TX	—	—	SCI1SCK	SCI1SCK	—	—	—	—
P5_4	IO	—	—	—	—	—	—	RLIN30RX	—	—	—	SCI1RXD	—	—	—	—	—
P5_5	IO	—	—	—	—	—	—	—	RLIN30TX	—	—	—	SCI1TXD	—	$\overline{\text{ERROROUT\_C}}$	—	—
P5_6	IO	—	—	—	—	—	TPBA1O	—	—	—	—	TAPA0ESO	—	—	—	—	—
P5_7	IO	—	—	—	—	—	—	—	—	—	—	SCI2SCK	SCI2SCK	—	—	—	—
P5_8	IO	—	—	—	—	—	—	—	—	—	—	SCI2RXD	—	—	—	—	—
P5_9	IO	—	—	—	—	—	—	—	—	—	—	—	SCI2TXD	—	—	—	—

Table 2.39 List of Pin Alternative Functions in C1M-A2 (BGA) Port Group 6

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P6_0	IO	TAUD3I0	TAUD3O0	TAUJ1I2	TAUJ1O2	—	—	—	—	—	—	—	—	—	$\overline{\text{ERROROUT\_C}}$	—	—
P6_1	IO	TAUD3I1	TAUD3O1	TAUJ1I3	TAUJ1O3	—	—	—	—	—	—	—	—	—	—	—	—
P6_2	IO	TAUD3I2	TAUD3O2	—	—	ENCA1TIN0	RDC3A0_OUT_W	ADCC0TRG	—	—	—	—	—	—	—	—	—
P6_3	IO	TAUD3I3	TAUD3O3	—	—	—	RDC3A1_OUT_W	—	—	—	—	—	—	—	—	—	—
P6_4	IO	TAUD3I4	TAUD3O4	—	—	ENCA1TIN1	RDC3A0_OUT_V	—	—	—	—	—	—	—	—	—	—
P6_5	IO	TAUD3I5	TAUD3O5	—	—	—	RDC3A1_OUT_V	—	—	—	—	—	—	—	—	—	—
P6_6	IO	TAUD3I6	TAUD3O6	—	—	—	RDC3A0_OUT_U	—	—	—	—	TAPA0ESO	—	—	—	—	—
P6_7	IO	TAUD3I7	TAUD3O7	—	—	—	RDC3A1_OUT_U	—	—	—	—	TAPA1ESO	—	—	—	—	—
P6_8	IO	TAUD3I8	TAUD3O8	TAUJ1I0	TAUJ1O0	—	—	—	—	—	—	—	—	—	—	—	—
P6_9	IO	TAUD3I9	TAUD3O9	TAUJ1I1	TAUJ1O1	—	—	—	—	—	—	—	—	—	—	—	—
P6_10	IO	TAUD3I10	TAUD3O10	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_11	IO	TAUD3I11	TAUD3O11	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_12	IO	TAUD3I12	TAUD3O12	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_13	IO	TAUD3I13	TAUD3O13	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_14	IO	TAUD3I14	TAUD3O14	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_15	IO	TAUD3I15	TAUD3O15	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Table 2.40 List of Pin Alternative Functions in C1M-A2 (BGA) Port Group 7

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P7_0	IO	TAUD0I5	TAUD0O5	TAUJ0I0	TAUJ0O0	—	—	ADCC1TRG	—	—	—	—	—	CSIH2SC	CSIH2SC	—	—
P7_1	IO	—	—	—	—	ENCA1TIN0	—	ADCC0TRG	—	—	—	—	—	—	CSIH2CSS1	—	—
P7_2	IO	TAUD0I7	TAUD0O7	TAUJ0I2	TAUJ0O2	—	—	—	—	—	—	SCI0RXD	—	—	CSIH2SO	—	—
P7_3	IO	—	—	—	—	ENCA1TIN1	—	—	—	—	—	—	—	—	CSIH2CSS2	—	—
P7_4	IO	TAUD0I8	TAUD0O8	TAUJ0I3	TAUJ0O3	—	—	—	—	—	—	—	SCI0TXD	CSIH2SI	—	—	—
P7_5	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	CSIH2CSS0	—	—
P7_6	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P7_7	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P7_8	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

2.2.2 C1M-A1 Port Function (QFP)

2.2.2.1 List of the C1M-A1 Port Registers (QFP)

Table 2.41 to **Table 2.48** show detailed bitmaps of control registers in each port group. In the bitmap field, "√" means an effective bit and "—" means a reserved one. Reserved bits are always read as value after resets. The write value also should be a value after reset.

Table 2.41 List of Registers in Port Group 0 of C1M-A1 (QFP)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	0	P0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR0	R/W	0000 0000H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR0	R	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM0	R/W	FFFFH	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT0	W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR0	R/W	0000 FFFFH	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR0	R/W	0000 0000H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC0	R/W	0000H	16	√	√	√	√	√	√	—	—	—	—	—	—	—	—	—	—	
PORT		PU0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD0	R/W	0000H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	0	PCR0_0	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_1	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_2	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_3	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_4	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_5	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_6	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_7	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_8	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_9	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR0_10	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_11	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_12	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_13	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_14	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR0_15	R/W	0000 0010H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.42 List of Registers in Port Group 1 of C1M-A1 (QFP)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	1	P1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR1	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM1	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT1	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR1	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC1	R/W	0000_H	16	√	√	√	√	√	√	—	—	√	√	√	√	√	√	—	—	
PORT		PU1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCA	PFCE	PFC
PORT	1	PCR1_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR1_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR1_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR1_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR1_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR1_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.43 List of Registers in Port Group 2 of C1M-A1 (QFP)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	2	P2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR2	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PM2	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCAE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PNOT2	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR2	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PBDC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	—	—	
PORT		PU2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCA	1 PFCE	0 PFC
PORT	2	PCR2_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR2_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR2_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.44 List of Registers in Port Group 3 of C1M-A1 (QFP)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	3	P3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR3	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PM3	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCAE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PNOT3	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR3	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PBDC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	—	√	
PORT		PU3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCA	PFCE	PFC
PORT	3	PCR3_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR3_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.45 List of Registers in Port Group 4 of C1M-A1 (QFP)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	4	P4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR4	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM4	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	—	—	—	
PORT		PFCE4	R/W	0000_H	16	—	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT4	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR4	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC4	R/W	0000_H	16	—	—	√	—	—	—	√	√	—	—	—	—	—	√	√	—	
PORT		PU4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	4	PCR4_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	—
PORT		PCR4_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—
PORT		PCR4_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—
PORT		PCR4_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√
PORT		PCR4_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√

Table 2.46 List of Registers in Port Group 5 of C1M-A1 (QFP)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	5	P5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PSR5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR5	R	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PM5	R/W	$FFFF_H$	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PMC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT5	W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR5	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	—	√	√	√	√	√	√	
PORT		PU5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PD5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
PORT	5	PCR5_0	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_1	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_2	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_3	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_4	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_5	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_6	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR5_7	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_8	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR5_9	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.47 List of Registers in Port Group 6 of C1M-A1 (QFP)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	6	P6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PSR6	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	Lower 16 bits
						—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	Upper 16 bits
PORT		PPR6	R	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PM6	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PMC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PFC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PFCE6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PFCAE6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PNOT6	W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PMSR6	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	Lower 16 bits
						—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	Upper 16 bits
PORT		PMCSR6	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	Lower 16 bits
						—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	Upper 16 bits
PORT		PIBC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PBDC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PU6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	
PORT		PD6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	—	√	—	√	—	—	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	6	PCR6_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR6_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR6_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√

Table 2.48 List of Registers in Port Group 7 of C1M-A1 (QFP)

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	7	P7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PSR7	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	Upper 16 bits
PORT		PPR7	R	0000_H	16	—	—	—	—	—	—	—	√	—	—	—	√	—	√	—	√	
PORT		PM7	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PMC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PFC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PFCE7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PFCAE7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PNOT7	W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PMSR7	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	Upper 16 bits
PORT		PMCSR7	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	Upper 16 bits
PORT		PIBC7	R/W	0000_H	16	—	—	—	—	—	—	—	√	—	—	—	√	—	√	—	√	
PORT		PBDC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PIPC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PU7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	
PORT		PD7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	—	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap													
						25	24	19	18	17	16	12	8	6	5	4	2	1	0
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC
PORT	7	PCR7_0	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR7_2	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR7_4	R/W	00000010_H	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR7_8	R/W	00000000_H	32	—	—	—	—	—	√	—	√	—	—	—	—	—	—

2.2.2.2 List of Pin Alternative Functions in C1M-A1 (QFP)

Table 2.49 to **Table 2.56** show the list of alternative functions of each port pin. In the tables, "—" means a reserved bit which cannot be selected.

Table 2.49 List of Pin Alternative Functions in C1M-A1 (QFP) Port Group 0

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P0_0	IO	TAUD0I0	TAUD0O0	TAUJ0I0	TAUJ0O0	—	—	—	—	—	—	—	—	—	—	—	—
P0_1	IO	TAUD0I1	TAUD0O1	TAUJ0I1	TAUJ0O1	—	—	—	—	—	—	—	—	—	—	—	—
P0_2	IO	TAUD0I2	TAUD0O2	TAUJ0I2	TAUJ0O2	—	—	—	—	—	—	TAPA3ESO	—	—	—	—	—
P0_3	IO	TAUD0I3	TAUD0O3	TAUJ0I3	TAUJ0O3	—	—	—	—	CAN2RX	—	—	—	INTP0	—	—	—
P0_4	IO	TAUD0I4	TAUD0O4	—	—	—	—	—	—	—	CAN2TX	—	—	INTP1	—	—	—
P0_5	IO	TAUD0I5	TAUD0O5	TAUJ0I0	TAUJ0O0	—	—	—	—	—	—	TAPA4ESO	—	INTP2	—	—	—
P0_6	IO	TAUD0I6	TAUD0O6	TAUJ0I1	TAUJ0O1	ENCA0E0	RDC3A0_OUT_U	—	—	—	—	—	—	INTP3	—	—	—
P0_7	IO	TAUD0I7	TAUD0O7	TAUJ0I2	TAUJ0O2	ENCA0E1	RDC3A0_OUT_V	—	—	—	—	—	—	INTP4	—	—	—
P0_8	IO	TAUD0I8	TAUD0O8	TAUJ0I3	TAUJ0O3	ENCA0EC	RDC3A0_OUT_W	—	—	—	—	—	—	INTP5	—	—	—
P0_9	IO	TAUD0I9	TAUD0O9	—	—	—	—	—	—	—	—	—	—	INTP6	—	—	—
P0_10	IO	TAUD0I10	TAUD0O10	—	TAPA0UP	—	—	—	—	—	—	—	—	INTP7	—	—	—
P0_11	IO	TAUD0I11	TAUD0O11	—	TAPA0UN	—	—	—	—	—	—	TSGTRG	—	—	—	—	—
P0_12	IO	TAUD0I12	TAUD0O12	—	TAPA0VP	—	—	—	—	RSENT0RX	RSENT0SPCO	—	—	—	—	—	—
P0_13	IO	TAUD0I13	TAUD0O13	—	TAPA0VN	—	—	—	—	—	RSENT0SPCO	—	—	—	—	—	—
P0_14	IO	TAUD0I14	TAUD0O14	—	TAPA0WP	—	—	—	—	RSENT1RX	RSENT1SPCO	—	—	—	—	—	—
P0_15	IO	TAUD0I15	TAUD0O15	—	TAPA0WN	—	—	—	—	—	RSENT1SPCO	—	—	—	—	—	—

Table 2.50 List of Pin Alternative Functions in C1M-A1 (QFP) Port Group 1

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P1_0	IO	TAUD1I0	TAUD1O0	—	—	ENCA0TIN0	—	—	TAUD1O1	—	TSG30O0	—	—	—	—	—	—
P1_1	IO	TAUD1I1	TAUD1O1	—	—	ENCA0TIN1	—	—	—	—	TSG30O7	—	—	—	—	—	—
P1_2	IO	TAUD1I2	TAUD1O2	—	—	—	—	—	TAUD1O3	—	TSG30O1	—	—	—	—	—	—
P1_3	IO	TAUD1I3	TAUD1O3	—	—	—	—	—	—	—	TSG30O3	—	—	—	—	—	—
P1_4	IO	TAUD1I4	TAUD1O4	—	—	—	—	—	TAUD1O5	—	TSG30O5	—	—	—	—	—	—
P1_5	IO	TAUD1I5	TAUD1O5	—	—	—	—	—	—	—	TSG30O2	—	—	—	—	—	—
P1_6	IO	TAUD1I6	TAUD1O6	—	—	—	—	—	TAUD1O7	—	TSG30O4	—	—	—	—	—	—
P1_7	IO	TAUD1I7	TAUD1O7	—	—	—	—	—	—	—	TSG30O6	—	—	—	—	—	—
P1_8	IO	TAUD1I8	TAUD1O8	—	—	—	—	—	TAUD1O9	—	—	—	—	—	—	—	—
P1_9	IO	TAUD1I9	TAUD1O9	—	—	—	—	—	—	—	—	TAPA4ESO	—	—	—	—	—
P1_10	IO	TAUD1I10	TAUD1O10	—	TAPA1UP	—	—	—	TAUD1O11	—	—	—	—	—	—	—	—
P1_11	IO	TAUD1I11	TAUD1O11	—	TAPA1UN	—	—	—	—	—	—	—	—	—	—	—	—
P1_12	IO	TAUD1I12	TAUD1O12	—	TAPA1VP	—	—	—	TAUD1O13	—	—	—	—	—	—	—	—
P1_13	IO	TAUD1I13	TAUD1O13	—	TAPA1VN	—	—	—	—	—	—	—	—	—	—	—	—
P1_14	IO	TAUD1I14	TAUD1O14	—	TAPA1WP	—	—	—	TAUD1O15	—	—	—	—	—	—	—	—
P1_15	IO	TAUD1I15	TAUD1O15	—	TAPA1WN	—	—	—	—	—	—	—	—	—	—	—	—

Table 2.51 List of Pin Alternative Functions in C1M-A1 (QFP) Port Group 2

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P2_0	IO	—	—	—	—	—	—	—	—	—	TSG31O0	—	—	INTP0	—	—	—
P2_1	IO	—	—	—	—	—	—	—	—	—	TSG31O7	—	—	INTP1	—	—	—
P2_2	IO	—	—	—	—	—	—	—	—	—	TSG31O1	—	—	INTP2	—	—	—
P2_3	IO	—	—	—	—	—	—	—	—	—	TSG31O3	—	—	INTP3	—	—	—
P2_4	IO	—	—	—	—	—	—	—	—	—	TSG31O5	—	—	INTP4	—	—	—
P2_5	IO	—	—	—	—	—	—	—	—	—	TSG31O2	—	—	INTP5	—	—	—
P2_6	IO	—	—	—	—	—	—	—	—	—	TSG31O4	—	—	INTP6	—	—	—
P2_7	IO	—	—	—	—	—	—	—	—	—	TSG31O6	—	—	INTP7	—	—	—

Table 2.52 List of Pin Alternative Functions in C1M-A1 (QFP) Port Group 3

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P3_0	IO	—	—	—	—	—	—	—	—	RSENT2RX	RSENT2SPCO	TAPA1ESO	—	—	—	—	—
P3_1	IO	—	—	—	—	—	—	ADCC0TRG	—	—	RSENT2SPCO	—	—	—	—	—	—
P3_2	IO	—	—	—	—	—	—	ADCC1TRG	—	—	—	—	—	—	—	—	—
P3_3	IO	—	—	—	—	ENCA1E0	—	—	—	—	—	—	—	—	—	—	—
P3_4	IO	—	—	—	—	ENCA1E1	—	ADCC0TRG	—	—	—	TAPA3ESO	—	—	—	—	—
P3_5	IO	—	—	—	—	ENCA1EC	—	—	—	—	—	TAPA0ESO	—	—	—	—	—
P3_6	IO	—	—	—	—	—	—	—	—	RSENT3RX	RSENT3SPCO	—	—	—	—	—	—
P3_7	IO	—	—	—	—	—	—	ADCC2TRG	—	—	RSENT3SPCO	—	—	—	—	—	—

Table 2.53 List of Pin Alternative Functions in C1M-A1 (QFP) Port Group 4

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P4_0	IO	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SI	—	—	—
P4_1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SO	—	—
P4_2	IO	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SC	CSIH1SC	—	—
P4_3	IO	—	—	—	—	—	—	—	—	CAN0RX	—	—	—	—	CSIH1CSS0	—	—
P4_4	IO	—	—	—	—	—	—	—	—	—	CAN0TX	—	—	—	CSIH1CSS1	—	—
P4_5	IO	—	—	—	—	—	—	—	—	CAN1RX	—	—	CSIH0CSS2	—	CSIH1CSS2	—	—
P4_6	IO	—	—	—	—	—	—	—	—	—	CAN1TX	—	CSIH0CSS3	—	CSIH1CSS3	—	—
P4_7	IO	—	—	—	—	—	—	—	—	—	—	CSIH0SI	—	$\overline{\text{CSIH1SSI}}$	—	—	—
P4_8	IO	—	—	—	—	—	—	—	—	—	—	—	CSIH0SO	CSIH1RYI	CSIH1RYO	—	—
P4_9	IO	—	—	—	—	—	—	—	—	—	—	CSIH0SC	CSIH0SC	—	—	—	—
P4_10	IO	—	—	—	—	—	TPBA0O	—	—	CAN2RX	—	—	CSIH0CSS0	—	—	—	—
P4_11	IO	—	—	—	—	—	—	—	—	—	CAN2TX	—	CSIH0CSS1	—	—	—	—
P4_12	IO	—	—	—	—	—	—	RLIN30RX	—	—	—	$\overline{\text{CSIH0SSI}}$	—	—	—	—	—
P4_13	IO	—	—	—	—	—	—	—	RLIN30TX	—	—	CSIH0RYI	CSIH0RYO	—	—	—	—
P4_14	IO	—	—	—	—	—	—	—	—	CAN3RX	—	—	—	—	—	—	—
P4_15	IO	—	—	—	—	—	—	—	—	—	CAN3TX	—	$\overline{\text{ERROROUT\_C}}$	—	—	—	—

Table 2.54 List of Pin Alternative Functions in C1M-A1 (QFP) Port Group 5

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P5_0	IO	—	—	—	—	—	—	RLIN32RX	—	—	—	SCI0RXD	—	—	CSIH2CSS3	—	—
P5_1	IO	—	—	—	—	—	—	—	RLIN32TX	—	—	—	SCI0TXD	$\overline{\text{CSIH2SSI}}$	—	—	—
P5_2	IO	—	—	—	—	—	—	RLIN31RX	—	—	—	SCI0SCK	SCI0SCK	CSIH2RYI	CSIH2RYO	—	—
P5_3	IO	—	—	—	—	—	—	—	RLIN31TX	—	—	SCI1SCK	SCI1SCK	—	—	—	—
P5_4	IO	—	—	—	—	—	—	RLIN30RX	—	—	—	SCI1RXD	—	—	—	—	—
P5_5	IO	—	—	—	—	—	—	—	RLIN30TX	—	—	—	SCI1TXD	—	$\overline{\text{ERROROUT\_C}}$	—	—
P5_6	IO	—	—	—	—	—	—	—	—	—	—	TAPA0ESO	—	—	—	—	—
P5_7	IO	—	—	—	—	—	—	—	—	—	—	SCI2SCK	SCI2SCK	—	—	—	—
P5_8	IO	—	—	—	—	—	—	—	—	—	—	SCI2RXD	—	—	—	—	—
P5_9	IO	—	—	—	—	—	—	—	—	—	—	—	SCI2TXD	—	—	—	—

Table 2.55 List of Pin Alternative Functions in C1M-A1 (QFP) Port Group 6

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P6_2	IO	—	—	—	—	ENCA1TIN0	RDC3A0_OUT_W	ADCC0TRG	—	—	—	—	—	—	—	—	—
P6_4	IO	—	—	—	—	ENCA1TIN1	RDC3A0_OUT_V	—	—	—	—	—	—	—	—	—	—
P6_6	IO	—	—	—	—	—	RDC3A0_OUT_U	—	—	—	—	TAPA0ESO	—	—	—	—	—

Table 2.56 List of Pin Alternative Functions in C1M-A1 (QFP) Port Group 7

Port	General Purpose Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	ALT_IN8	ALT_OUT8
P7_0	IO	TAUD0I5	TAUD0O5	TAUJ0I0	TAUJ0O0	—	—	ADCC1TRG	—	—	—	—	—	CSIH2SC	CSIH2SC	—	—
P7_2	IO	TAUD0I7	TAUD0O7	TAUJ0I2	TAUJ0O2	—	—	—	—	—	—	SCI0RXD	—	—	CSIH2SO	—	—
P7_4	IO	TAUD0I8	TAUD0O8	TAUJ0I3	TAUJ0O3	—	—	—	—	—	—	—	SCI0TXD	CSIH2SI	—	—	—
P7_8	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

2.3 DNF

Digital Noise Filter (DNF) eliminates digital noise from external input signals. This product incorporates peripheral function DNF.

2.3.1 Example of Noise Elimination

Figure 2.7 shows an example of noise elimination in peripheral function DNF. In this example, the sampling clock, the sampling count, and the current output level are set to 1/2 of the DNF input clock, two (twice), and low, respectively. "○" in the figure means that high level is detected.

In input examples 1, 2, and 3, the output level changes from low to high because the same level is detected twice in a row through sampling. In input examples 4, 5, and 6, on the other hand, the same level is not detected twice consecutively. Therefore, these inputs are regarded as noise and the input signal state is eliminated.

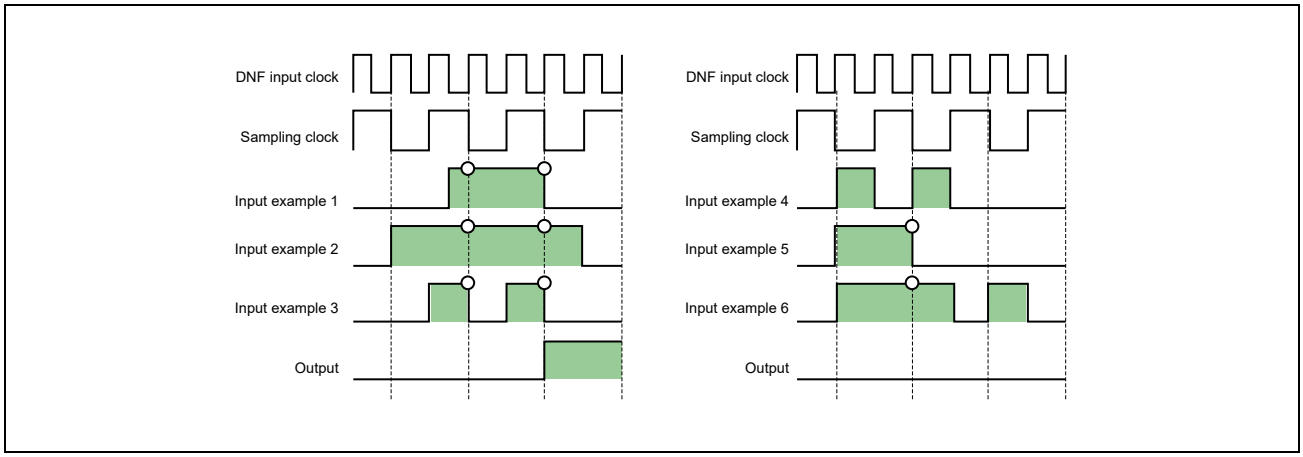


Figure 2.7 Timing Chart of Digital Noise Elimination

2.3.2 Peripheral Function DNF

2.3.2.1 Overview of Peripheral Function DNF

This DNF eliminates noise from the input function pins for the peripheral function.

Peripheral function DNF has the following functions:

- Eliminates digital noise from input signals and outputs noiseless signals.
- Selects whether to output signals from which digital noise is eliminated or signals containing digital noise.
- Selects the digital noise elimination width from 2, 3, 4, and 5 counts of the sampling clock.
- Selects five types of sampling frequency shown below:
  1/1, 1/2, 1/4, 1/8, and 1/16 of the DNF input clock.
- The conditions for noise elimination can be set by each channel via the registers.
- The DNF input clock of DNF group number 0 is a low-speed peripheral clock.
- The DNF input clock of DNF group number 1, 2, 3, 4, 5, 6, and 7 is an unmodulated high-speed peripheral clock.
- The DNF input clock of DNF group number 8 is non-modulated low-speed peripheral clock.

2.3.2.2 Details of the Control Registers

Base addresses of peripheral function DNF are listed in the following table.

Register addresses of peripheral function DNF are given as offsets from the base addresses in general.

Table 2.57 Register Base Address

Base Address Name	Base address
<DNF0_base>	FFC3 0000_H
<DNF1_base>	FFC3 0400_H
<DNF2_base>	FFC3 0800_H
<DNF3_base>	FFC3 $0C00_H$
<DNF4_base>	FFC3 1000_H
<DNF5_base>	FFC3 1400_H
<DNF6_base>	FFC3 1800_H
<DNF7_base>	FFC3 $1C00_H$
<DNF8_base>	FFC3 2000_H

2.3.2.3 DNFP01nCTLm — Digital Noise Elimination Control Register

This register sets conditions for noise elimination of channel number m in DNF group number n.

Access: This register is readable/writable in 8-bit units.

Address: <DNFn_base> + 4_H × m (m: channel number)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NFEN	SLST[1:0]		—	—	PRS[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W

Table 2.58 DNFP01nCTLm Register Contents

Bit Position	Bit Name	Function
7	NFEN	Enables/disables digital noise elimination. 0: Digital noise is not eliminated. 1: Digital noise is eliminated.
6, 5	SLST[1:0]	Specifies the sampling count for digital noise elimination. 00: Sampling count = 2 01: Sampling count = 3 10: Sampling count = 4 11: Sampling count = 5
4, 3	Reserved	When read, 0 is read. When writing, write 0.
2 to 0	PRS[2:0]	Specifies the sampling clock division ratio for digital noise elimination. 000: DNF input clock/1 001: DNF input clock/2 010: DNF input clock/4 011: DNF input clock/8 100: DNF input clock/16 Other than the above: Setting is prohibited.

2.3.2.4 Setting Procedures of Peripheral Function DNF

The following shows the procedures to set peripheral function DNF. If the input level to a pin varies during (4) and (5), an unexpected signal may be input to the peripheral function. Therefore, the corresponding flag at the peripheral function side or the like should be cleared in (6).

(1) Set the PRS2 to PRS0 bits (bits 2 to 0) and the SLST1 and SLST0 bits (bits 6 and 5) in the DNFP01nCTLm register.

(2) Set the NFEN bit (bit 7) in the DNFP01nCTLm register. This setting can be made at the same time as (1).

(3) Set a port register to select an alternative function.

(4) Wait for the following time: sampling clock period × the number of samples + DNF input clock period × 2.

(5) Enable an operation of a peripheral function in the destination to which DNF is connected.

(6) Clear the flag of peripheral function, and the like.

To change the setting while the DNF is in operation, clear the NFEN bit (bit 7) in the DNFP01nCTLm register first, and then execute the procedure in steps (1) to (6) above to remake the settings.

2.3.2.5 Peripheral Function Pin for DNF Insertion

The signals shown in the table below are the targets of DNF insertion.

Table 2.59 DNF Insertion Targets (1/3)

DNF Group Number n	DNF Channel Number m	Target Pins for DNF Insertion		
		Pin Name	Pin Function	Peripheral IP
0	0	INTP0	External interrupt input	INTC
	1	INTP1		
	2	INTP2		
	3	INTP3		
	4	INTP4		
	5	INTP5		
	6	INTP6		
	7	INTP7		
	8	TAPA0ESO	Emergency Hi-Z request input (for TAUD0 PWM)	TAPA (via PIC1B)
	9	TAPA1ESO	Emergency Hi-Z request input (for TAUD1 PWM)	
	10	TAPA2ESO*1	Emergency Hi-Z request input (for TAUD2 PWM)	
	11	TAPA3ESO	Emergency Hi-Z request input (for TSG30 PWM)	
	12	TAPA4ESO	Emergency Hi-Z request input (for TSG31 PWM)	
	13	TAPA5ESO*1	Emergency Hi-Z request input (for TSG32 PWM)	
1	0	ADCC0TRG	SAR-AD conversion startup trigger input	SAR-AD0
	1	ADCC1TRG		SAR-AD1
	2	ADCC2TRG		SAR-AD2
	3	TSGTRG	PIC external trigger input	PIC1B
2	0	ENCA0TIN0	Capture trigger input	ENCA0
	1	ENCA0TIN1		
	2	ENCA0E0	Encoder input	
	3	ENCA0E1		
	4	ENCA0EC		
	5	ENCA1TIN0	Capture trigger input	ENCA1
	6	ENCA1TIN1		
	7	ENCA1E0	Encoder input	
	8	ENCA1E1		
	9	ENCA1EC		

Table 2.59 DNF Insertion Targets (2/3)

DNF Group Number n	DNF Channel Number m	Target Pins for DNF Insertion		
		Pin Name	Pin Function	Peripheral IP
3	0	TAUD0I0	TAUD0 channel input	TAUD0
	1	TAUD0I1		
	2	TAUD0I2		
	3	TAUD0I3		
	4	TAUD0I4		
	5	TAUD0I5		
	6	TAUD0I6		
	7	TAUD0I7		
	8	TAUD0I8		
	9	TAUD0I9		
	10	TAUD0I10		
	11	TAUD0I11		
	12	TAUD0I12		
	13	TAUD0I13		
	14	TAUD0I14		
	15	TAUD0I15		
4	0	TAUD1I0	TAUD1 channel input	TAUD1
	1	TAUD1I1		
	2	TAUD1I2		
	3	TAUD1I3		
	4	TAUD1I4		
	5	TAUD1I5		
	6	TAUD1I6		
	7	TAUD1I7		
	8	TAUD1I8		
	9	TAUD1I9		
	10	TAUD1I10		
	11	TAUD1I11		
	12	TAUD1I12		
	13	TAUD1I13		
	14	TAUD1I14		
	15	TAUD1I15		
5	0	TAUJ0I0	TAUJ0 channel input	TAUJ0
	1	TAUJ0I1		
	2	TAUJ0I2		
	3	TAUJ0I3		

Table 2.59 DNF Insertion Targets (3/3)

DNF Group Number n	DNF Channel Number m	Target Pins for DNF Insertion		
		Pin Name	Pin Function	Peripheral IP
5	4	TAUJ1I0*1	TAUJ1 channel input	TAUJ1
	5	TAUJ1I1*1		
	6	TAUJ1I2*1		
	7	TAUJ1I3*1		
6	0	TAUD2I0*1	TAUD2 channel input	TAUD2
	1	TAUD2I1*1		
	2	TAUD2I2*1		
	3	TAUD2I3*1		
	4	TAUD2I4*1		
	5	TAUD2I5*1		
	6	TAUD2I6*1		
	7	TAUD2I7*1		
	8	TAUD2I8*1		
	9	TAUD2I9*1		
	10	TAUD2I10*1		
	11	TAUD2I11*1		
	12	TAUD2I12*1		
	13	TAUD2I13*1		
	14	TAUD2I14*1		
	15	TAUD2I15*1		
7	0	TAUD3I0*1	TAUD3 channel input	TAUD3
	1	TAUD3I1*1		
	2	TAUD3I2*1		
	3	TAUD3I3*1		
	4	TAUD3I4*1		
	5	TAUD3I5*1		
	6	TAUD3I6*1		
	7	TAUD3I7*1		
	8	TAUD3I8*1		
	9	TAUD3I9*1		
	10	TAUD3I10*1		
	11	TAUD3I11*1		
	12	TAUD3I12*1		
	13	TAUD3I13*1		
	14	TAUD3I14*1		
	15	TAUD3I15*1		
8	0	RSENT0RX	RSENT0 receive data input	RSENT0
	1	RSENT1RX	RSENT1 receive data input	RSENT1
	2	RSENT2RX	RSENT2 receive data input	RSENT2
	3	RSENT3RX	RSENT3 receive data input	RSENT3

Note 1. C1M-A1 does not have this pin.

2.4 Pin Description

2.4.1 Overview

This subsection describes pin functions, external pin lists, and external pin states at a reset and in other each operating status.

2.4.2 List of Pin Functions

Function of each pin is described below.

Table 2.60 C1M-A2 Pin Function (1/3)

Pin Name	I/O	Function
AnVREFH (n = 0 to 2)	—	ADCCn voltage supply and reference voltage
AnVSS (n = 0 to 2)	—	ADCCn ground
$\overline{\text{ADCCnTRG}}$ (n = 0 to 2)	I	ADCCn trigger
ADCCnIpq (n = 0 to 2, p = 0 to 3, q = 0 to 3)	I	ADCCn input channel pq
AUDATAm (m = 0 to 3)	IO	AUDR command / address / data / flag m
AUDCK	I	AUDR clock
$\overline{\text{AUDRST}}$	I	AUDR reset
$\overline{\text{AUDSYNC}}$	I	AUDR timing control
AnVCC (n = 0 to 2)	—	ADCCn voltage supply
CANmRX (m = 0 to 3)	I	CANm receive data input
CANmTX (m = 0 to 3)	O	CANm transmit data output
CSIHnCSS0 (n = 0 to 2)	O	CSIHn serial peripheral chip select signal 0
CSIHnCSS1 (n = 0 to 2)	O	CSIHn serial peripheral chip select signal 1
CSIHnCSS2 (n = 0 to 2)	O	CSIHn serial peripheral chip select signal 2
CSIHnCSS3 (n = 0 to 2)	O	CSIHn serial peripheral chip select signal 3
CSIHnRYI (n = 0 to 2)	I	CSIHn ready (1) / busy (0) input signal
CSIHnRYO (n = 0 to 2)	O	CSIHn ready (1) / busy (0) output signal
CSIHnSC (n = 0 to 2)	IO	CSIHn serial clock signal
CSIHnSI (n = 0 to 2)	I	CSIHn serial data input
CSIHnSO (n = 0 to 2)	O	CSIHn serial data output
$\overline{\text{CSIHnSSI}}$ (n = 0 to 2)	I	CSIHn slave select input signal
$\overline{\text{DCURDY}}$	O	Debug ready
DCUTCK	I	Debug clock
DCUTDI	I	Debug data input
DCUTDO	O	Debug data output
DCUTMS	I	Debug mode select
$\overline{\text{DCUTRST}}$	I	Debug reset
FLSCI3TX (FPDT)	O	Transmit data output
FLSCI3RX (FPDR)	I	Receive data input
FLSCI3SCK (FPCK)	I	Serial clock input
ENCAnE0 (n = 0, 1)	I	ENCAn encoder input (count pulse 0)
ENCAnE1 (n = 0, 1)	I	ENCAn encoder input (count pulse 1)

Table 2.60 C1M-A2 Pin Function (2/3)

Pin Name	I/O	Function
ENCAnTINm (n = 0, 1, m = 0, 1)	I	ENCAn capture trigger input nm
ENCAnEC (n = 0, 1)	I	ENCAn encoder input (clear pulse)
$\overline{\text{ERROROUT\_M}}$	O	ECM error output (main)
$\overline{\text{ERROROUT\_C}}$	O	ECM error output (checker)
FLMODE	I	Operating mode select pin
MD0	I	Operating mode select pin 0
MD1	I	Operating mode select pin 1
INTPm (m = 0 to 7)	I	External interrupt input m
LPDCLK	I	LPD clock input (4-pin mode)
LPDCLKOUT	O	LPD clock output (4-pin mode)
LPDI	I	LPD data input (4-pin mode)
LPDO	O	LPD data output (4-pin mode)
$\overline{\text{LPDRST}}$	I	LPD Reset (4-pin mode)
P0_m (m = 0 to 15)	IO	Port 0_m
P1_m (m = 0 to 15)	IO	Port 1_m
P2_m (m = 0 to 7)	IO	Port 2_m
P3_m (m = 0 to 7)	IO	Port 3_m
P4_m (m = 0 to 15)	IO	Port 4_m
P5_m (m = 0 to 9)	IO	Port 5_m
P6_m (m = 0 to 15)	IO	Port 6_m
P7_m (m = 0 to 7)	IO	Port 7_m
P7_m (m = 8)	I	Port 7_m
SYSVCC	—	Voltage supply for system and PLL
VCC	—	Voltage supply for oscillator, flash memory, and port buffer
VDD	—	Voltage regulators voltage supply
VSS	—	Ground
RVCC	—	Voltage supply for RDC
RVSS	—	Ground for RDC
$\overline{\text{RESET}}$	I	External reset input
RDC3AnCOM (n = 0, 1)	IO	Excitation common signal input/output
RDC3AnCOSMNT (n = 0, 1)	O	COS-side monitoring signal output
RDC3AnRSO (n = 0, 1)	IO	Excitation signal input/output
RDC3AnS1 (n = 0, 1)	I	Resolver signal input
RDC3AnS2 (n = 0, 1)	I	Resolver signal input
RDC3AnS3 (n = 0, 1)	I	Resolver signal input
RDC3AnS4 (n = 0, 1)	I	Resolver signal input
RDC3AnSINMNT (n = 0, 1)	O	SIN-side monitoring signal output
RLIN3mRX (m = 0 to 2)	I	RLIN3m receive data input
RLIN3mTX (m = 0 to 2)	O	RLIN3m transmit data output
SCInRXD (n = 0 to 2)	I	SCIn receive data

Table 2.60 C1M-A2 Pin Function (3/3)

Pin Name	I/O	Function
SCInSCK (n = 0 to 2)	IO	SCIn clock
SCInTXD (n = 0 to 2)	O	SCIn transmit data
TAPAnESO (n = 0 to 5)	I	Hi-Z control
TAPAnUN (n = 0 to 2)	O	Motor control output U phase (negative)
TAPAnUP (n = 0 to 2)	O	Motor control output U phase (positive)
TAPAnVN (n = 0 to 2)	O	Motor control output V phase (negative)
TAPAnVP (n = 0 to 2)	O	Motor control output V phase (positive)
TAPAnWN (n = 0 to 2)	O	Motor control output W phase (negative)
TAPAnWP (n = 0 to 2)	O	Motor control output W phase (positive)
TPBAnO (n = 0, 1)	O	TPBAn channel output
TAUDnIm (n = 0 to 3, m = 0 to 15)	I	TAUDn channel input m
TAUDnOm (n = 0 to 3, m = 0 to 15)	O	TAUDn channel output m
TAUJnIm (n = 0 to 1, m = 0 to 3)	I	TAUJn channel input m
TAUJnOm (n = 0 to 1, m = 0 to 3)	O	TAUJn channel output m
TSG3nOm (n = 0 to 2, m = 0 to 7)	O	TSG3n channel output m
X1, X2	—	Crystal oscillator connections
RSENTnRX (n = 0 to 3)	I	SENT input
RSENTnSPCO (n = 0 to 3)	O	SENT control output
RDC3An_OUT_U (n = 0, 1)	O	RDC U-phase output
RDC3An_OUT_V (n = 0, 1)	O	RDC V-phase output
RDC3An_OUT_W (n = 0, 1)	O	RDC W-phase output

CAUTION

In C1M-A2, use SCI30 with the alternative function pins within the same port group.

- When serial clock I/O signals are used
  - P5_0(SCI0RXD), P5_1(SCI0TXD), P5_2(SCI0SCK)
- When serial clock I/O signals are not used
  - P5_0(SCI0RXD), P5_1(SCI0TXD)
  - P7_2(SCI0RXD), P7_4(SCI0TXD)

Table 2.61 C1M-A1 Pin Function (1/3)

Pin Name	I/O	Function
AnVREFH (n = 0 to 2)	—	ADCCn voltage supply and reference voltage
AnVSS (n = 0 to 2)	—	ADCCn ground
$\overline{\text{ADCCnTRG}}$ (n =0 to 2)	I	ADCCn trigger
ADCC0Ipq (p = 0 to 3, q = 0 to 3) This excludes the combination of p = 2 and q = 2 or 3 and of p = 3 and q = 1 to 3.	I	ADCC0 input channel pq
ADCC1Ipq (p = 0 to 3, q = 0 to 3) This excludes the combination of p = 3 and q = 0 or 3.	I	ADCC1 input channel pq
ADCC2Ipq (p = 1, q = 1 to 3)	I	ADCC2 input channel pq
AUDATAm (m = 0 to 3)	IO	AUDR command / address / data / flag m
AUDCK	I	AUDR clock
$\overline{\text{AUDRST}}$	I	AUDR reset
$\overline{\text{AUDSYNC}}$	I	AUDR timing control
AnVCC (n = 0 to 2)	—	ADCCn voltage supply
CANmRX (m = 0 to 3)	I	CANm receive data input
CANmTX (m = 0 to 3)	O	CANm transmit data output
CSIHnCSS0 (n = 0 to 2)	O	CSIHn serial peripheral chip select signal 0
CSIHnCSS1 (n = 0 to 2)	O	CSIHn serial peripheral chip select signal 1
CSIHnCSS2 (n = 0 to 2)	O	CSIHn serial peripheral chip select signal 2
CSIHnCSS3 (n = 0 to 2)	O	CSIHn serial peripheral chip select signal 3
CSIHnRYI (n = 0 to 2)	I	CSIHn ready (1) / busy (0) input signal
CSIHnRYO (n = 0 to 2)	O	CSIHn ready (1) / busy (0) output signal
CSIHnSC (n = 0 to 2)	IO	CSIHn serial clock signal
CSIHnSI (n = 0 to 2)	I	CSIHn serial data input
CSIHnSO (n = 0 to 2)	O	CSIHn serial data output
$\overline{\text{CSIHnSSI}}$ (n = 0 to 2)	I	CSIHn slave select input signal
$\overline{\text{DCURDY}}$	O	Debug ready
DCUTCK	I	Debug clock
DCUTDI	I	Debug data input
DCUTDO	O	Debug data output
DCUTMS	I	Debug mode select
$\overline{\text{DCUTRST}}$	I	Debug reset
FLSCI3TX (FPDT)	O	Transmit data output
FLSCI3RX (FPDR)	I	Receive data input
FLSCI3SCK (FPCK)	I	Serial clock input
ENCAnE0 (n = 0, 1)	I	ENCAn encoder input (count pulse 0)
ENCAnE1 (n = 0, 1)	I	ENCAn encoder input (count pulse 1)

Table 2.61 C1M-A1 Pin Function (2/3)

Pin Name	I/O	Function
ENCAnTINm (n = 0, 1, m = 0, 1)	I	ENCAn capture trigger input nm
ENCAnEC (n = 0, 1)	I	ENCAn encoder input (clear pulse)
$\overline{\text{ERROROUT\_M}}$	O	ECM error output (main)
$\overline{\text{ERROROUT\_C}}$	O	ECM error output (checker)
FLMODE	I	Operating mode select pin
MD0	I	Operating mode select pin 0
MD1	I	Operating mode select pin 1
INTPm (m = 0 to 7)	I	External interrupt input m
LPDCLK	I	LPD clock input (4-pin mode)
LPDCLKOUT	O	LPD clock output (4-pin mode)
LPDI	I	LPD data input (4-pin mode)
LPDO	O	LPD data output (4-pin mode)
$\overline{\text{LPDRST}}$	I	LPD Reset (4-pin mode)
P0_m (m = 0 to 15)	IO	Port 0_m
P1_m (m = 0 to 15)	IO	Port 1_m
P2_m (m = 0 to 7)	IO	Port 2_m
P3_m (m = 0 to 7)	IO	Port 3_m
P4_m (m = 0 to 15)	IO	Port 4_m
P5_m (m = 0 to 9)	IO	Port 5_m
P6_m (m = 2, 4, 6)	IO	Port 6_m
P7_m (m = 0, 2, 4)	IO	Port 7_m
P7_m (m = 8)	I	Port 7_m
SYSVCC	—	Voltage supply for system and PLL
VCC	—	Voltage supply for oscillator, flash memory, and port buffer
VDD	—	Voltage regulators voltage supply
VSS	—	Ground
RVCC	—	Voltage supply for RDC
RVSS	—	Ground for RDC
$\overline{\text{RESET}}$	I	External reset input
RDC3AnCOM (n = 0)	IO	Excitation common signal input/output
RDC3AnCOSMNT (n = 0)	O	COS-side monitoring signal output
RDC3AnRSO (n = 0)	IO	Excitation signal input/output
RDC3AnS1 (n = 0)	I	Resolver signal input
RDC3AnS2 (n = 0)	I	Resolver signal input
RDC3AnS3 (n = 0)	I	Resolver signal input
RDC3AnS4 (n = 0)	I	Resolver signal input
RDC3AnSINMNT (n = 0)	O	SIN-side monitoring signal output
RLIN3mRX (m = 0 to 2)	I	RLIN3m receive data input
RLIN3mTX (m = 0 to 2)	O	RLIN3m transmit data output
SCInRXD (n = 0 to 2)	I	SCIn receive data

Table 2.61 C1M-A1 Pin Function (3/3)

Pin Name	I/O	Function
SCInSCK (n = 0 to 2)	IO	SCIn clock
SCInTXD (n = 0 to 2)	O	SCIn transmit data
TAPAnESO (n = 0 to 1,3 to 4)	I	Hi-Z control
TAPAnUN (n = 0 to 1)	O	Motor control output U phase (negative)
TAPAnUP (n = 0 to 1)	O	Motor control output U phase (positive)
TAPAnVN (n = 0 to 1)	O	Motor control output V phase (negative)
TAPAnVP (n = 0 to 1)	O	Motor control output V phase (positive)
TAPAnWN (n = 0 to 1)	O	Motor control output W phase (negative)
TAPAnWP (n = 0 to 1)	O	Motor control output W phase (positive)
TPBAnO (n = 0)	O	TPBAn channel output
TAUDnIm (n = 0 to 1, m = 0 to 15)	I	TAUDn channel input m
TAUDnOm (n = 0 to 1, m = 0 to 15)	O	TAUDn channel output m
TAUJnIm (n = 0, m = 0 to 3)	I	TAUJn channel input m
TAUJnOm (n = 0, m = 0 to 3)	O	TAUJn channel output m
TSG3nOm (n = 0 to 1, m = 0 to 7)	O	TSG3n channel output m
X1, X2	—	Crystal oscillator connections
RSENTnRX (n = 0 to 3)	I	SENT input
RSENTnSPCO (n = 0 to 3)	O	SENT control output
RDC3An_OUT_U (n = 0)	O	RDC U-phase output
RDC3An_OUT_V (n = 0)	O	RDC V-phase output
RDC3An_OUT_W (n = 0)	O	RDC W-phase output

CAUTION

In C1M-A1, use SCI30 with the alternative function pins within the same port group.

- When serial clock I/O signals are used
  - P5_0(SCI0RXD), P5_1(SCI0TXD), P5_2(SCI0SCK)
- When serial clock I/O signals are not used
  - P5_0(SCI0RXD), P5_1(SCI0TXD)
  - P7_2(SCI0RXD), P7_4(SCI0TXD)

2.4.3 Pin State

Definition of Reset State

In description of pin state, each reset state is defined as shown in **Table 2.62**.

Table 2.62 Definition of Reset State

Reset State	Definition
External reset	Reset state from an external pin ($\overline{\text{RESET}}$ = L)
Internal reset	Between external reset release to internal reset release
After internal reset release	State where internal reset is released

Table 2.62 and **Table 2.63** show detailed pin states. Some pins may be excluded depending on grades and packages of this product. For the pins included in each product, see Section **1.2, Pin Connection Diagram (Top View).**

Table 2.63 Pin State (1/2)

Pin Function		Pin Function		
		$\overline{\text{RESET}}$ = L	$\overline{\text{RESET}}$ = H	
Classification	Pin Name	External Reset State	Before Internal Reset Release	After Internal Reset Release
Clock	X1	I	I	I
	X2	O	O	O
System control	$\overline{\text{RESET}}$	I (Pull-down)	I (Pull-down)	I (Pull-down)
	MD0	I (Pull-down)	I (Pull-down)	I (Pull-down)
	MD1	I (Pull-down)	I (Pull-down)	I (Pull-down)
	FLMODE	I (Pull-down)	I (Pull-down)	I (Pull-down)
ECM	$\overline{\text{ERROROUT\_M}}$	O	O	O
General-purpose	P0_x	Z	Z	Z
I/O ports	P1_x	Z	Z	Z
	P2_x	Z	Z	Z
	P3_x	Z	Z	Z
	P4_x	Z	Z	Z
	P5_x	Z	Z	Z
	P6_x	Z	Z	Z
	P7_x (x = 0 to 7)	Z	Z	Z
	P7_8	I (Pull-down)	I (Pull-down)	I (Pull-down)
SAR A/D	ADCC0Ixx	Z	Z	Z
	ADCC1Ixx	Z	Z	Z
	ADCC2Ixx	Z	Z	Z
Resolver signal input	RDC3A0Sx	Z	Z	Z
	RDC3A1Sx	Z	Z	Z
Excitation signal output	RDC3A0RSO, RDC3A0COM	Z	Z	Z
	RDC3A1RSO, RDC3A1COM	Z	Z	Z
AUD RAM	$\overline{\text{AUDRST}}$	I (Pull-down)	I (Pull-down)	I (Pull-down)
monitoring	AUDCK	I (Pull-up)	I (Pull-up)	I (Pull-up)
	$\overline{\text{AUDSYNC}}$	I (Pull-up)	I (Pull-up)	I (Pull-up)
	AUDATA0 to AUDATA3	I (Pull-up)	I (Pull-up)	I (Pull-up)

Table 2.63 Pin State (2/2)

Pin Function			Pin Function		
Classification	Pin Name		$\overline{\text{RESET}}$ = L	$\overline{\text{RESET}}$ = H	
			External Reset State	Before Internal Reset Release	After Internal Reset Release
Debug system	DCUTDI/LPDI/FLSCI3RX	Nexus: DCUTDI	Z	I (Pull-up)	I (Pull-up)
		LPD-4pin: LPDI	Z	I (Pull-up)	I (Pull-up)
		Writer I/F: FLSCI3RX	Z	Z	Z
		BSCAN: DCUTDI	I (Pull-up)	I (Pull-up)	I (Pull-up)
	DCUTDO/LPDO/ FLSCI3TX	Nexus: DCUTDO	Z	Z	Z
		LPD-4pin: LPDO	Z	O	O
		Writer I/F: FLSCI3TX	Z	Z	Z
		BSCAN: DCUTDO	Z	Z	Z
	DCUTCK/LPDCLK/ FLSCI3SCK	Nexus: DCUTCK	Z	I (Pull-up)	I (Pull-up)
		LPD-4pin: LPDCLK	Z	I (Pull-up)	I (Pull-up)
		Writer I/F: FLSCI3SCK	Z	Z	Z
		BSCAN: DCUTCK	I (Pull-up)	I (Pull-up)	I (Pull-up)
	DCUTMS	Nexus: DCUTMS	Z	I (Pull-up)	I (Pull-up)
		LPD-4pin: $\overline{\text{EVTO}}$ *1	Z	O	O
		Writer I/F: (w/o function)	Z	Z	Z
		BSCAN: DCUTMS	I (Pull-up)	I (Pull-up)	I (Pull-up)
	$\overline{\text{DCUTRST}}$ / $\overline{\text{LPDRST}}$	Nexus: $\overline{\text{DCUTRST}}$	I (Pull-down)	I (Pull-down)	I (Pull-down)
		LPD-4pin: $\overline{\text{LPDRST}}$			
		Writer I/F: (w/o function)			
		BSCAN: $\overline{\text{DCUTRST}}$			
	$\overline{\text{DCURDY}}$ /LPDCLKOUT	Nexus: $\overline{\text{DCURDY}}$	Z	O	O
		LPD-4pin: LPDCLKOUT	Z	O	O
		Writer I/F: (w/o function)	Z	Z	Z
		BSCAN:(w/o function)	Z	Z	Z

Note: I: Input
O: Output
Z: High impedance
Pull-up: On-chip pull-up resistor
Pull-down: On-chip pull-down resistor

Note 1. $\overline{\text{EVTO}}$ is not used in this device.

2.4.4 Handling of Unused Pins

Table 2.64 shows an example handling of unused pins.

Table 2.64 Example Handling of Unused Pins (1/2)

Category	Pins	IO	Example handling of unused pins	Internal pull-up/pull-down resistor
Clock	X1	I	(Required)	None
	X2	O	(Required)	None
System control	$\overline{\text{RESET}}$	I	(Required)	An internal pull-down resistor is included.
	MD0, FLMODE	I	(Required. When they are used in user boot mode, separately connect them with VSS via resistors.)	An internal pull-down resistor is included.
	MD1	I	Separately connect the pin with VSS via a resistor.	An internal pull-down resistor is included.
ECM	$\overline{\text{ERROROUT\_M}}$	O	Leave the pin open.	None
General-purpose I/O port	P0_m, P1_m, P2_m, P3_m, P4_m, P5_m, P6_m, P7_m (m = 0 to 7)	IO	[Input mode] • Leave the pins open, and disable “input enable” (PMCn_m = 0, PMn_m = 1, and PIBCn_m = 0 (values after reset)). • Leave the pins open, and enable the internal pull-up/pull-down resistors (use PUn_m and PDn_m). • Separately connect each pin with the power supply/GND via a resistor. [Output mode] • Leave the pins open.	Internal pull-up/pull-down resistors that can be set by the registers are included.
General-purpose input port	P7_8	I	• Leave the pin open. • Separately connect the pin with VSS via a resistor.	An internal pull-down resistor is included.
ADCC	ADCCnIpq (Analog input-only)	I	Leave the pin open.	None
RDC3A	RDC3AnCOM, RDC3AnRSO	I/O	Leave the pins open and stop the analog circuit.	None
	RDC3AnS1, RDC3AnS2, RDC3AnS3, RDC3AnS4	I	Leave the pins open and stop the analog circuit.	None
Debug system (AUDRAM)	$\overline{\text{AUDRST}}$	I	• Leave the pin open. • Separately connect the pin with VSS via a resistor.	An internal pull-down resistor is included.
	AUDCK, $\overline{\text{AUDSYNC}}$	I	• Leave the pin open. • Separately connect the pin with VCC via a resistor.	An internal pull-up resistor is included.
	AUDATAn	IO	Leave the pin open.	An internal pull-up resistor is included.

Table 2.64 Example Handling of Unused Pins (2/2)

Category	Pins	IO	Example handling of unused pins	Internal pull-up/pull-down resistor
Debug system (NEXUS/LPD)	DCUTDI	I	• Leave the pin open. • Separately connect the pin with VCC via a resistor. (Serial programming mode is disabled.)	An internal pull-up resistor is included.
	DCUTDO	O	• Leave the pin open. (Serial programming mode is disabled.)	None
	DCUTCK	I	• Leave the pin open. • Separately connect the pin with VCC via a resistor. (Serial programming mode is disabled.)	An internal pull-up resistor is included.
	$\overline{\text{DCUTMS}}$	I	• Leave the pin open. • Separately connect the pin with VCC via a resistor.	An internal pull-up resistor is included.
	$\overline{\text{DCUTRST}}$	I	Separately connect the pin with VSS via a resistor.	An internal pull-down resistor is included.
	DCURDY	O	Leave the pin open.	None
N.C.	A0VSS(N.C.), A2VSS(N.C.), VSS(N.C.)	—	Connect the pins to the power supply of the same name as those that do not include N.C.	None
Power system	SYSVCC	—	(Required)	
	VCC	—	(Required)	
	VDD	—	(Required)	
	VSS	—	(Required)	
	AnVCC	—	(Required)	
	AnVSS	—	(Required)	
	AnVREFH	—	(Required)	
	RVCC	—	(Required)	
	RVSS	—	(Required)	

Note 1. When a pull-up/pull-down is performed via a resistor outside the product, set the resistance value to 1 kΩ or more.

Note 2. When a pull-up/pull-down is also performed via a resistor outside the product for a pin that includes internal pull-up/pull-down resistors, be careful so that the resistance voltage is not divided.

Section 3 CPU System

3.1 Overview

3.1.1 Block Configuration

A block diagram of RH850/C1M-A is shown in **Figure 3.1**.

RH850/C1M-A1 does not have CPU2 (PE2).

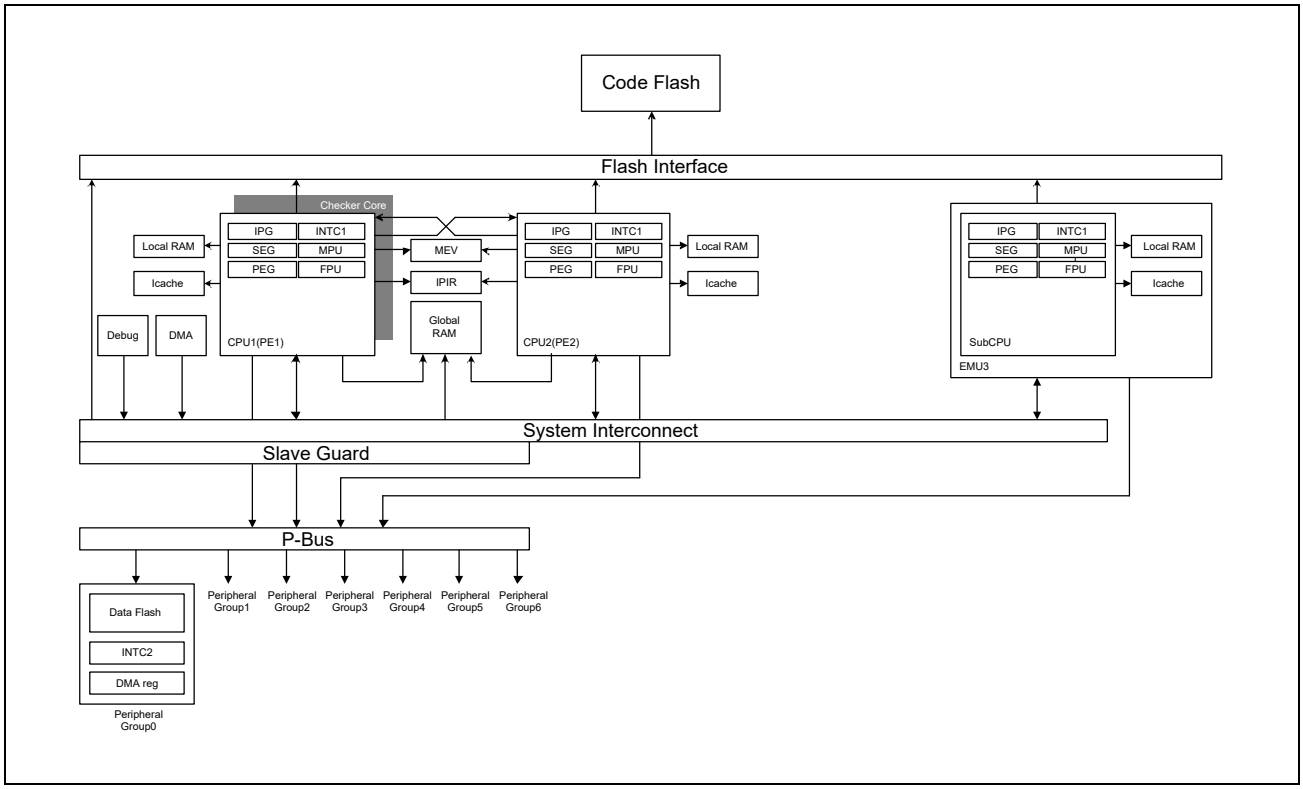


Figure 3.1 Block Diagram of RH850/C1M-A1 and C1M-A2

CPU1 (PE1)

The RH850 G3MH core is included as a main CPU. CPU1 also includes the checker core for safety assurance.

CPU2 (PE2)

The RH850 G3MH core is included for enhanced performance CPU (RH850/C1M-A2).

SubCPU (PE3) in EMU3

The G3MH core is included as the SubCPU (PE3) in EMU3, the motor controlling IP.

Local RAM

Each PE has a RAM to which high speed access is possible.

Global RAM

A large-capacity shared RAM.

Code Flash

A large-capacity flash memory is included for program storage. CPU1, CPU2, and the SubCPU share the code flash and they are connected with each other via the flash interface.

Data Flash

This is a flash memory which can be rewritten by the CPUs.

P-Bus

The P-Bus connects the peripheral IPs. The P-Bus is divided into seven peripheral groups (0 to 6). For details, see **Section 3.1.2, Configuration of Peripheral Groups**.

INTC1, INTC2

INTC1 is an interrupt controller exclusive to each PE. INTC2 is a common interrupt controller that all PEs share, being able to set the binding destination PE of an interrupt request by the registers.

DMA

Two DMA transfer modules, DMAC and DTS, are included.

Slave Guard

The slave guard is a function to prevent unauthorized access from the specific bus master, consisting of the following guard functions:

(1) PE guard (PEG)
The PE guard is a function to prevent unauthorized access to the resources in the PE from the bus master other than the PE itself. Access from the PE itself is only enabled but all other accesses are disabled after release from the reset state. For details, see **Section 3.2.4.1, PE Guard Function (PEG)**.

(2) Global RAM guard (GRG)
The global RAM guard is a function to prevent unauthorized access to the global RAM from the bus master. The global RAM is in the unprotected state (accessible from all bus masters) after release from the reset state. For details, see **29.4.2, GRG (Global RAM Guard)** in **Section 29, Functional Safety**.

(3) Peripheral guard (PBG)
The peripheral guard is a function to prevent unauthorized access to peripheral devices from the bus master. Peripheral devices are in the unprotected state (accessible from all bus masters) after release from the reset state. For details, see **29.4.3, PBG** in **Section 29, Functional Safety**.

3.1.2 Configuration of Peripheral Groups

P-Bus is connected to peripheral groups 0 to 6 and CPUs. The individual modules belong to one of the peripheral groups as shown in the table below. For module names, see the corresponding list of registers in each chapter.

Table 3.1 Peripheral Group Configuration (1/4)

Peripheral Group	Module Name*1
CPU	INTC1
	IPIRSS
	IPG
	MEV
	PEG
	SEG
	TESTCOMP
CPU (DEBUG)	AUDR
0	APDP[INTC2]
	APDP[PDMA]
	DMASS
	ECCCPU1
	ECCCPU2
	ECCEEP
	ECCEEPC
	ECCFLI
	ECCGRAM
	ECCIC1
	ECCIC2
	FACI[FCUFAREA*2]
	FLASH[FRDCYCLD]
	INTC2
	MGDGR
	MISG (CPU1, CPU2)
	PBG[PBG0, PBG1]
	ICUSE
	DataFlash

Table 3.1 Peripheral Group Configuration (2/4)

Peripheral Group	Module Name*1
1	OSTMn (n = 0, 2)
	WDTAn (n = 0)
	SWDTAn (n = 0)
	SCI3n (n = 0, 2)
	RSENTn (n = 0, 2)
	APDP[RSENTn (n = 0, 2)]
	PBG[PBG2]
	TAUDn (n = 0, 2, 3)
	TAUJn (n = 0)
	TSG3n (n = 0, 2)
	TAPAn (n = 0, 2, 3, 5)
	TPBAn (n = 0)
	ENCAn (n = 0)
	PIC1Bn (n = 0)
	PIC2D
	PBG[PBG4]
2	CSIHn (n = 0, 1, 2)
	APDP[CSIHn (n = 0, 1, 2)]
	E7RC1M/C
	E7RC2M/C
	E7CS0M/C
	E7CS1M/C
	E7CS2M/C
	INTIF
	APDP[INFTIF]
	RS-CANFD
	DMA/DTSTRGSEL
	EMU3*3
	ECCCPU3
	ECCIC3
	MISG (SubCPU)
	RDC3An (n = 0, 1)
	APDP[RDC3An (n = 0, 1)]
	PBG[PBG6]
	PBG[PBG7]
	PBG[PBG8]

Table 3.1 Peripheral Group Configuration (3/4)

Peripheral Group	Module Name*1
3	ADCCn (n = 0, 2)
	APDP[ADCCn (n = 0, 2)]
	ADPA
	DCRAn (n = 0)
	DNFn (n = 0 to 8)
	APDP[DNFn (n = 0 to 8)]
	RLIN3n (n = 0, 2)
	ECM
	ECM[ECMC]
	ECM[ECMM]
	EINT
	APDP[EINT]
	EMUSINT
	APDP[EMUEINT]
	FACI[other than FCUFAREA]
	FLASH[SELFID, SELFIDST)]
	FLASH[OPBT, PRDNAME]
	PORT
	APDP[PORT]
	PBG[PBG9]
	PBG[PBG10]
	PBG[PBG11]
4	OSTMn (n = 1, 3)
	WDTAn (n = 1)
	SWDTn (n = 1)
	SCI3n (n = 1)
	RSENTn (n = 1, 3)
	APDP[RSENTn (n = 1, 3)]
	PBG[PBG3]
	TAUDn (n = 1)
	TAUJn (n = 1)
	TSG3n (n = 1)
	TAPAn (n = 1, 4)
	TPBAn (n = 1)
	ENCAn (n = 1)
	PIC1B (n = 1)
	PBG[PBG5]

Table 3.1 Peripheral Group Configuration (4/4)

Peripheral Group	Module Name*1
5	CLMAC
	CLMAn (n = 0 to 3)
	FLASH[FHVE15]
	FLASH[FHVE3]
	SYS
	PBG[PBG13]
6	ADCCn (n = 1)
	APDP[ADCCn (n = 1)]
	DCRAn (n = 1)
	RLIN3n (n = 1)
	PBG[PBG12]

Note 1. The applicable module or register is indicated with [ ].

Note 2. Refer to *RH850/C1M-A Flash Memory User's Manual: Hardware Interface*.

Note 3. Access to EMU3 from the SubCPU is not via peripheral group 2 but through a dedicated path.

3.2 CPU

3.2.1 Core Functions

3.2.1.1 Features

Table 3.2 lists features of the RH850G3MH core.

Table 3.2 Features of the RH850G3MH Core

Item	Feature
CPU	● Advanced 32-bit architecture for embedded control ● 32-bit internal data bus ● Thirty-two 32-bit general registers ● RISC-type instruction sets – Long-/short- format load/store instructions – Three-operand instructions – Instruction sets based on C language ● CPU operating modes – User mode and supervisor mode ● Address space: 4-Gbyte linear address space for both data and instructions ● Instructions: A snooze instruction (SNOOZE) is included for temporary suspension by switching the CPU clock signal (CLK_CPU*1) off for 32 clock cycles.
Coprocessor	● A floating-point operation coprocessor (FPU) mounted – Supports single precision (32 bits) and double precision (64 bits). – Supports data types and exceptions conforming to IEEE754. – Rounding mode: Neighborhood, 0 direction, + ∞ direction, and – ∞ direction – Handling subnormal numbers: Rounding down to 0 or exception notification to conform to IEEE754
Exception/Interrupt	● 16 interrupt priority levels settable for each channel ● Vector selection method selectable according to performance request or memory usage – Direct branching exception vectors – Indirect branching exception vectors referring to the address table ● Supports the high-speed save/return processing of the context by the dedicated instructions (PUSHSP and POPSP) at the generation of an interrupt
Memory Management	● Memory protection function (MPU): 16 areas settable
Cache	● Instruction cache

Note 1. CLK_EMU_H for the SubCPU of EMU3.

3.2.1.2 Register Set

The program registers and system registers are explained in this section.

(1) Program registers

Program registers include the general-purpose registers (r0 to r31) and a program counter (PC). The general-purpose register r0 always retains 0, but the values of the registers r1 to r31 after reset are undefined.

Table 3.3 List of Program Registers

Program Register	Name	Function	Description
General-purpose registers	r0	Zero register	Always retains “0”.
	r1	Assembler reserved register	Used as working register for generating addresses
	r2	Register for address and data variables (this register is used when the real-time OS in use does not use this register)	
	r3	Stack pointer (SP)	Used for generating a stack frame when a function is called
	r4	Global pointer (GP)	Used for accessing a global variable in the data area
	r5	Text pointer (TP)	Used as a register that indicates the start of the text area (area where program code is placed).
	r6-29	Register for addresses and data variables	
	r30	Element pointer (EP)	Used as a base pointer for generating addresses when accessing memory
	r31	Link pointer (LP)	Used when the compiler calls a function
Program counter	PC	Retains instruction addresses during execution of programs	

Note: For further descriptions of r1, r3 to r5, and r31 used for an assembler and/or C compiler, see the manual of each software development environment.

(a) General-purpose registers

A total of 32 general-purpose registers (r0 to r31) are provided. All of these registers can be used for either data variables or address variables. The general-purpose register r0 always retains 0, but the values of the registers r1 to r31 after reset are undefined.

Of the general-purpose registers, r0 to r5, r30, and r31 are assumed to be used for special purposes in software development environments, so it is necessary to note the following when using them.

1. r0, r3, and r30
   These registers are implicitly used by instructions.

   r0 is a register that always retains "0". It is used for operations that use 0 and addressing with base address being 0.

   r3 is implicitly used by the PREPARE, DISPOSE, PUSHSP, and POPSP instructions.

   r30 is used as a base pointer when the SLD or SST instruction accesses memory

2. r1, r4, r5, and r31
   These registers are implicitly used by the assembler and C compiler.

   When using these registers, register contents must first be saved so they are not lost and can be restored after the registers are used.

3. r2
   This register might be used by a real-time OS in some cases. If the real-time OS that is being used is not using r2, r2 can be used as a register for address variables or data variables.

(b) PC — Program counter

The PC retains the address of the instruction being executed.

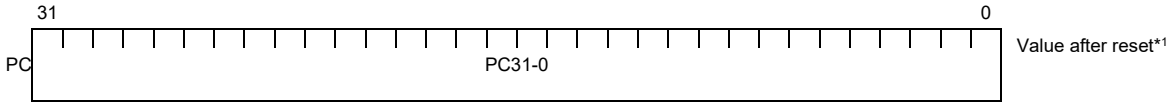


Table 3.4 PC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	PC31-1	The address of the instruction being executed.	R/W	*1
0	PC0	Fixed to 0. Branching to an odd number is disabled.	R/W	0

Note 1. The value after reset differs depending on the startup area. For details, see **Section 4, Address Space**.

(2) Basis System Registers

The basic system registers are used to control CPU status and to retain exception information.

System registers are read from or written to by using the LDSR or STSR instruction and specifying the system register number, which is made up of a register number and a selection ID.

Table 3.5 Basic System Registers

Register Number (reg.ID, sel.ID)	Name	Function	Access Permission
SR0, 0	EIPC	Status save registers when acknowledging EI level exception	SV
SR1, 0	EIPSW	Status save registers when acknowledging EI level exception	SV
SR2, 0	FEPC	Status save registers when acknowledging FE level exception	SV
SR3, 0	FEPSW	Status save registers when acknowledging FE level exception	SV
SR5, 0	PSW	Program status word	*1
SR6, 0	FPSR	(See **(4) FPU function register**)	CU0 and SV
SR7, 0	FPEPC	(See **(4) FPU function register**)	CU0 and SV
SR8, 0	FPST	(See **(4) FPU function register**)	CU0
SR9, 0	FPCC	(See **(4) FPU function register**)	CU0
SR10, 0	FPCFG	(See **(4) FPU function register**)	CU0
SR13, 0	EIIC	EI level exception cause	SV
SR14, 0	FEIC	FE level exception cause	SV
SR16, 0	CTPC	CALLT execution status save register	UM
SR17, 0	CTPSW	CALLT execution status save register	UM
SR20, 0	CTBP	CALLT base pointer	UM
SR28, 0	EIWR	EI level exception working register	SV
SR29, 0	FEWR	FE level exception working register	SV
SR0, 1	MCFG0	Machine configuration	SV
SR2, 1	RBASE	Reset vector address	SV
SR3, 1	EBASE	Exception handler vector address	SV
SR4, 1	INTBP	Base addresses of the interrupt handler address table	SV
SR5, 1	MCTL	CPU control	SV
SR6, 1	PID	Processor ID	SV
SR11, 1	SCCFG	SYSCALL operation setting	SV
SR12, 1	SCBP	SYSCALL base pointer	SV
SR0, 2	HTCFG0	Thread configuration	SV
SR6, 2	MEA	Memory error address	SV
SR7, 2	ASID	Address space ID	SV
SR8, 2	MEI	Memory error information	SV

Note 1. The access permission differs depending on the bit. For details, see **(e) PSW — Program status word**.

(a) EIPC ─ Status save register when acknowledging EI level exception

When an EI level exception is acknowledged, the address of the instruction that was being executed when the EI level exception occurred, or of the next instruction, is saved to the EIPC register (refer to *section 4.1.3, Types of Exceptions in the RH850G3MH User's Manual: Software*).

Because there is only one pair of EI level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

Be sure to set an even-numbered address to the EIPC register. An odd-numbered address must not be specified.

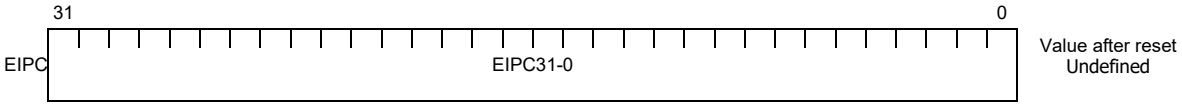


Table 3.6 EIPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	EIPC31-1	These bits indicate the PC saved when an EI level exception is acknowledged.	R/W	Undefined
0	EIPC0	This bit indicates the PC saved when an EI level exception is acknowledged. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the EIRET instruction is executed is 0.	R/W	Undefined

(b) EIPSW ─ Status save register when acknowledging EI level exception

When an EI level exception is acknowledged, the current PSW setting is saved to the EIPSW register.

Because there is only one pair of EI level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

CAUTION

Bits 11 to 9 are related to the debug function and therefore cannot normally be changed.

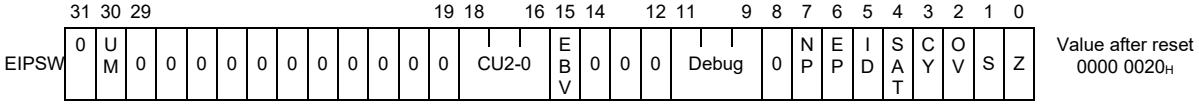


Table 3.7 EIPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	Reserved	This bit is read as 0. The write value should be 0.	R	0
30	UM	This bit saves the PSW.UM bit when an EI level exception is acknowledged.	R/W	0
29 to 19	Reserved	These bits are read as 0. The write value should be 0.	R	0
18 to 16	CU2-0	These bits save the PSW.CU2-0 field setting when an EI level exception is acknowledged. (CU2 and CU1 are reserved for future expansion. Be sure to clear them to 0.)	R/W	0
15	EBV	This bit saves the PSW.EBV bit setting when an EI level exception is acknowledged.	R/W	0
14 to 12	Reserved	These bits are read as 0. The write value should be 0.	R	0
11 to 9	Debug	These bits save the PSW.Debug field setting when an EI level exception is acknowledged.	R/W	0
8	Reserved	This bit is read as 0. The write value should be 0.	R	0
7	NP	These bits save the PSW.NP bit setting when an EI level exception is acknowledged.	R/W	0
6	EP	These bits save the PSW.EP bit setting when an EI level exception is acknowledged.	R/W	0
5	ID	These bits save the PSW.ID bit setting when an EI level exception is acknowledged.	R/W	1
4	SAT	These bits save the PSW.SAT bit setting when an EI level exception is acknowledged.	R/W	0
3	CY	These bits save the PSW.CY bit setting when an EI level exception is acknowledged.	R/W	0
2	OV	These bits save the PSW.OV bit setting when an EI level exception is acknowledged.	R/W	0
1	S	These bits save the PSW.S bit setting when an EI level exception is acknowledged.	R/W	0
0	Z	These bits save the PSW.Z bit setting when an EI level exception is acknowledged.	R/W	0

(c) FEPC ─ Status save register when acknowledging FE level exception

When an FE level exception is acknowledged, the address of the instruction that was being executed when the FE level exception occurred, or of the next instruction, is saved to the FEPC register (refer to *section 4.1.3, Types of Exceptions in the RH850G3MH User's Manual: Software*).

Because there is only one pair of FE level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program. Be sure to set an even-numbered address to the FEPC register. An odd-numbered address must not be specified.

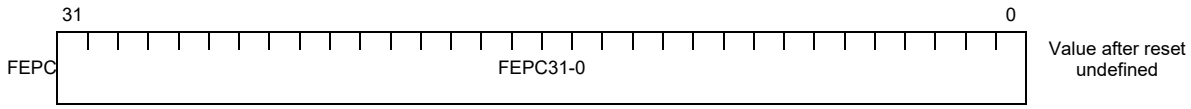


Table 3.8 FEPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	FEPC31-1	These bits indicate the PC saved when an FE level exception is acknowledged.	R/W	Undefined
0	FEPC0	This bit indicates the PC saved when an FE level exception is acknowledged. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the FERET instruction is executed is 0.	R/W	Undefined

(d) FEPSW ─ Status save register when acknowledging FE level exception

When an FE level exception is acknowledged, the current PSW setting is saved to the FEPSW register. Because there is only one pair of FE level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

CAUTION

Bits 11 to 9 are related to the debug function and therefore cannot normally be changed.

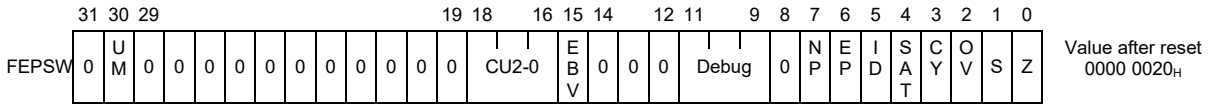


Table 3.9 FEPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	Reserved	This bit is read as 0. The write value should be 0.	R	0
30	UM	This bit saves the PSW.UM bit setting when an FE level exception is acknowledged.	R/W	0
29 to 19	Reserved	These bits are read as 0. The write value should be 0.	R	0
18 to 16	CU2-0	These bits save the PSW.CU2-0 field setting when an FE level exception is acknowledged. (CU2 and CU1 are reserved for future expansion. Be sure to clear them to 0.)	R/W	0
15	EBV	PSW.EBV bit setting when an FE level exception is acknowledged.	R/W	0
14 to 12	Reserved	These bits are read as 0. The write value should be 0.	R	0
11 to 9	Debug	These bits save the PSW.Debug bit setting when an FE level exception is acknowledged.	R/W	0
8	Reserved	This bit is read as 0. The write value should be 0.	R	0
7	NP	This bit saves the PSW.NP bit setting when an FE level exception is acknowledged.	R/W	0
6	EP	This bit saves the PSW.EP bit setting when an FE level exception is acknowledged.	R/W	0
5	ID	This bit saves the PSW.ID bit setting when an FE level exception is acknowledged.	R/W	1
4	SAT	This bit saves the PSW.SAT bit setting when an FE level exception is acknowledged.	R/W	0
3	CY	This bit saves the PSW.CY bit setting when an FE level exception is acknowledged.	R/W	0
2	OV	This bit saves the PSW.OV bit setting when an FE level exception is acknowledged.	R/W	0
1	S	This bit saves the PSW.S bit setting when an FE level exception is acknowledged.	R/W	0
0	Z	This bit saves the PSW.Z bit setting when an FE level exception is acknowledged.	R/W	0

(e) PSW ─ Program status word

PSW (program status word) is a set of flags that indicate the program status (instruction execution result) and bits that indicate the operation status of the CPU (flags are bits in the PSW that are referenced by a condition instruction (Bcond, CMOV, etc.)).

CAUTIONS

1. When the LDSR instruction is used to change the contents of bits 7 to 0 in this register, the changed contents become valid from the instruction following the LDSR instruction.
2. The access permission for the PSW register differs depending on the bit. All bits can be read, but some bits can only be written under certain conditions. See **Table 3.10 Access Permission for PSW Register** for the access permission for each bit.

Table 3.10 Access Permission for PSW Register

Bit		Access Permission when Reading	Access Permission when Writing
30	UM	UM	SV*1
18 to 16	CU2-0	UM	SV*1
15	EBV	UM	SV*1
11 to 9	Debug	UM	Special*1
7	NP	UM	SV*1
6	EP	UM	SV*1
5	ID	UM	SV*1
4	SAT	UM	UM
3	CY	UM	UM
2	OV	UM	UM
1	S	UM	UM
0	Z	UM	UM

Note 1. The access permission for the whole PSW register is UM, so the PIE exception does not occur even if the register is written by using an LDSR instruction when PSW.UM is 1. In this case, writing is ignored.

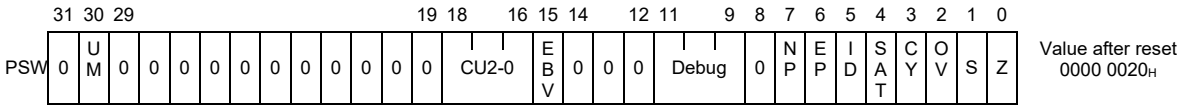


Table 3.11 PSW Register Contents (1/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
31	Reserved	This bit is read as 0. The write value should be 0.	R	0
30	UM	This bit indicates that the CPU is in the user mode (the UM mode). 0: Supervisor mode 1: User mode	R/W	0
29 to 19	Reserved	These bits are read as 0. The write value should be 0.	R	0
18 to 16	CU2-CU0	These bits indicate the coprocessor use permissions. When the bit corresponding to the coprocessor is 0, a coprocessor unusable exception occurs if an instruction for the coprocessor is executed or a coprocessor resource (system register) is accessed. CU2 bit 18: Reserved for future expansion. Be sure to clear to 0. CU1 bit 17: Reserved for future expansion. Be sure to clear to 0. CU0 bit 16: FPU	R/W	000
15	EBV	This bit indicates the reset vector and exception vector operation. See the descriptions of RBASE and EBASE registers for details.	R/W	0
14 to 12	Reserved	These bits are read as 0. The write value should be 0.	R	0
11 to 9	Debug	These bits are used for the debug function for the development tool. Always set these bits to 0.	—	0
8	Reserved	These bits are read as 0. The write value should be 0.	R	0
7	NP	This bit disables acknowledgment of FE level exception. When an FE level exception is acknowledged, this bit is set to 1, which disables acknowledgment of EI and FE level exceptions. For the exceptions of which acknowledgement is disabled by the NP bit, refer to *Table 4.1, Exception Cause List* of *RH850G3MH User's Manual: Software.* 0: Acknowledgement of FE level exception is enabled. 1: Acknowledgement of FE level exception is disabled.	R/W	0
6	EP	This bit indicates that an exception other than an interrupt is being serviced. It is set to 1 when the corresponding exception occurs. This bit does not affect acknowledging an exception request even when it is set to 1. 0: Handling of an exception other than an interrupt is not in progress. 1: Handling of an exception other than an interrupt is in progress.	R/W	0
5	ID	This bit disables acknowledgement of EI level exception. When an EI or FE level exception is acknowledged, this bit is set to 1, which disables acknowledgement of EI level exceptions. For the exceptions of which acknowledgement is disabled by the ID bit, refer to *Table 4.1, Exception Cause List* of *RH850G3MH User's Manual: Software.* This bit is also used to disable EI level exceptions from being acknowledged as a critical section while an ordinary program or interrupt is being serviced. It is set to 1 when the DI instruction is executed, and cleared to 0 when the EI instruction is executed. The change made in the ID bit by the EI or DI instruction is reflected from the next instruction. 0: Acknowledgement of EI level exception is enabled. 1: Acknowledgement of EI level exception is disabled.	R/W	1

Table 3.11 PSW Register Contents (2/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
4	SAT*1	This bit indicates that a saturation arithmetic operation instruction resulted in overflow and saturation processing is applied to the result. This is a cumulative flag, that is, it is set (1) once a saturation occurs and not cleared (0) by subsequent instructions with unsaturated results. This bit is cleared by the LDSR instruction. Note that execution of an arithmetic operation instruction neither set nor clear this flag. 0: The result was not saturated 1: The result was saturated	R/W	0
3	CY	This bit indicates whether a carry or borrow has occurred in the operation result. 0: Carry and borrow have not occurred. 1: Carry or borrow has occurred.	R/W	0
2	OV*1	This bit indicates whether or not an overflow has occurred during an operation. 0: Overflow has not occurred. 1: Overflow has occurred.	R/W	0
1	S*1	This bit indicates whether or not the result of an operation is negative. 0: Result of operation is positive or 0. 1: Result of operation is negative.	R/W	0
0	Z	This bit indicates whether or not the result of an operation is 0. 0: Result of operation is not 0. 1: Result of operation is 0.	R/W	0

Note 1. Saturation processing is applied to the operation result in accordance with the contents of the OV and S flags. The SAT flag is set (1) only when the OV flag is set (1) in the saturation arithmetic operation.

Operation Result Status	Flag Status			Operation Result after Saturation Processing
	SAT	OV	S	
Exceeded positive maximum value	1	1	0	7FFF FFFF$_H$
Exceeded negative maximum value	1	1	1	8000 0000$_H$
Positive (maximum value not exceeded)	Value prior to operation is retained.	0	0	Operation result as is
Negative (maximum value not exceeded)			1	

(f) EIIC ─ EI level exception cause

The EIIC register retains the cause of any EI level exception that occurs. The value retained in this register is an exception code corresponding to a specific exception cause.

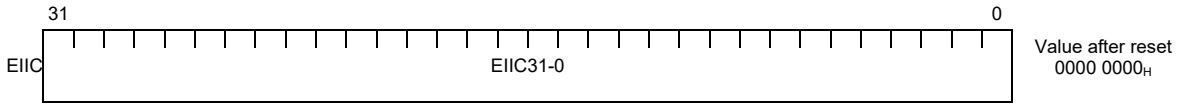


Table 3.12 EIIC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	EIIC31-0	These bits hold the exception cause code when an EI level exception occurs. For the exception cause code to be stored, refer to the exception cause list of **Table 6.11, Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2)**, **Table6.12 Interrupt Exception Handlers and Orders of Priority (SubCPU in EMU3)**, and *RH850G3MH User's Manual: Software*. Detailed exception cause codes defined individually for each exception are stored in the EIIC31-16 field. If there is no particular definition, these bits are set to 0.	R/W	0

(g) FEIC ─ FE level exception cause

The FEIC register retains the cause of any FE level exception that occurs. The value retained in this register is an exception code corresponding to a specific exception cause.

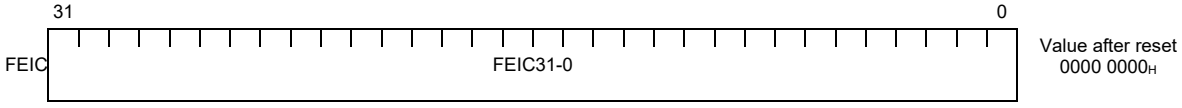


Table 3.13 FEIC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	FEIC31-0	These bits hold the exception cause code when an FE level exception occurs. For the exception cause code to be stored, refer to the exception cause list of **Table 6.11, Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2)** and *RH850G3MH User's Manual: Software*. The FEIC31-16 field stores detailed exception cause codes defined individually for each exception. If there is no particular definition, these bits are set to 0.	R/W	0

(h) CTPC ─ Status save register when executing CALLT

When a CALLT instruction is executed, the address of the next instruction after the CALLT instruction is saved to CTPC. Be sure to set an even-numbered address to the CTPC register. An odd-numbered address must not be specified.

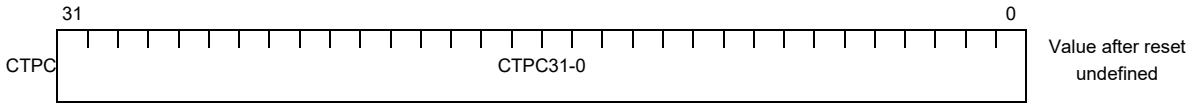


Table 3.14 CTPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	CTPC31-1	These bits indicate the PC of the instruction after the CALLT instruction.	R/W	Undefined
0	CTPC0	This bit indicates the PC of the instruction after the CALLT instruction. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the CTRET instruction is executed is 0.	R/W	Undefined

(i) CTPSW ─ Status save register when executing CALLT

When a CALLT instruction is executed, some of the PSW (program status word) settings are saved to CTPSW.

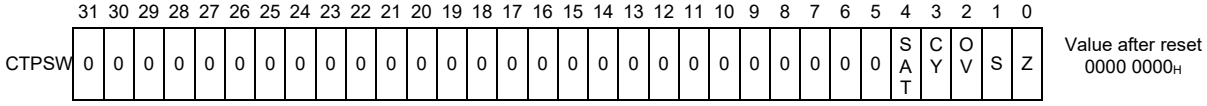


Table 3.15 CTPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	Reserved	These bits are read as 0. The write value should be 0.	R	0
4	SAT	This bit stores the PSW.SAT bit setting when the CALLT instruction is executed.	R/W	0
3	CY	This bit stores the PSW.CY bit setting when the CALLT instruction is executed.	R/W	0
2	OV	This bit stores the PSW.OV bit setting when the CALLT instruction is executed.	R/W	0
1	S	This bit stores the PSW.S bit setting when the CALLT instruction is executed.	R/W	0
0	Z	This bit stores the PSW.Z bit setting when the CALLT instruction is executed.	R/W	0

(j) CTBP ─ CALLT base pointer

The CTBP register is used to specify table addresses of the CALLT instruction and generate target addresses. Be sure to set the CTBP register to a halfword address.

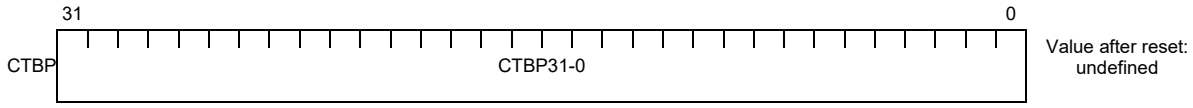


Table 3.16 CTBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	CTBP31-1	These bits indicate the base pointer address of the CALLT instruction. These bits indicate the start address of the table used by the CALLT instruction.	R/W	Undefined
0	CTBP0	This bit indicates the base pointer address of the CALLT instruction. This bit indicates the start address of the table used by the CALLT instruction. Always set this bit to 0.	R	0

(k) ASID ─ Address space ID

This is the address space ID. This is used to identify the address space provided by the memory management function.

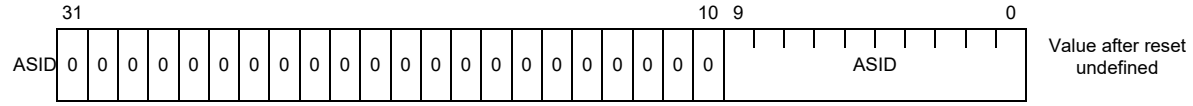


Table 3.17 ASID Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 10	Reserved	These bits are read as 0. The write value should be 0.	R	0
9 to 0	ASID9-0	These bits are the address space ID.	R/W	Undefined

(l) EIWR ─ EI level exception working register

The EIWR register is used as a working register when an EI level exception has occurred.

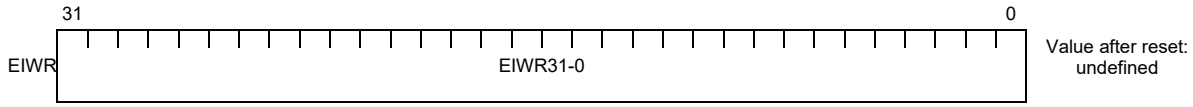


Table 3.18 EIWR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	EIWR31-0	These bits constitute a working register that can be used for any purpose during the processing of an EI level exception. Use this register for purposes such as saving the values of general-purpose registers.	R/W	Undefined

(m) FEWR ─ FE level exception working register

The FEWR register is used as a working register when an FE level exception has occurred.

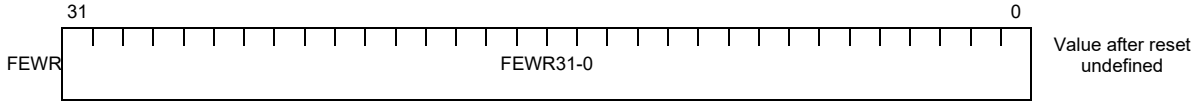


Table 3.19 FEWR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	FEWR31-0	These bits constitute a working register that can be used for any purpose during the processing of an FE level exception. Use this register for purposes such as saving the values of general-purpose registers.	R/W	Undefined

(n) HTCFG0 ─ Thread configuration register

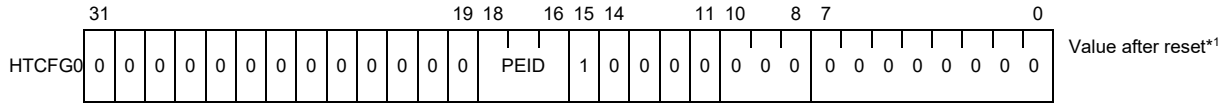


Table 3.20 HTCFG0 Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 19	Reserved	These bits are read as 0. The write value should be 0	R	0
18 to 16	PEID	These bits indicate the processor element number.	R	*2
15	Reserved	These bits are read as 1. The write value should be 1.	R	1
14 to 0	Reserved	These bits are read as 0. The write value should be 0.	R	0

Note 1. This value is 0001 8000_H for CPU1 (PE1), 0002 8000_H for CPU2 (PE2), and 0003 8000_H for the SubCPU (PE3).

Note 2. This value is 001_B for CPU1 (PE1), 010_B for CPU2 (PE2), and 011_B for the SubCPU (PE3).

(o) MEA ─ Memory error address

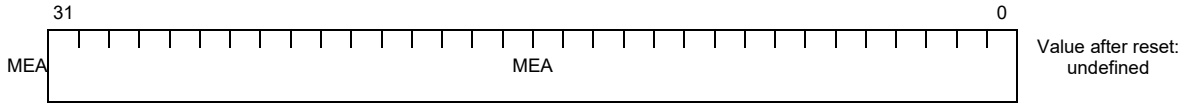


Table 3.21 MEA Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MEA	These bits store the virtual address when an MAE (misaligned) or MPU exception occurs.	R/W	Undefined

(p) MEI ─ Memory error information register

This register is used to store information about the instruction that caused the exception when a misaligned (MAE) or memory protection (MDP) exception occurs.

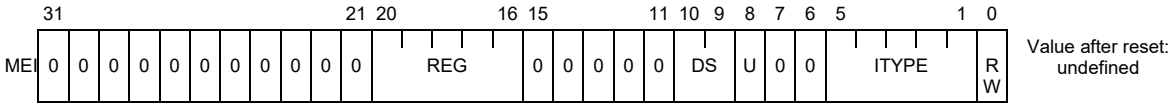


Table 3.22 MEI Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 21	Reserved	These bits are read as 0. The write value should be 0.	R	0
20 to 16	REG	These bits indicate the number of the source or destination register accessed by the instruction that caused the exception. For details, see **Table 3.23 Instructions that Cause Exceptions and Values of MEI Registers**.	R/W	Undefined
15 to 11	Reserved	These bits are read as 0. The write value should be 0.	R	0
10, 9	DS	These bits indicate the type of data handled by the instruction that caused the exception.*1 0: Byte (8 bits) 1: Halfword (16 bits) 2: Word (32 bits) 3: Double-word (64 bits) For details, see **Table 3.23 Instructions that Cause Exceptions and Values of MEI Registers**.	R/W	Undefined
8	U	These bits indicate the sign extension method of the instruction that caused the exception. 0: Signed 1: Unsigned For details, see **Table 3.23 Instructions that Cause Exceptions and Values of MEI Registers**.	R/W	Undefined
7, 6	Reserved	These bits are read as 0. The write value should be 0.	R	0
5 to 1	ITYPE	These bits indicate the instruction that caused the exception. For details, see **Table 3.23 Instructions that Cause Exceptions and Values of MEI Registers**.	R/W	Undefined
0	RW	This bit indicates whether the operation of the instruction that caused the exception was read (Load-memory) or write (Store-memory). 0: Read (Load-memory) 1: Write (Store-memory) For details, see **Table 3.23 Instructions that Cause Exceptions and Values of MEI Registers**.	R/W	Undefined

Note 1. Even if the data is divided and access is made several times due to the specifications of the hardware, the original data type indicated by the instruction is stored.

Table 3.23 Instructions that Cause Exceptions and Values of MEI Registers

Instruction	REG	DS	U	RW	ITYPE
SLD.B	dst	0 (Byte)	0 (Signed)	0 (Read)	00000_B
SLD.BU	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00000_B
SLD.H	dst	1 (Half-word)	0 (Signed)	0 (Read)	00000_B
SLD.HU	dst	1 (Half-word)	1 (Unsigned)	0 (Read)	00000_B
SLD.W	dst	2 (Word)	0 (Signed)	0 (Read)	00000_B
SST.B	src	0 (Byte)	0 (Signed)	1 (Write)	00000_B
SST.H	src	1 (Half-word)	0 (Signed)	1 (Write)	00000_B
SST.W	src	2 (Word)	0 (Signed)	1 (Write)	00000_B
LD.B (disp16)	dst	0 (Byte)	0 (Signed)	0 (Read)	00001_B
LD.BU (disp16)	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00001_B
LD.H (disp16)	dst	1 (Half-word)	0 (Signed)	0 (Read)	00001_B
LD.HU (disp16)	dst	1 (Half-word)	1 (Unsigned)	0 (Read)	00001_B
LD.W (disp16)	dst	2 (Word)	0 (Signed)	0 (Read)	00001_B
ST.B (disp16)	src	0 (Byte)	0 (Signed)	1 (Write)	00001_B
ST.H (disp16)	src	1 (Half-word)	0 (Signed)	1 (Write)	00001_B
ST.W (disp16)	src	2 (Word)	0 (Signed)	1 (Write)	00001_B
LD.B (disp23)	dst	0 (Byte)	0 (Signed)	0 (Read)	00010_B
LD.BU (disp23)	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00010_B
LD.H (disp23)	dst	1 (Half-word)	0 (Signed)	0 (Read)	00010_B
LD.HU (disp23)	dst	1 (Half-word)	1 (Unsigned)	0 (Read)	00010_B
LD.W (disp23)	dst	2 (Word)	0 (Signed)	0 (Read)	00010_B
ST.B (disp23)	src	0 (Byte)	0 (Signed)	1 (Write)	00010_B
ST.H (disp23)	src	1 (Half-word)	0 (Signed)	1 (Write)	00010_B
ST.W (disp23)	src	2 (Word)	0 (Signed)	1 (Write)	00010_B
LD.DW (disp23)	dst	3 (Double-word)	0 (Signed)	0 (Read)	00010_B
ST.DW (disp23)	src	3 (Double-word)	0 (Signed)	1 (Write)	00010_B
LDL.W	dst	2 (Word)	0 (Signed)	0 (Read)	00111_B
STC.W	src	2 (Word)	0 (Signed)	1 (Write)	00111_B
CAXI	dst	2 (Word)	0 (Signed)*1	0 (Read)*2	01000_B
SET1	—	0 (Byte)	0 (Signed)*1	0 (Read)*2	01001_B
CLR1	—	0 (Byte)	0 (Signed)*1	0 (Read)*2	01001_B
NOT1	—	0 (Byte)	0 (Signed)*1	0 (Read)*2	01001_B
TST1	—	0 (Byte)	0 (Signed)*1	0 (Read)	01001_B
PREPARE	src*1	2 (Word)	0 (Signed)*1	1 (Write)	01100_B
DISPOSE	dst*1	2 (Word)	0 (Signed)*1	0 (Read)	01100_B
PUSHSP	src*1	2 (Word)	0 (Signed)*1	1 (Write)	01101_B
POPSP	dst*1, *3	2 (Word)	0 (Signed)*1	0 (Read)	01101_B
SWITCH	—	1 (Half-word)	0 (Signed)	0 (Read)	10000_B
CALLT	—	1 (Half-word)	1 (Unsigned)	0 (Read)	10001_B
SYSCALL	—	2 (Word)	0 (Signed)*1	0 (Read)	10010_B
CACHE	—	—	—	0 (Read)	10100_B
Interrupt (table reference method)*4	—	2 (Word)	0 (Signed)*1	0 (Read)	10101_B

Note 1. This value is different from that of G3M.

Note 2. This exception occurs when the instruction executes a read access.

Note 3. If the instruction is for general-purpose register r3, 0 is stored.

Note 4. When the interrupt vector of the table reference method is read.

NOTE

dst: destination register number, src: source register number

(q) RBASE ─ Reset vector base address

This register indicates the reset vector address when there is a reset. If the PSW.EBV bit is 0, this vector address is also used as the exception vector address.

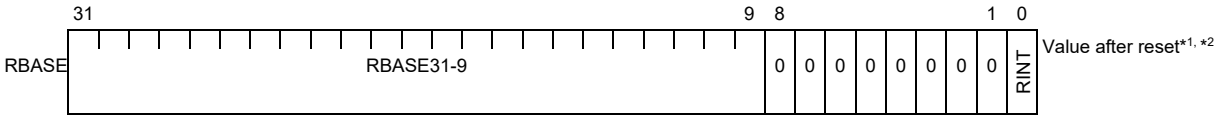


Table 3.24 RBASE Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	RBASE31-9	These bits indicate the reset vector when there is a reset. When PSW.EBV = 0, this address is also used as the exception vector. The RBASE8-0 bits are implicitly set to 0.	R	*1
8 to 1	Reserved	These bits are read as 0. The write value should be 0.	R	0
0	RINT*2	When the RINT bit is set, the exception handler address for interrupt processing is reduced. Refer to *Section 4.4.1 (1) Direct Vector Method* of *RH850G3MH User's Manual: Software* for details. This bit is valid when PSW.EBV = 0.	R	*2

Note 1. The value after reset depends on the startup area. For details, see **Section 4, Address Space**.

Note 2. This is 0_B for these products.

(r) EBASE ─ Exception handler vector address

This register indicates the exception handler vector address. This register is valid when the PSW.EBV bit is 1.

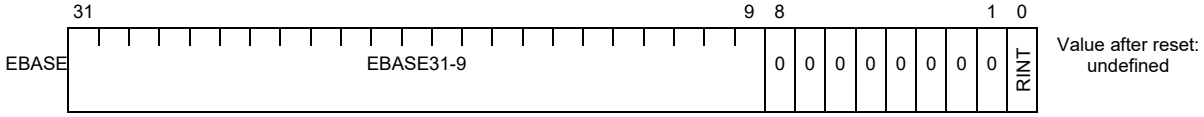


Table 3.25 EBASE Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	EBASE31-9	The exception handler routine address is changed to the address resulting from adding the offset address of each exception to the base address specified for this register. The EBASE8-0 bits are are implicitly set to 0.	R/W	Undefined
8 to 1	Reserved	These bits are read as 0. The write value should be 0.	R	0
0	RINT	While this bit is set, the exception handler address for interrupt processing is reduced. Refer to *Section 4.4.1 (1) Direct Vector Method* of *RH850G3MH User's Manual: Software* for details.	R/W	Undefined

(s) INTBP ─ Base address of the interrupt handler table

This register indicates the base address of the table when the table reference method is selected as the interrupt handler address selection method.

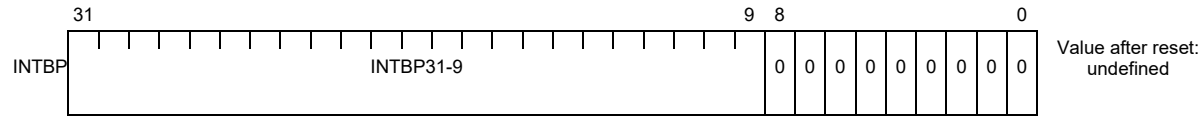


Table 3.26 INTBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	INTBP31-9	These bits indicate the base pointer address for an interrupt when the expanded specification is used. The value indicated by these bits is the first address in the table used to determine the exception handler when the interrupt specified by the expanded specification (EIINT0-383) is acknowledged. The INTBP8-0 bits are are implicitly set to 0.	R/W	Undefined
8 to 0	Reserved	These bits are read as 0. The write value should be 0.	R	0

(t) PID ─ Processor ID

The PID register retains a processor identifier that is unique to the CPU. The PID register is a read-only register.

CAUTION

The PID register indicates information used to identify the incorporated CPU core and CPU core configuration. Usage such that the software behavior varies dynamically according to the PID register information is not assumed.

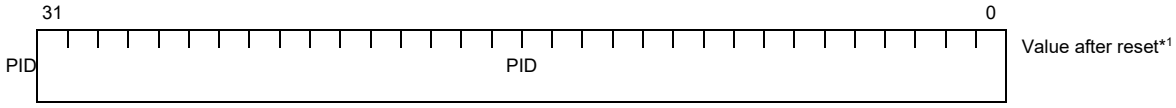


Table 3.27 PID Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 24	PID	Architecture identifier This identifier indicates the architecture of the processor.	R	05_H
23 to 8		Function identifier This identifier indicates the functions of the processor. These bits indicate whether or not functions defined per bit are implemented (1: implemented, 0: not implemented). Bits 23 to 11: Reserved Bit 10 : Double-precision floating-point operation function Bit 9 : Single-precision floating-point operation function Bit 8 : Memory protection unit (MPU) function	R	8007_H
7 to 0		Version identifier This identifier indicates the version of the processor.	R	81_H

Note 1. This value is 0580 0781_H for these products.

(u) SCCFG ─ SYSCALL operation setting

This register is used to set operations related to the SYSCALL instruction. Be sure to set an appropriate value to this register before using the SYSCALL instruction.

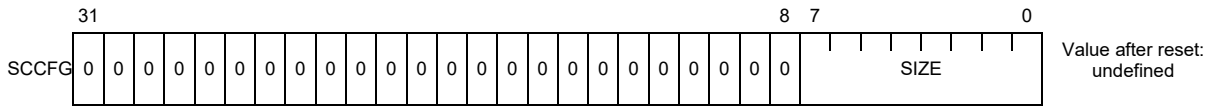


Table 3.28 SCCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 8	Reserved	These bits are read as 0. The write value should be 0.	R	0
7 to 0	SIZE	These bits specify the maximum number of entries of a table that the SYSCALL instruction references. The maximum number of entries the SYSCALL instruction references is 1 if SIZE is 0, and 256 if SIZE is 255. By setting the maximum number of entries appropriately in accordance with the number of functions branched by the SYSCALL instruction, the memory area can be effectively used. If a vector exceeding the maximum number of entries is specified for the SYSCALL instruction, the first entry is selected. Place an error processing routine at the first entry.	R/W	Undefined

(v) SCBP ─ SYSCALL base pointer

The SCBP register is used to specify a table address of the SYSCALL instruction and generate a target address. Be sure to set an appropriate value to this register before using the SYSCALL instruction.

Be sure to set a word address to the SCBP register.

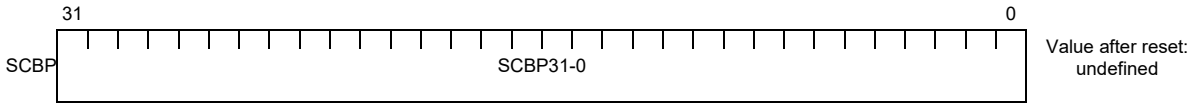


Table 3.29 SCBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	SCBP31-2	These bits indicate the base pointer address of the SYSCALL instruction. These bits indicate the start address of the table used by the SYSCALL instruction.	R/W	Undefined
1, 0	SCBP1-0	These bits indicate the base pointer address of the SYSCALL instruction. These bits indicate the start address of the table used by the SYSCALL instruction. Always set these bits to 0.	R	0

(w) MCFG0 ─ Machine configuration

This register indicates the CPU configuration.

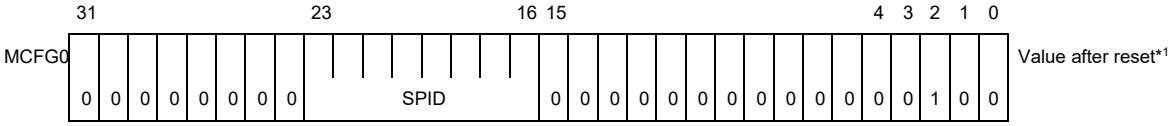


Table 3.30 MCFG0 Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 24	Reserved	These bits are read as 0. The write value should be 0.	R	0
23 to 16	SPID	Bits 23 to 18: Not used for these products. (Reserved for future expansion. Be sure to clear to 0.) Bits 17, 16: These bits indicate the system protection number.	R/W	*2
15 to 3	Reserved	These bits are read as 0. The write value should be 0.	R	0
2	Reserved	These bits are read as 1. The write value should be 1.	R	1
1, 0	Reserved	These bits are read as 0. The write value should be 0.	R	0

Note 1. This value is 0001 0004_H for CPU1 (PE1), 0002 0004_H for CPU2 (PE2), and 0003 0004_H for the SubCPU (PE3) for these products.

Note 2. This value is 01_H for CPU1 (PE1), 02_H for CPU2 (PE2), and 03_H for the SubCPU (PE3) for these products.

(x) MCTL ─ Machine control

This register is used to control the CPU.

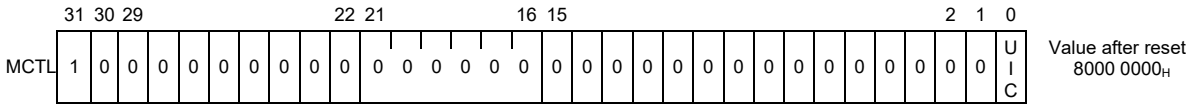


Table 3.31 MCTL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	Reserved	This bit is read as 1. The write value should be 1.	R	1
30 to 1	Reserved	These bits are read as 0. The write value should be 0.	R	0
0	UIC	This bit is used to control the interrupt enable/disable operation in the user mode. When this bit is set to 1, executing the EI/DI instruction become possible.	R/W	0

(3) Interrupt function registers

The interrupt function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.32 Interrupt Function System Registers

Register Number (reg.ID, sel.ID)	Name	Function	Access Permission
SR10, 2	ISPR	Priority of interrupt being serviced	SV
SR11, 2	PMR	Interrupt priority masking	SV
SR12, 2	ICSR	Interrupt control status	SV
SR13, 2	INTCFG	Interrupt function setting	SV

(a) ISPR — Priority of interrupt being serviced register

This register retains the priority of the EIINTn interrupt being serviced. This priority value is then used to perform priority ceiling processing when multiple interrupts are generated.

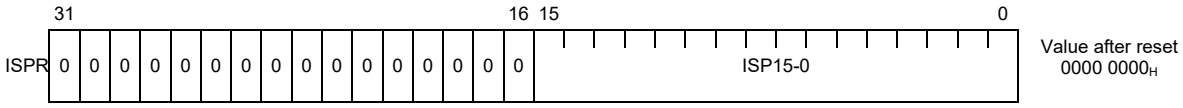


Table 3.33 ISPR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 16	Reserved	These bits are read as 0. The write value should be 0.	R	0
15 to 0	ISP15-0	These bits indicate the acknowledgment status of an EIINTn interrupt with a priority*1 that corresponds to the relevant bit position. 0: An interrupt request for an interrupt whose priority corresponds to the relevant bit position has not been acknowledged. 1: An interrupt request for an interrupt whose priority corresponds to the relevant position is being serviced by the CPU core. The bit positions correspond to the following priority levels: Bit 0 — Priority 0 (highest) Bit 1 — Priority 1 : Bit 14 — Priority 14 Bit 15 — Priority 15 When an interrupt request (EIINTn) is acknowledged, the bit corresponding to the acknowledged interrupt request is automatically set to 1. If PSW.EP is 0 when the EIRET instruction is executed, the bit with the highest priority among the ISP15-0 bits that are set (0 is the highest priority) is cleared to 0*2. While a bit in this register is set to 1, same or lower priority interrupts (EIINTn) are masked. Priority level judgment is therefore not performed when the system is determining whether to acknowledge an exception, meaning that exceptions will not be acknowledged. For details, refer to *Section 4.1.5, Interrupt Exception Priority and Priority Masking* of *RH850G3MH User's Manual: Software*. When performing software-based priority control using the PMR register, be sure to clear this register by using the INTCFG.ISPC bit.	R*3	0

Note 1. For details, refer to *Section 4.1.5, Interrupt Exception Priority and Priority Masking* of *RH850G3MH User's Manual: Software.*

Note 2. Interrupt acknowledgment and auto-updating of values when the EIRET instruction is executed are disabled by setting (1) to the INTCFG.ISPC bit. It is recommended to enable auto-updating of values, so in normal cases, the INTCFG.ISPC bit should be cleared to 0.

Note 3. This is R or R/W, depending on the setting of the INTCFG.ISPC bit. It is recommended to use this register as a read-only (R) register.

(b) PMR — Interrupt priority masking register

This register is used to mask the specified interrupt priority.

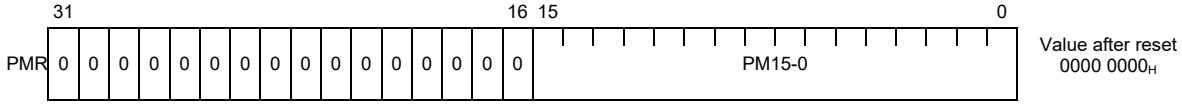


Table 3.34 PMR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 16	Reserved	These bits are read as 0. The write value should be 0.	R	0
15 to 0	PM15-0	These bits mask an interrupt request with a priority level that corresponds to the relevant bit position. 0: Servicing of an interrupt request with a priority that corresponds to the relevant bit position is enabled. 1: Servicing of an interrupt request with a priority that corresponds to the relevant bit position is disabled. The bit positions correspond to the following priority levels: Bit 0 — Priority 0 (highest) Bit 1 — Priority 1 : Bit 14 — Priority 14 Bit 15 — Priority 15 and Priority 16 (lowest) While a bit in this register is set to 1, interrupts (EIINTn) with the priority corresponding to that bit are masked. Priority level judgment is therefore not performed when the system is determining whether to acknowledge an exception, meaning that exceptions will not be acknowledged*1.	R/W	0

Note 1. Specify the masks by setting the bits to 1 in order from the lowest-priority bit. For example, $FF00_H$ can be set, but $F0F0_H$ or $00FF_H$ cannot be set.

(c) ICSR — Interrupt control status register

This register indicates the interrupt control status in the CPU.

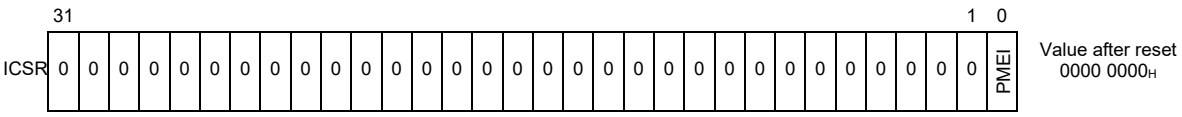


Table 3.35 ICSR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	Reserved	These bits are read as 0. The write value should be 0.	R	0
0	PMEI	This bit indicates that an interrupt (EIINTn) with the priority level masked by the PMR register exists.	R	0

(d) INTCFG ─ Interrupt function setting register

This register is used to specify settings related to the CPU's internal interrupt function.

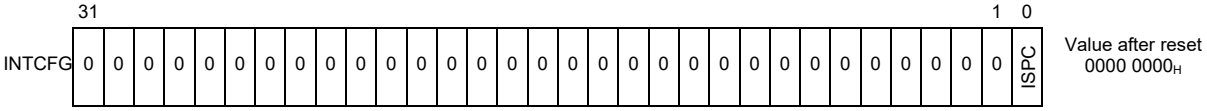


Table 3.36 INTCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	Reserved	These bits are read as 0. The write value should be 0.	R	0
0	ISPC	This bit changes how the ISPR register is written. 0: The ISPR register is automatically updated. Updates triggered by the program (via execution of an LDSR instruction) are ignored. 1: The ISPR register is not automatically updated. Updates triggered by the program (via execution of an LDSR instruction) are performed. If this bit is cleared to 0, the bits of the ISPR register are automatically set to 1 when an interrupt (EIINTn) is acknowledged, and cleared to 0 when the EIRET instruction is executed. In this case, the bits are not updated by an LDSR instruction executed by the program. If this bit is set to 1, the bits of the ISPR register are not updated by the acknowledgement of an interrupt (EIINTn) or by execution of the EIRET instruction. In this case, the bits can be updated by an LDSR instruction executed by the program. In normal cases, the ISPC bit should be cleared. When performing software-based priority control, however, set this bit to 1 and perform priority control by using the PMR register.	R/W	0

(4) FPU function registers

(a) Floating-point registers

The FPU uses the CPU general-purpose registers (r0 to r31). There are no register files used only for floating-point operations.

- Single-precision floating-point instruction:
  Thirty-two 32-bit registers can be specified. These corresponds to the general-purpose registers r0 to r31.
- Double-precision floating-point instruction:
  Sixteen 64-bit registers can be specified. Paired general-purpose registers are used as register pairs ({r1, r0},{r3, r2}•••{r31, r30}). Each register pair is specified in the instruction format with an even numbered register. Because r0 is a zero register (always holds "0"), in principle, {r1, r0} cannot be used by a double-precision floating-point instruction.

(b) Floating-point system registers

The FPU can use the following system registers to control floating-point operations.

The floating-point system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID. For details of these registers, refer to *Section 3.4.2 Floating-Point Function System Registers* of *RH850G3MH User's Manual: Software*.

Table 3.37 List of FPU Function Registers

Register Number (reg.ID, sel.ID)	Name	Function	Access Permission
SR6, 0	FPSR	Configuration and status of floating-point operation	CU0 and SV
SR7, 0	FPEPC	Program counter for floating-point operation exception	CU0 and SV
SR8, 0	FPST	State of a floating-point operation	CU0
SR9, 0	FPCC	Results of comparison of floating-point operations	CU0
SR10, 0	FPCFG	Settings for floating-point operations	CU0

(5) MPU function registers

(a) MPU function system registers

The MPU function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.38 List of MPU Function System Registers (1/2)

Register Number (reg.ID, sel.ID)	Name	Function	Access Permission
SR0, 5	MPM	Memory protection operation mode setting	SV
SR1, 5	MPRC	MPU region control	SV
SR4, 5	MPBRGN	MPU base region number	SV
SR5, 5	MPTRGN	MPU end region number	SV
SR8, 5	MCA	Memory protection setting check address	SV
SR9, 5	MCS	Memory protection setting check size	SV
SR10, 5	MCC	Memory protection setting check command	SV
SR11, 5	MCR	Memory protection setting check result	SV
SR0, 6	MPLA0	The lower limit address of the protection area	SV
SR1, 6	MPUA0	The upper limit address of the protection area	SV
SR2, 6	MPAT0	The attribute of the protection area	SV
SR4, 6	MPLA1	The lower limit address of the protection area	SV
SR5, 6	MPUA1	The upper limit address of the protection area	SV
SR6, 6	MPAT1	The attribute of the protection area	SV
SR8, 6	MPLA2	The lower limit address of the protection area	SV
SR9, 6	MPUA2	The upper limit address of the protection area	SV
SR10, 6	MPAT2	The attribute of the protection area	SV
SR12, 6	MPLA3	The lower limit address of the protection area	SV
SR13, 6	MPUA3	The upper limit address of the protection area	SV
SR14, 6	MPAT3	The attribute of the protection area	SV
SR16, 6	MPLA4	The lower limit address of the protection area	SV
SR17, 6	MPUA4	The upper limit address of the protection area	SV
SR18, 6	MPAT4	The attribute of the protection area	SV
SR20, 6	MPLA5	The lower limit address of the protection area	SV
SR21, 6	MPUA5	The upper limit address of the protection area	SV
SR22, 6	MPAT5	The attribute of the protection area	SV
SR24, 6	MPLA6	The lower limit address of the protection area	SV
SR25, 6	MPUA6	The upper limit address of the protection area	SV
SR26, 6	MPAT6	The attribute of the protection area	SV
SR28, 6	MPLA7	The lower limit address of the protection area	SV
SR29, 6	MPUA7	The upper limit address of the protection area	SV
SR30, 6	MPAT7	The attribute of the protection area	SV
SR0, 7	MPLA8	The lower limit address of the protection area	SV
SR1, 7	MPUA8	The upper limit address of the protection area	SV
SR2, 7	MPAT8	The attribute of the protection area	SV
SR4, 7	MPLA9	The lower limit address of the protection area	SV
SR5, 7	MPUA9	The upper limit address of the protection area	SV
SR6, 7	MPAT9	The attribute of the protection area	SV
SR8, 7	MPLA10	The lower limit address of the protection area	SV

Table 3.38 List of MPU Function System Registers (2/2)

Register Number (reg.ID, sel.ID)	Name	Function	Access Permission
SR9, 7	MPUA10	The upper limit address of the protection area	SV
SR10, 7	MPAT10	The attribute of the protection area	SV
SR12, 7	MPLA11	The lower limit address of the protection area	SV
SR13, 7	MPUA11	The upper limit address of the protection area	SV
SR14, 7	MPAT11	The attribute of the protection area	SV
SR16, 7	MPLA12	The lower limit address of the protection area	SV
SR17, 7	MPUA12	The upper limit address of the protection area	SV
SR18, 7	MPAT12	The attribute of the protection area	SV
SR20, 7	MPLA13	The lower limit address of the protection area	SV
SR21, 7	MPUA13	The upper limit address of the protection area	SV
SR22, 7	MPAT13	The attribute of the protection area	SV
SR24, 7	MPLA14	The lower limit address of the protection area	SV
SR25, 7	MPUA14	The upper limit address of the protection area	SV
SR26, 7	MPAT14	The attribute of the protection area	SV
SR28, 7	MPLA15	The lower limit address of the protection area	SV
SR29, 7	MPUA15	The upper limit address of the protection area	SV
SR30, 7	MPAT15	The attribute of the protection area	SV

(b) MPM ─ Memory protection operation mode register

This register is used to define the basic operating state of the memory protection function.

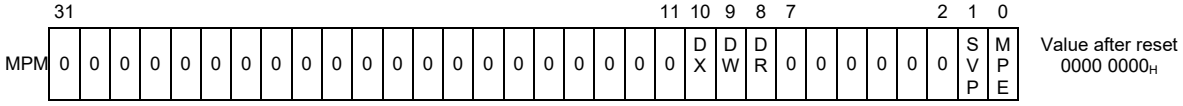


Table 3.39 MPM Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 11	Reserved	These bits are read as 0. The write value should be 0.	R	0
10	DX	This bit is not used in these products. This bit is read as 0. The write value should be 0.	R	0
9	DW	This bit is not used in these products. This bit is read as 0. The write value should be 0.	R	0
8	DR	This bit is not used in these products. This bit is read as 0. The write value should be 0.	R	0
7 to 2	Reserved	These bits are read as 0. The write value should be 0.	R	0
1	SVP	In SV mode (when PSW.UM = 0), this bit is used to specify whether to restrict access according to the SX, SW, and SR bits of the MPAT register for each protection area*1. 0: As usual, implicitly enable all access in SV mode. 1: Restrict access according to the SX, SW, and SR bits even in SV mode*2.	R/W	0
0	MPE	This bit is used to specify whether to enable or disable the MPU function. 0: Disable 1: Enable	R/W	0

Note 1. When the SVP bit is set to 1, access restriction will be given according to the settings for each protection area, even in SV mode. Therefore, specify the protection area before setting the SVP bit in order to prevent restriction of access by the program itself.

Note 2. If access is restricted in SV mode, execution of MDP exceptions or the MIP exception handling itself might not be possible depending on the settings. Be careful to specify settings so that access to the memory area necessary for the exception handler and exception handling is permitted.

(c) MPRC ─ MPU region control

Bits used to perform special memory protection function operations are located in this register.

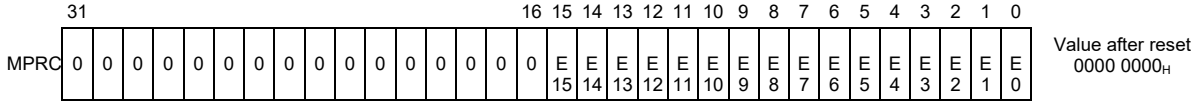


Table 3.40 MPRC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 16	Reserved	These bits are read as 0. The write value should be 0.	R	0
15 to 0	E15-E0	These are the enable bits for each protection area. Bit En is a copy of bit MPATn.E (where n = 15 to 0). These products are provided with sixteen protection areas.	R/W	0

(d) MPBRGN ─ MPU base region register

This register indicates the minimum usable MPU area number.

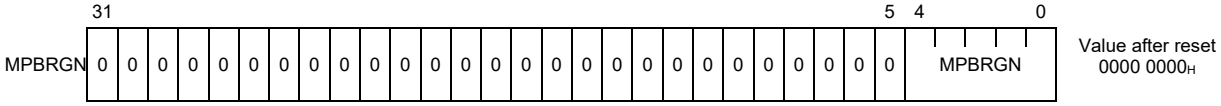


Table 3.41 MPBRGN Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	Reserved	These bits are read as 0. The write value should be 0.	R	0
4 to 0	MPBRGN	These bits indicate the smallest number of an MPU area. These bits are always read as 0.	R	0

(e) MPTRGN — MPU end region register

This register indicates the maximum usable MPU area number + 1.

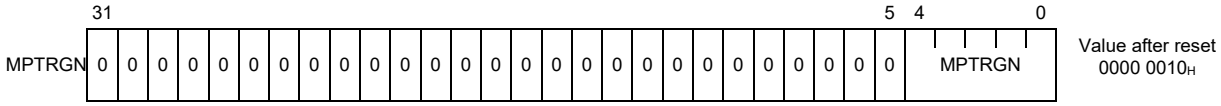


Table 3.42 MPTRGN Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	Reserved	These bits are read as 0. The write value should be 0.	R	0
4 to 0	MPTRGN	These bits indicate the largest number of an MPU area plus one. These bits always indicate the maximum number of MPU areas that the hardware can support . These products are provided with sixteen protection areas.	R	10_H

(f) MCA ─ Memory protection setting check address register

This register is used to specify the base address of the area for which a memory protection setting check is to be performed.

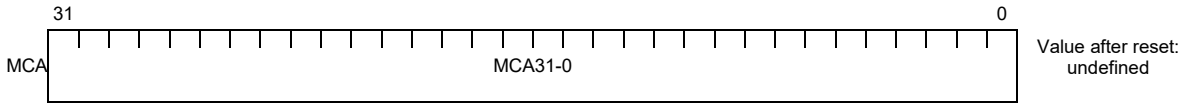


Table 3.43 MCA Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MCA31-MCA0	These bits are used to specify the start address of the memory area subject to a memory protection setting check in bytes.	R/W	Undefined

(g) MCS ─ Memory protection setting check size register

This register is used to specify the size of the area for which a memory protection setting check is to be performed.

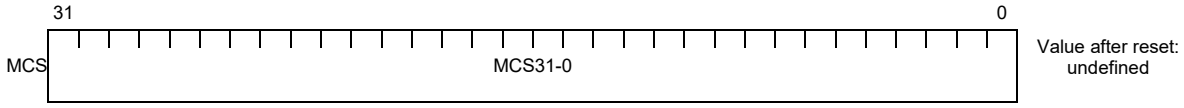


Table 3.44 MCS Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MCS31-MCS0	These bits are used to specify the size of the memory area in bytes which subjects to a memory protection setting check. Because the specified size is assumed to represent an unsigned integer, it is not possible to check an area in the direction in which the address value decreases relative to the MCA register value. Do not specify 0000 0000_H for the MCS register.	R/W	Undefined

(h) MCC ─ Memory protection setting check command register

This command register is used to start a memory protection setting check.

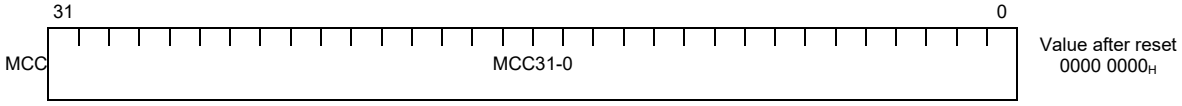


Table 3.45 MCC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MCC31-MCC0	When any value is written to the MCC register, a memory protection setting check starts. Setting the MCA and MCS registers and then writing to this register leads to storage of the result of checking in the MCR register. Since writing any value to this register starts the check, doing so does not require any extra registers when r0 is used as a source register. The result of checking is reflected in MCR according to any area setting regardless of the setting of the PSW.UM bit. The value read from the MCC register is always 0000 0000_H.	R/W	0

(i) MCR ─ Memory protection setting check result register

This register is used to store the results of a memory protection setting check.

Be sure to clear (0) bits 31 to 9, 7 and 6.

CAUTIONS

1. If the specified area to be checked crosses 0000 0000$_H$, it is judged as an area setting error, and the MCR.OV bit is set to 1. This means that the MCR.OV bit must be checked to access the check results. Do not use the check result until it is confirmed that the result is not invalid (OV = 0).
2. If the default operations specified by using the MPM.DX, DW, and DR bits are enabled (1), the correct result might not be able to be obtained. If enabling the specified default operation, do not use the memory protection setting check function.

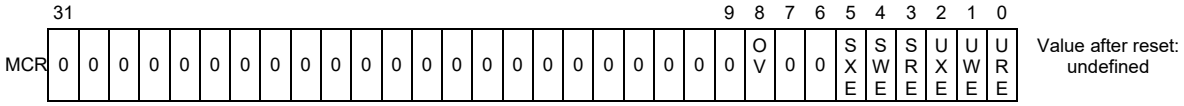


Table 3.46 MCR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	Reserved	These bits are read as 0. The write value should be 0.	R	0
8	OV	If the specified area includes 0000 0000$_H$ or 7FFF FFFF$_H$, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
7, 6	Reserved	These bits are read as 0. The write value should be 0.	R	0
5	SXE	If the specified area is contained within one protection area and execution is permitted for that area in the supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
4	SWE	If the specified area is contained within one protection area and writing to that area is permitted in the supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
3	SRE	If the specified area is contained within one protection area and reading from that area is permitted in the supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
2	UXE	If the specified area is contained within one protection area and execution is permitted for that area in the user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
1	UWE	If the specified area is contained within one protection area and writing to that area is permitted in the user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
0	URE	If the specified area is contained within one protection area and reading from that area is permitted in the user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined

(j) MPLAn ─ The lower limit address of the protection area

These registers indicate the minimum address of area n (where n = 0 to 15). These products are provided with sixteen protection areas.

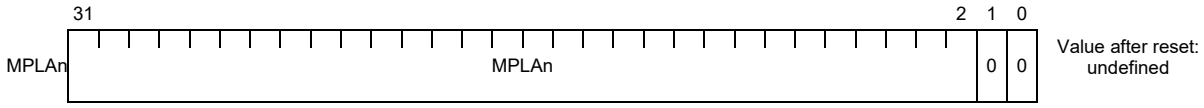


Table 3.47 MPLAn Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	MPLA31-2	These bits indicate the lower-limit address of area n. MPLA1 and 0 are implicitly set to 0.	R/W	Undefined
1, 0	Reserved	These bits are read as 0. The write value should be 0.	R	0

(k) MPUAn ─ The upper limit address of the protection area

These registers indicate the upper limit address of area n (where n = 0 to 15). These products are provided with sixteen protection areas.

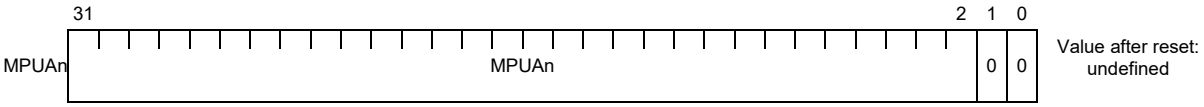


Table 3.48 MPUAn Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	MPUA31-2	These bits indicate the upper limit address of area n. MPUA1 and 0 are implicitly set to 1.	R/W	Undefined
1, 0	Reserved	These bits are read as 0. The write value should be 0.	R	0

(l) MPATn ─ The attribute of the protection area

These registers indicate the attributes of area n (where n = 0 to 15). These products are provided with sixteen protection areas.

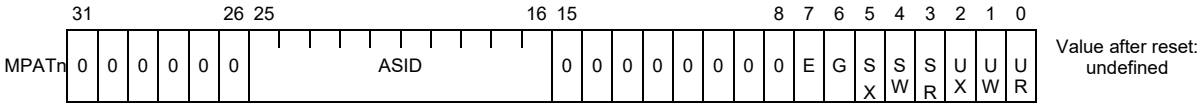


Table 3.49 MPATn Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 26	Reserved	These bits are read as 0. The write value should be 0.	R	0
25 to 16	ASID	These bits indicate the ASID value to be used as the area match condition	R/W	Undefined
15 to 8	Reserved	These bits are read as 0. The write value should be 0.	R	0
7	E	This bit indicates whether area n is enabled or disabled. 0: Area n is disabled. 1: Area n is enabled.	R/W	0
6	G	0: ASID match is the condition. 1: ASID match is not the condition. When this bit is 0, the condition of the area match is MPATn.ASID =ASID.ASID. When this bit is 1, the area match of the values of MPATn.ASID and ASID.ASID is not the condition.	R/W	Undefined
5	SX	This bit indicates the execution privilege in supervisor mode*1. 0: Execution is disabled. 1: Execution is enabled.	R/W	Undefined
4	SW	This bit indicates the write permission in supervisor mode*1. 0: Writing is disabled. 1: Writing is enabled.	R/W	Undefined
3	SR	This bit indicates the read permission in supervisor mode*1. 0: Reading is disabled. 1: Reading is enabled.	R/W	Undefined
2	UX	This bit indicates the execution privilege in user mode. 0: Execution is disabled. 1: Execution is enabled.	R/W	Undefined
1	UW	This bit indicates the write permission in user mode. 0: Writing is disabled. 1: Writing is enabled.	R/W	Undefined
0	UR	This bit indicates the read permission in user mode. 0: Reading is disabled. 1: Reading is enabled.	R/W	Undefined

Note 1. If access is restricted in SV mode, execution of MDP exceptions or the MIP exception handling itself might not be possible depending on the settings. Be careful with giving access restrictions so that access to the memory area necessary for the exception handler and exception handling is permitted.

(6) Cache operation function register

(a) Cache control function system register

Cache control function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and a selection ID.

Table 3.50 List of Cache Operation Function System Registers

Register Number (reg.ID, sel.ID)	Name	Function	Access Permission
SR16, 4	ICTAGL	Instruction cache tag Lo access	SV
SR17, 4	ICTAGH	Instruction cache tag Hi access	SV
SR18, 4	ICDATL	Instruction cache data Lo access	SV
SR19, 4	ICDATH	Instruction cache data Hi access	SV
SR24, 4	ICCTRL	Instruction cache control	SV
SR26, 4	ICCFG	Instruction cache configuration	SV
SR28, 4	ICERR	Instruction cache error	SV

(b) ICTAGL ─ Instruction cache tag Lo access

This register is used for CIST and CILD instructions for the instruction cache. This register retains values to be stored in the tag RAM of the instruction cache by the execution of CIST instructions and values read from the tag RAM of the instruction cache by the execution of CILD instructions.

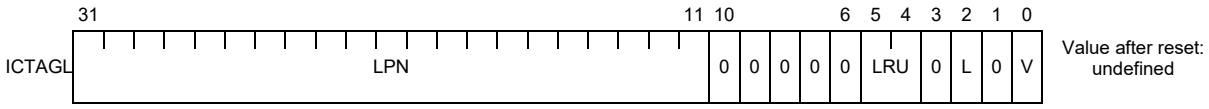


Table 3.51 ICTAGL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 11	LPN	These bits retain the values of bits 24 to 11, i.e. the physical page numbers. When writing, always write 0 to bits 31 to 25.	R/W	Undefined
10 to 6	Reserved	These bits are read as 0. The write value should be 0.	R	0
5, 4	LRU	These bits indicate the LRU information of the specified cache line. The CIST instruction cannot be used to change the LRU information to desired values.	R/W	Undefined
3	Reserved	This bit is read as 0. The write value should be 0.	R	0
2	L	This bit retains the lock information.	R/W	Undefined
1	Reserved	This bit is read as 0. The write value should be 0.	R	0
0	V	This bit retains whether the specified cache line is enabled or disabled.	R/W	Undefined

(c) ICTAGH — Instruction cache tag Hi access

This register is used for a CIST or CILD instruction for the instruction cache. This register holds the value to be stored in the instruction cache tag RAM by the execution of CIST instruction and the value read from the instruction cache tag RAM by the execution of CILD instruction.

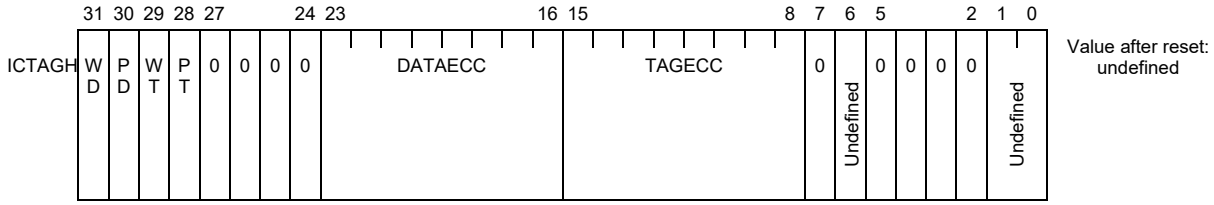


Table 3.52 ICTAGH Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	WD	When this bit is set to 1, executing a CIST instruction updates the cache data RAM.	R/W	Undefined
30	PD	When this bit is set to 1, the value in the DATAECC field of this register is written to the ECC field of the data RAM on execution of a CIST instruction. When this bit is cleared to 0, the ECC is automatically generated from the written data.	R/W	Undefined
29	WT	When this bit is set to 1, executing a CIST instruction updates the cache data RAM.	R/W	Undefined
28	PT	When this bit is set to 1, the value in the TAGECC field is written to the ECC of the tag RAM on execution of a CIST execution. When this bit is cleared to 0, the ECC is automatically generated from the written data.	R/W	Undefined
27 to 24	Reserved	These bits are read as 0. The write value should be 0.	R	0
23 to 16	DATAECC	These bits retain ECC of the data RAM.	R/W	Undefined
15 to 8	TAGECC	These bits retain ECC of the tag RAM. Be sure to set bits 15 and 14 to 0.	R/W	Undefined
7	Reserved	This bit is read as 0. The write value should be 0.	R	0
6	Reserved	When read, an undefined value is returned. The write value should be 0.	R	Undefined
5 to 2	Reserved	These bits are read as 0. The write value should be 0.	R	0
1, 0	Reserved	When read, an undefined value is returned. The write value should be 0.	R	Undefined

(d) ICDATL — Instruction cache data Lo access

This register is used for a CIST or CILD instruction for the instruction cache. This register stores the value to be stored in the instruction cache data RAM by the execution of CIST instruction and the value read from the instruction cache data RAM by the execution of CILD instruction.

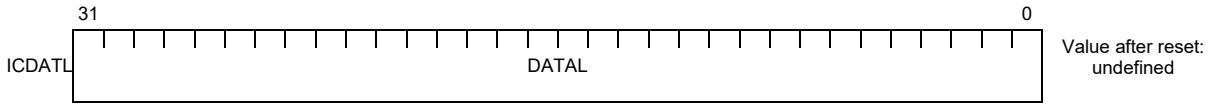


Table 3.53 ICDATL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	DATAL	These bits retain the values of the 32 lower-order bits from 64 bits of the instruction data of the block within the specified cache line. The offset of the index specifies the target range of bit numbers. Index offset = 00000: bits 31 to 0 Index offset = 01000: bits 95 to 64 Index offset = 10000: bits 159 to 128 Index offset = 11000: bits 223 to 192 The arrangement of data is as shown in **Figure 3.4**.	R/W	Undefined

(e) ICDATH — Instruction cache data Hi access

This register is used for a CIST or CILD instruction for the instruction cache. This register stores the value to be stored in the instruction cache data RAM by the execution of CIST instruction and the value read from the instruction cache data RAM by the execution of CILD instruction.

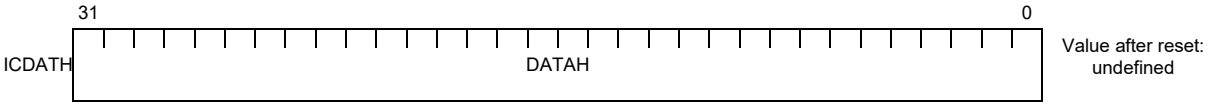


Table 3.54 ICDATH Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	DATAH	These bits retain the values of the 32 higher-order bits from 64 bits of the instruction data of the block within the specified cache line. The index offset specifies the bit number to be retained. Index offset = 00000: bits 63 to 32 Index offset = 01000: bits 127 to 96 Index offset = 10000: bits 191 to 160 Index offset = 11000: bits 255 to 224 The arrangement of data is as shown in **Figure 3.4**.	R/W	Undefined

(f) ICCTRL ─ Instruction cache control

This register controls the instruction cache.

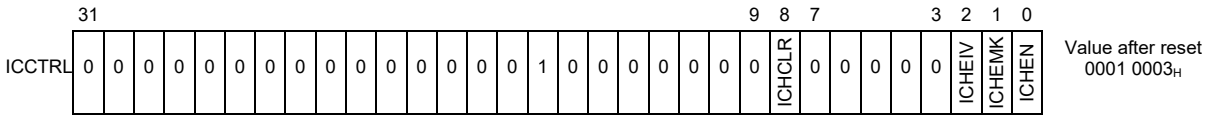


Table 3.55 ICCTRL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 17	Reserved	These bits are read as 0. The write value should be 0.	R	0
16	Reserved	This bits are read as 1. The write value should be 1.	R	1
15 to 9	Reserved	These bits are read as 0. The write value should be 0.	R	0
8	ICHCLR	Setting this bit to 1 selects clearing of the whole instruction cache in a single operation. After this bit has been set to 1, it will be read as 1 until clearing is completed. The bit is cleared to 0 once clearing of the cache is completed.	R/W	0
7 to 3	Reserved	These bits are read as 0. The write value should be 0.	R	0
2	ICHEIV	Setting this bit to 1 allows the instruction cache to be automatically disabled (ICHEN bit to be cleared to 0) when a cache error occurs.	R/W	0
1	ICHEMK	Setting this bit to 1 selects masking of cache error exception notifications for the CPU when a cache error occurs.	R/W	1
0	ICHEN	This bit disables or enables the instruction cache. 0: Instruction cache is disabled. 1: Instruction cache is enabled. This bit is read as the previous value until the setting is actually reflected in the instruction cache.	R/W	1

(g) ICCFG ─ Instruction cache configuration register

This register shows the configuration of the instruction cache.

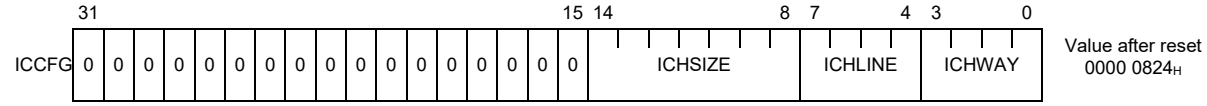


Table 3.56 ICCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 15	Reserved	These bits are read as 0. The write value should be 0.	R	0
14 to 8	ICHSIZE	These bits indicate the capacity (in Kbytes) of the instruction cache. 000 1000: 8 Kbytes	R	08_H
7 to 4	ICHLINE	These bits indicate the number of lines per 1 way in the instruction cache. 0100: 64 lines	R	2_H
3 to 0	ICHWAY	These bits indicate the number of ways of the instruction cache. 0100: 4 ways	R	4_H

(h) ICERR ─ Instruction cache error

This register stores cache error data of the instruction cache.

Once the ICHERR bit is set to 1, indicators that subsequent cache errors have occurred are not stored in this register until the ICHERR bit is cleared to 0. However, bits that indicate error states (ESMH, ESPBSE, ESTE1, ESTE2, ESDC, ESDE) continue to be cumulatively updated.

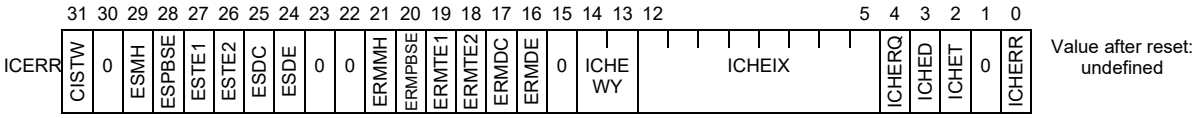


Table 3.57 ICERR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	CISTW	This bit is set to indicate that the destination way specified for a CISTI instruction was in error. Although the entry information is overwritten so that writing is completed, the V bit will be cleared the next time the cache line is read (i.e. reading will be judged to have missed the cache). However, setting of this bit is not accompanied by an exception for the CPU.	R/W	Undefined
30	Reserved	This bit is read as 0. The write value should be 0.	R	0
29	ESMH	Error status: Multi-hit	R/W	Undefined
28	ESPBSE	Error status: WAY error	R/W	Undefined
27	ESTE1	Error status: 1-bit error in the tag RAM	R/W	Undefined
26	ESTE2	Error status: 2-bit error in the tag RAM	R/W	Undefined
25	ESDC	Error status: 1-bit correction in the data RAM	R/W	Undefined
24	ESDE	Error status: 2-bit error in the data RAM	R/W	Undefined
23, 22	Reserved	These bits are read as 0. The write value should be 0.	R	0
21	ERMMH	Error exception notification mask: Multi-hit	R/W	0
20	ERMPBSE	Error exception notification mask: WAY error	R/W	0
19	ERMTE1	Error exception notification mask: 1-bit error in the tag RAM	R/W	0
18	ERMTE2	Error exception notification mask: 2-bit error in the tag RAM	R/W	0
17	ERMDC	Error exception notification mask: 1-bit correction in the data RAM	R/W	0
16	ERMDE	Error exception notification mask: 2-bit error in the data RAM	R/W	0
15	Reserved	This bit is read as 0. The write value should be 0.	R	0
14, 13	ICHEWY	These bits retain the WAY number where a cache error occurred.	R/W	Undefined
12 to 5	ICHEIX	These bits retain the cache index where a cache error occurred.	R/W	Undefined
4	ICHERQ	Setting this bit to 1 indicates that the CPU is being notified of a cache error exception. If cache error exceptions are masked, however, the CPU is not notified of an exception even when this bit is set to 1.	R/W	0
3	ICHED	This bit indicates that an error occurred in the data RAM.	R/W	0
2	ICHET	This bit indicates that an error occurred in the tag RAM.	R/W	0
1	Reserved	This bit is read as 0. The write value should be 0.	R	0
0	ICHERR	This bit is set to 1 when a cache error occurred.	R/W	0

(7) Data buffer operation function register

(a) Data buffer control function system register

Data buffer control function system registers are read from or written to by using the LDSR or STSR instruction and specifying the system register number, which is made up of a register number and a selection ID.

Table 3.58 List of Data Buffer Operation Function Registers

egister Number (reg.ID, sel.ID)	Name	Function	Access Permission
SR24, 13	CDBCR	Data Buffer Control Function Registers	SV

(b) CDBCR ─ Data buffer control register

This register controls the data buffers.

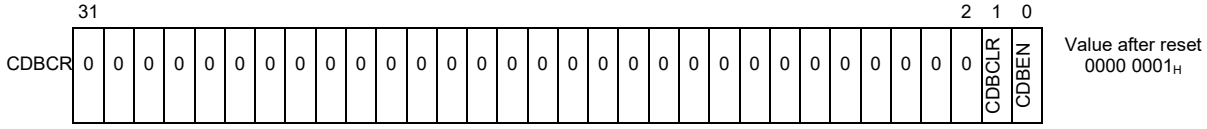


Table 3.59 CDBCR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	Reserved	These bits are read as 0. The write value should be 0.	R	0
1	CDBCLR	When this bit is set to 1, the data buffer is cleared at a time. This bit is always read as 0.	W	0
0	CDBEN	This bit disables or enables the data buffer. 0: Data buffer is disabled. 1: Data buffer is disabled.	R/W	1

3.2.2 Instruction Cache and Data Buffer

3.2.2.1 Features

An 8-Kbyte and 4-way set-associative instruction cache is mounted between the CPU and the code flash. The instruction cache and the code flash are connected to each other via a 256-bit dedicated bus to minimize penalties caused by a cache miss-hit. Also a data buffer is mounted between the CPU and the code flash to achieve high-speed data access. The area of 32 Mbytes from 0000 0000_H to 01FF $FFFF_H$ in the address space is intended for the instruction cache and data buffer.

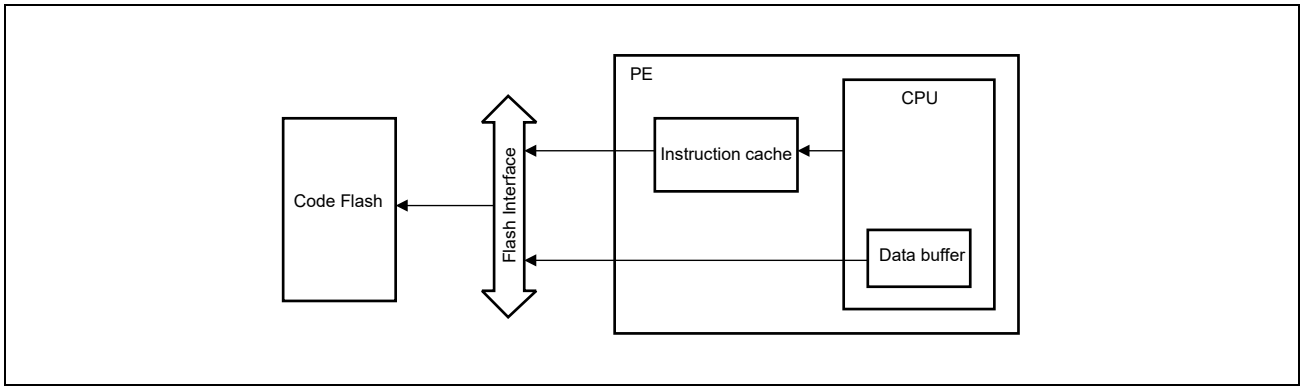


Figure 3.2 Instruction Cache and Data Buffer

3.2.2.2 Instruction Cache Function

The instruction cache is a 4-Way set associative cache with a total capacity of 8 Kbytes. The size of each cache is 32 bytes. When a cache is missed, a line is refilled, with LRU as the algorithm that controls replacement. The amount of data that can be fetched at once is 8 bytes, which means, the 8 bytes of data at a specified offset within one line are selected.

The Ways are divided into two groups: Way group 0, consisting of Ways 0 and 1, and Way group 1, consisting of Ways 2 and 3. The Way group for use is selected by decoding the address information of the destination for access.

The cache destination is the instruction fetch access to the code flash area.

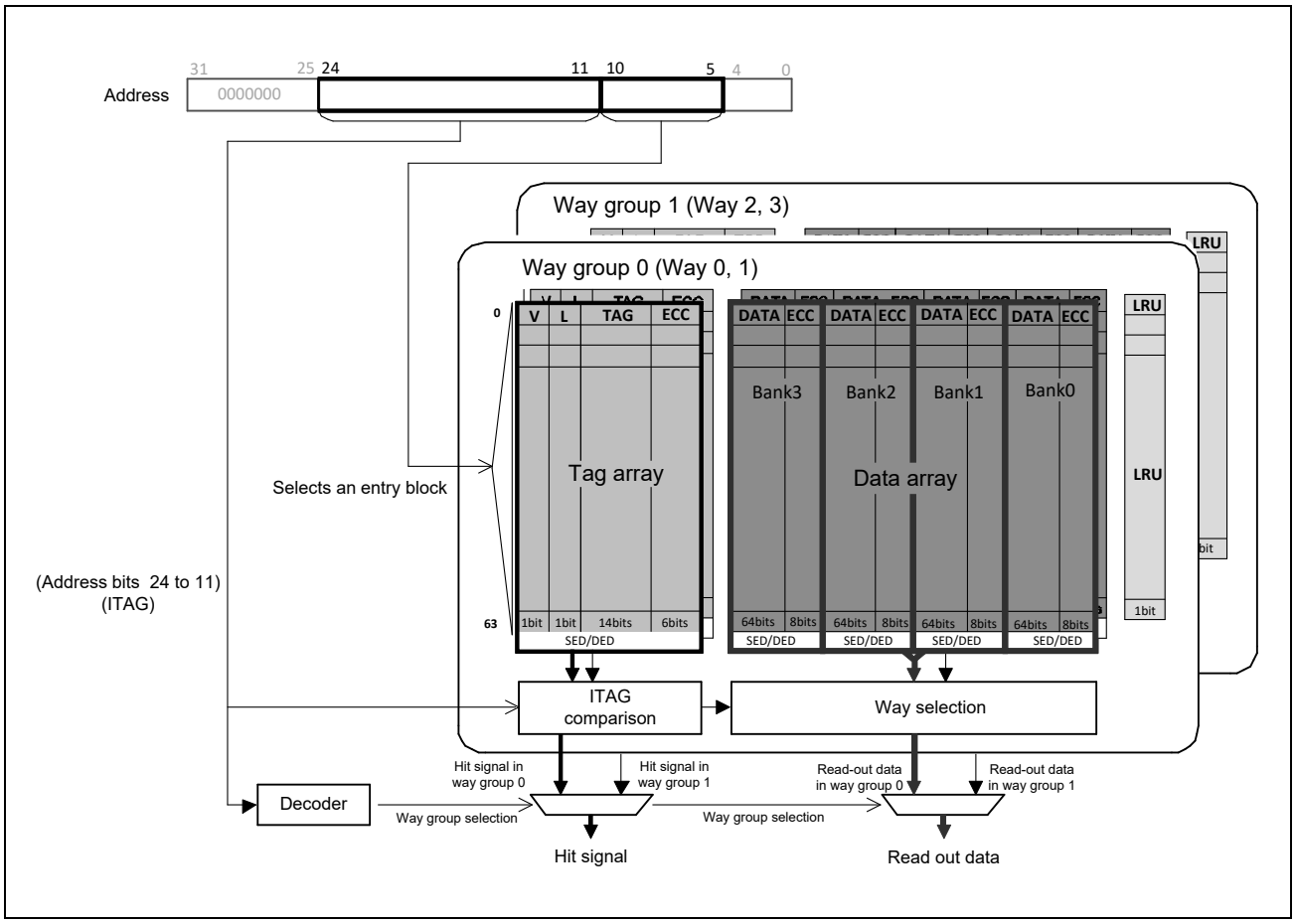


Figure 3.3 Instruction Cache Configuration

Tag Array

TAG	Among 32 bits in the operable addresses of the data line to be cached, bits 24 to 11 are stored in this bit. The TAG bit is not initialized by a reset.
ECC	The ECC of the tag array is stored in this bit. The ECC bit is not initialized by a reset.

Valid/Lock

V bit	This bit indicates whether valid data is stored in the cache line. Setting of this bit to 1 makes the cache line data valid. The V bit is initialized to 0 by a reset.
L bit	This bit indicates whether a cache line is locked or not. Setting of this bit to 1 locks the cache line and it cannot be replaced with new data. The L bit is valid only when the V bit is 1, and it is initialized by a reset.

Data Array

DATA	The 256 bits of data in cache line is stored with 64 bits in each bank. The way and offset determine the locations at which data are stored. The arrangement of data is as shown in **Figure 3.4**. The data are not initialized by a reset.
ECC	The ECC of data array are stored. The ECC is not initialized by a reset.

LRU

LRU	The information in the same Way group is stored in this data array. LRU is initialized by reset.

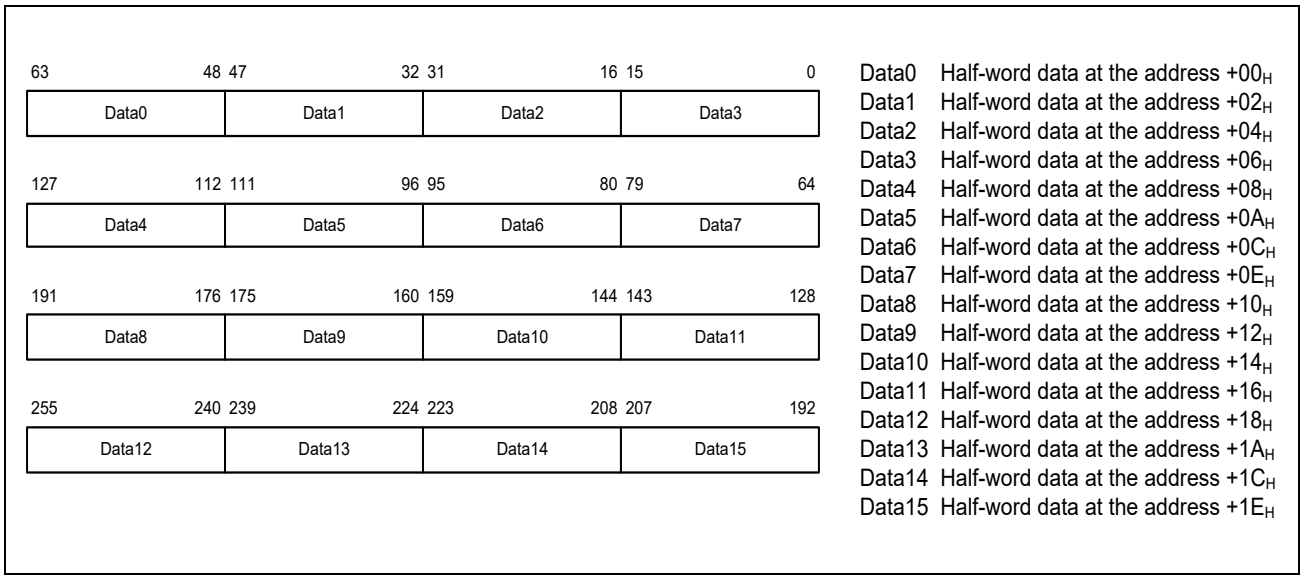


Figure 3.4 Arrangement of Data within the Cache

3.2.2.3 Data buffer function

The eight-line buffer with 128 bits per line is mounted as a data buffer. The data of 256 bits read from the code flash is stored in the data buffer with 128 bits per line. The data is read out from the data buffer after the next access to the same address, so the code flash is not accessed again.

3.2.3 Inter-Processor Interrupts

Registers (IPIR_CHn) for interrupt communication between PE1 and PE2 are provided for four channels. IPIR_CH0 to IPIR_CH3 are assigned to CH0 to CH3 of user interrupt (EIINT). An interrupt for specific PEs (including own PE) can be requested by manipulating bits corresponding to respective PEs.

3.2.3.1 Inter-Processor Interrupt Control Registers

Each of these registers are located in the CPU peripheral of the individual PEs. Each PE has its own IPIR_CH0 to IPIR_CH3 registers and cannot access those in other PEs.

Table 3.60 List of Registers

Register Symbol	Register Name	R/W	Value after Reset	Access Size				Address
				1	8	16	32	
IPIR_CH0	Inter-PE interrupt register 0	R/W	0000 0000$_H$	√	√	√	√	FFFE EC80$_H$
IPIR_CH1	Inter-PE interrupt register 1	R/W	0000 0000$_H$	√	√	√	√	FFFE EC84$_H$
IPIR_CH2	Inter-PE interrupt register 2	R/W	0000 0000$_H$	√	√	√	√	FFFE EC88$_H$
IPIR_CH3	Inter-PE interrupt register 3	R/W	0000 0000$_H$	√	√	√	√	FFFE EC8C$_H$

(1) IPIR_CHn ─ Inter-PE interrupt register n (n = 0 to 3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PE2	PE1
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 3.61 IPIR_CHn Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	These bits are read as 0. The write value should be 0.
1	PE2	Inter-PE Interrupt Request to PE2 Writing 1 to this bit makes an interrupt request to PE2. This bit is automatically cleared to 0 once the interrupt request has been notified. 0: Inter-PE interrupt request output is not specified or an interrupt request is not being output. 1: Interrupt request output is specified or an interrupt request is being output.
0	PE1	Inter-PE Interrupt Request to PE1 Writing 1 to this bit makes an interrupt request to PE1. This bit is automatically cleared to 0 once the interrupt request has been notified. 0: Inter-PE interrupt request output is not specified or an interrupt request is not being output. 1: Interrupt request output is specified or an interrupt request is being output.

3.2.4 Reliability Functions

3.2.4.1 PE Guard Function (PEG)

(1) Overview of the PEG function

The PEG is a constituent of the Slave Guard function and blocks unauthorized accesses to the resources in the PE (local RAM) from the external bus master. In the state after a reset, all access other than by the bus master of its own PE is blocked. Setting the registers listed in **Section 3.2.4.1 (3), List of Protection with PEG Setting Registers** enables access other than by the bus masters of its own PE.

(a) Detecting PE guard violation

If the bus master outside the PE for which PE guard is set makes an unauthorized access to the resource area in the PE, the access is detected as a PE guard violation.

(b) Blocking unauthorized accesses

When a PE guard violation is detected, unauthorized accesses to the internal resources of the PE are blocked to prevent the contents of PE resources from being modified illegally.

(c) Notifying occurrence of violation

When a PE guard violation is detected, it is notified to ECM. When the DMAC or DTS makes an attempt of unauthorized access, meanwhile, a DMA transfer error is detected.

(2) Protection Made by PEID and SPID

- Setting protection with PEG
  - The PE can set protection for up to four areas in its own local RAM by specifying the address.
  - The range of the target area is specified by the base address and the mask bit (4 Kbytes to 16 Mbytes).
  - Read access and write access are respectively enabled for the individual areas.
  - Protection is enabled or disabled for the individual areas on each system protection identifier (SPID) basis and on each processor element identifier (PEID) basis.

- Access permission by the system processor element identifier (PEID) and system protection identifier (SPID) (see **Figure 3.5**)

  1. When the local RAM area is to be accessed, go to step 2.

     Otherwise, return an error response.

  2. When any of enabled areas 0 to 3 is to be accessed, go to step 3.

     Otherwise, return an error response.

  3. Whether all the conditions below for the relevant area met or not.

     - The system protection identifier (SPID) is enabled
     - The processor element identifier (PEID) is enabled
     - The required operation (read or write) is enabled.

     Otherwise, return an error response.

Memory access which is not allowed in the table below leads to an error being returned.

Note that CAXI and bit operation instructions (BitOp in the table) cannot proceed unless both reading and writing are allowed.

Table 3.62 Access Allowed for Each Request Type

Request Type	Read Access	Write Access
READ	Allowed	Not allowed
WRITE	Not allowed	Allowed
BitOp	Allowed	Allowed
CAXI	Allowed	Allowed
LDL	Allowed	Not allowed
STC	Not allowed	Allowed

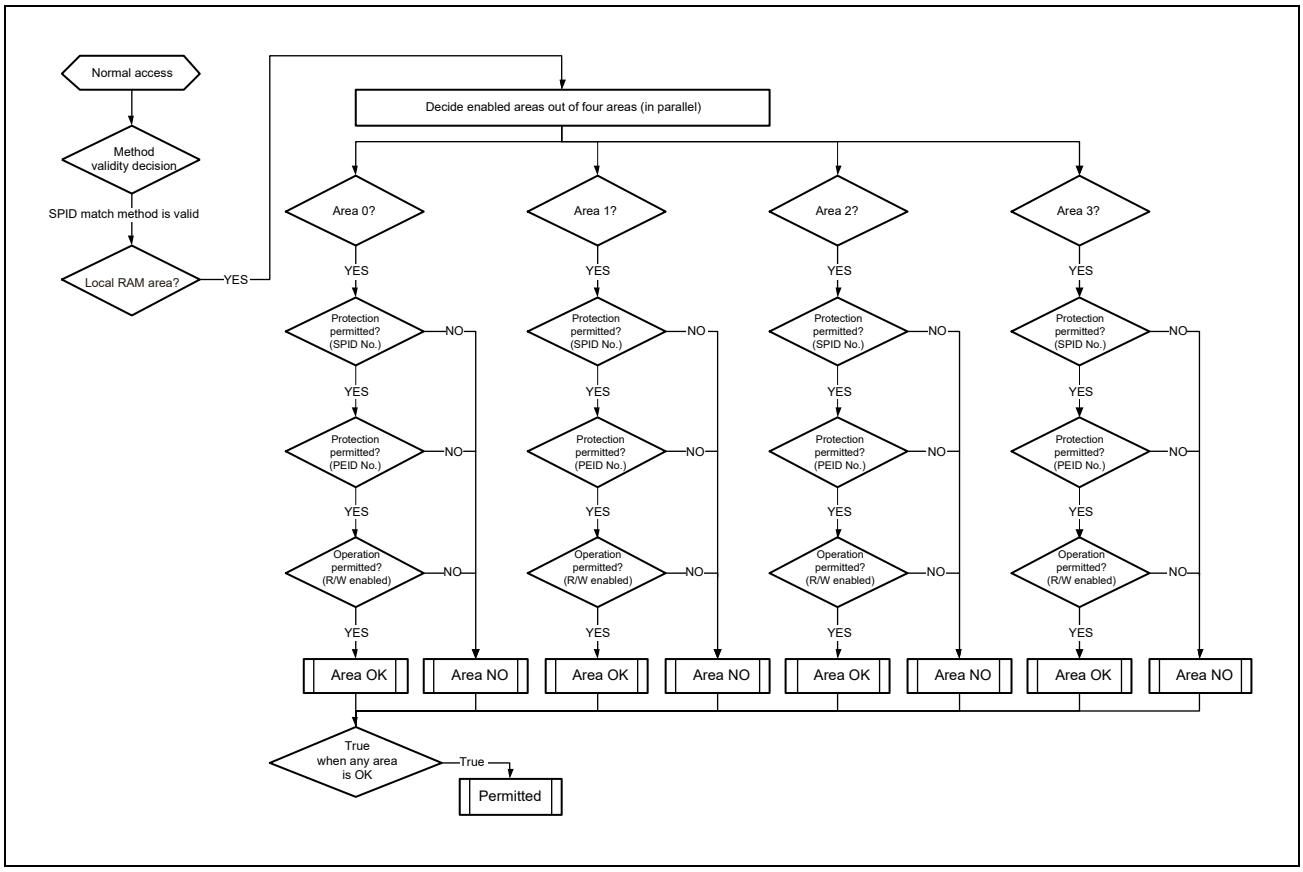


Figure 3.5 Access Permission by the Processor Element (PEID) and System Protection Identifier (SPID)

(3) List of Protection with PEG Setting Registers

Make necessary settings for the following registers to protect PE resources from unauthorized accesses by the external bus masters.

- Detection of accesses to the following areas are enabled: the areas for peripheral devices, and the local RAM area in the PE.
- Access permission for the register sets is not controlled by the PEG function. Make necessary settings for access permission by the IPG or other function.

Table 3.63 PEG Register Base Address: FFFE E600$_H$

Address Offset	Size (byte)	Register Name	Symbol	Permission	R/W	Operable Bit				Value after Reset
						1	8	16	32	
+00C$_H$	2	PE guard PEID&SPID master decision control register	PEGSP	—	R/W	—	√	√	—	0000$_H$
+080$_H$	4	PE guard area mask setting register 0	PEGG0MK	—	R/W	—	√	√	√	FFE0 0000$_H$
+084$_H$	4	PE guard area base setting register 0	PEGG0BA	—	R/W	—	√	√	√	*1
+088$_H$	4	PE guard area SPID setting register 0	PEGG0SP	—	R/W	—	√	√	√	0000 0000$_H$
+08C$_H$	4	PE guard area PEID setting register 0	PEGG0PE	—	R/W	—	√	√	√	0000 0000$_H$
+090$_H$	4	PE guard area mask setting register 1	PEGG1MK	—	R/W	—	√	√	√	FFE0 0000$_H$
+094$_H$	4	PE guard area base setting register 1	PEGG1BA	—	R/W	—	√	√	√	*1
+098$_H$	4	PE guard area SPID setting register 1	PEGG1SP	—	R/W	—	√	√	√	0000 0000$_H$
+09C$_H$	4	PE guard area PEID setting register 1	PEGG1PE	—	R/W	—	√	√	√	0000 0000$_H$
+0A0$_H$	4	PE guard area mask setting register 2	PEGG2MK	—	R/W	—	√	√	√	FFE0 0000$_H$
+0A4$_H$	4	PE guard area base setting register 2	PEGG2BA	—	R/W	—	√	√	√	*1
+0A8$_H$	4	PE guard area SPID setting register 2	PEGG2SP	—	R/W	—	√	√	√	0000 0000$_H$
+0AC$_H$	4	PE guard area PEID setting register 2	PEGG2PE	—	R/W	—	√	√	√	0000 0000$_H$
+0B0$_H$	4	PE guard area mask setting register 3	PEGG3MK	—	R/W	—	√	√	√	FFE0 0000$_H$
+0B4$_H$	4	PE guard area base setting register 3	PEGG3BA	—	R/W	—	√	√	√	*1
+0B8$_H$	4	PE guard area SPID setting register 3	PEGG3SP	—	R/W	—	√	√	√	0000 0000$_H$
+0BC$_H$	4	PE guard area PEID setting register 3	PEGG3PE	—	R/W	—	√	√	√	0000 0000$_H$

Note 1. This value is FEA0_0000$_H$ for CPU1 (PE1), FE80_0000$_H$ for CPU2 (PE2), and FE60_0000$_H$ for the SubCPU (PE3).

(4) Register set

(a) PEGSP ─ PE guard PEID & SPID master decision control register

This register is used to enable or disable access by the external master to the resources in the PE. The value of the SPEN bit is 0 at its initial state, which disables access to the resources in the PE from outside. Setting the SPEN bit to 1 enables access from the external master under the conditions set by PEGGnMK and PEGGnBA.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SPEN
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 3.64 PEGSP Register Contents

Bit Position	Bit Name	Function
15 to 1	Reserved	These bits are read as 0. The write value should be 0.
0	SPEN	This bit enables or disables access settings by the bus master having PEID and SPID. 0: Detection of access by the external master having PEID and SPID is disabled. 1: Detection of access by the external master having PEID and SPID is enabled.

(b) PEGGnMK ─ PE guard area n mask setting register

In combination with the PEGGnBA register, this register specifies a range or ranges of the areas n to be protected with PEG. Setting a GnMASK bit to 1 masks the corresponding address bit of the PEGGnBA register and places the corresponding area or areas within the range of PE guard area n. Note that the minimum setting unit for the PE guard area n is 4 Kbytes.

Though PEGGnMK[31:21] is readable and writable, the value of these bits is ignored when judgment of access permission by PEID and SPID is made. At this time, the value is treated as follows:

- PEGGnMK[31:21] = $1111\_1111\_111_B$

In combination with the PEGGnBA register, this register protects the areas from an external master, which are at $FEA0\_0000_H$ to $FEBF\_FFFF_H$ in CPU1 (PE1), at $FE80\_0000_H$ to $FE9F\_FFFF_H$ in CPU2 (PE2), and at $FE60\_0000_H$ to $FE7F\_FFFF_H$ in the SubCPU (PE3). These areas respectively corresponds to the local RAM area in each PE.

Example: When PEGGnBA[31:12] = $FEBF6_H$ and PEGGnMK[31:12] = 00008_H are set, the PE guard areas n are $FEBF\_6000_H$ to $FEBF\_6FFF_H$ and $FEBF\_E000_H$ to $FEBF\_EFFF_H$.

Writing to the PEGGnMK register automatically clears the GnEN register in the PEGGnBA registers to 0. This prevents unintended accesses from an external master during configuration of access areas.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GnMASK															
Value after Reset	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GnMASK				—	—	—	—	—	—	—	—	—	—	—	—
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 3.65 PEGGnMK Register Contents

Bit Position	Bit Name	Function
31 to 12	GnMASK	These bits determine whether to mask the base address bits PEGGnBA[31:12] that specify the range of the area n protected with PE guard. 0: Target address bits are compared when determining the PE guard area. 1: Target address bits are not compared when determining the PE guard area.
11 to 0	Reserved	These bits are read as 0. The write value should be 0.

(c) PEGGnBA ─ PE guard area n base setting register

In combination with the PEGGnMK register, this register specifies a range or ranges within PE guard area n and sets the access enable conditions for the specified area. Setting the GnEN bit to 1 brings the address enable conditions specified by this register and the PEGGnMK register into effect. Though PEGGnBA[31:21] is readable and writable, the value of these bits is ignored when decision of access permission by PEID and SPID is made. At this time, the value is treated as follows:

- When an access to the resource in CPU1 (PE1) is detected: PEGGnBA[31:21] = $1111\_1110\_101_B$
- When an access to the resource in CPU2 (PE2) is detected: PEGGnBA[31:21] = $1111\_1110\_100_B$
- When an access to the resource in the SubCPU (PE3) is detected: PEGGnBA[31:21] = $1111\_1110\_011_B$

In combination with the PEGGnMK registers, this register protects the areas from an external master, which are at $FEA0\_0000_H$ to $FEBF\_FFFF_H$ in CPU1 (PE1), $FE80\_0000_H$ to $FE9F\_FFFF_H$ in CPU2 (PE2), and at $FE60\_0000_H$ to $FE7F\_FFFF_H$ in the SubCPU (PE3). These areas respectively corresponds to the local RAM area in each PE.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GnBASE															
Value after Reset	1	1	1	1	1	1	1	0	*1	*1	*1	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GnBASE				—	—	—	—	—	—	—	GnLOCK	—	GnWR	GnRD	GnEN
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Note 1. This value is 101_B for CPU1 (PE1), 100_B for CPU2 (PE2), 011_B for the SubCPU (PE3).

Table 3.66 PEGGnBA Register Contents

Bit Position	Bit Name	Function
31 to 12	GnBASE	The value for a base address that specifies the PE guard area n is set in these bits.
11 to 5	Reserved	These bits are read as 0. The write value should be 0.
4	GnLOCK	Disables writing to the system register in the protected area n. This bit can only be set to 1 and cannot be set to 0. 0: Writing to PEGGnMK, PEGGnBA, PEGGnSP, and PEGGnBA is enabled. 1: Writing to PEGGnMK, PEGGnBA, PEGGnSP, and PEGGnBA is disabled.
3	Reserved	This bit is read as 0. The write value should be 0.
2	GnWR	Permits write access to the PE guard area n. 0: Write access is rejected. 1: Write access is permitted.
1	GnRD	Permits read access to the PE guard area n. 0: Read access is rejected. 1: Read access is permitted.
0	GnEN	Enables settings for access enable conditions for the PE guard area n. 0: Settings for the conditions are disabled. 1: Settings for the conditions are enabled.

CAUTION

PEGGnBA.GnEN is cleared by writing to the PEGGnMK register.

(d) PEGGnSP ─ PE guard area n SPID setting register

These registers enable or disable permission for access to PE guard areas n by external masters which have the given SPID.

Specifically, setting a PEGGnSP[m] bit to 1 permits access from an external master which has the SPID m.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GnSP			
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 3.67 PEGGnSP Register Contents

Bit Position	Bit Name	Function
31 to 4	Reserved	These bits are read as 0. The write value should be 0.
3 to 0	GnSP	Each bit determines permission for access by an external master with the same SPID as the bit position to the PE guard area n. For the bit position m: 0: Access from an external master with SPID = m is rejected. 1: Access from an external master with SPID = m is permitted.

(e) PEGGnPE ─ PE guard area n PEID setting register

These registers enable or disable permission for access to PE guard areas n by external masters which have the given PEID.

Specifically, setting a PEGGnPE[m] bit to 1 permits access from an external master which has the PEID m.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GnPE							
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 3.68 PEGGnPE Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	These bits are read as 0. The write value should be 0.
7 to 0	GnPE	Each bit determines permission for access by an external master with the same PEID as the bit position to the PE guard area n. For the bit position m: 0: Access from an external master with PEID = m is rejected. 1: Access from an external master with PEID = m is permitted.

3.2.4.2 PE-internal peripheral device guard function (IPG)

(1) Overview of the IPG function

The IPG is a function to prevent unauthorized accesses to peripheral devices from the CPU core. The following features are achieved. IPG covers the peripheral devices on the CPU.

(a) Detecting violation of peripheral device guard

If the CPU makes an unauthorized access to an area (peripheral device) for which peripheral device guard is set, the access is detected as “violation of peripheral device guard”.

(b) Storing information of the unauthorized access

When a violation of peripheral device guard is detected, the information of the unauthorized access is stored in the IPG's internal register.

(c) Blocking unauthorized accesses

When a violation of peripheral device guard is detected, such access is blocked to prevent contents of peripheral devices from being modified illegally.

(d) Notifying violations

When a violation of peripheral device guard is detected, a request for generating a system error exception (SYSERR exception) is made to ask the CPU to stop the processing.

For the details on system error exception (SYSERR exception), see **Section 3.2.4.3, System Error Notification Control Function (SEG)**.

(e) Invalidating subsequent accesses

When a violation of peripheral guard is detected, subsequent accesses (regardless of authorized or not) are blocked until instructions from the CPU are received.

NOTE

Even if a request for generating an exception is immediately sent to the CPU in step (d) above, a subsequent access which has been issued by the CPU before receiving a request from the IPG, which does not know an occurrence of violation, may illegally modify contents of peripheral devices. Accesses after a violation has occurred result in unauthorized accesses.

(2) IPG function

1. This function invalidates accesses according to their attributes (including address, transfer type, and access right).
2. After an access right violation is detected, subsequent accesses are invalidated until the error flag (described later) is cleared by a writing by the software. However, invalidation is applied only to accesses from the CPU and is not applied to accesses from outside the CPU core. Invalidation is performed independently of addresses.
3. When a request for accessing different peripheral devices simultaneously is made due to misalignment or double-word access, the access is executed when all such accesses are enabled.

(3) IPG setting registers for illegal users

To protect peripheral devices from unauthorized accesses by programs in user mode, necessary settings are required for the registers listed below.

- Accesses in user mode are to be detected.
- This register set is intended for IPG settings related to user mode and for reading of the IPG settings.

Table 3.69 IPG Register Base Addresses: FFFE E000$_H$

Address Offset	Size (byte)	Register Name	Symbol	Authorization*1	R/W	Operable Bit				Value after Reset
						1	8	16	32	
+002$_H$	2	Peripheral Device Guard Violation Access Information Register	IPGECRUM	SV	R/W	—	—	√	—	Undefined (retained)
+008$_H$	4	Peripheral Device Guard Violation Access Address Register	IPGADRUM	SV	R/W	—	—	—	√	Undefined (retained)
+00D$_H$	1	Peripheral Device Guard Enable Register	IPGENUM	SV	R/W	√	√	—	—	00$_H$
+022$_H$	1	Peripheral Device Guard Setting Register 2	IPGPMTUM2	SV	R/W	√	√	—	—	00$_H$
+023$_H$	1	Peripheral Device Guard Setting Register 3	IPGPMTUM3	SV	R/W	√	√	—	—	00$_H$
+024$_H$	1	Peripheral Device Guard Setting Register 4	IPGPMTUM4	SV	R/W	√	√	—	—	00$_H$

Note 1. The registers with “SV” are only accessible in SV mode (UM = 0).

(4) Register set

(a) IPGECRUM ─ Peripheral device guard violation access information register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DS				EX	WR	RD	VD
Value after Reset	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Remark: x means the value is undefined (retained)

Table 3.70 IPGECRUM Register Contents

Bit Position	Bit Name	Function
15 to 8	Reserved	These bits are read as 0. The write value should be 0.
7 to 4	DS	Data size of the violation access is stored in these bits. 0100: Word (4byte) 0010: Halfword (2byte) 0001: Byte Other than above: Reserved
3	EX	This bit is set to 1 when a violation occurred in an instruction fetch read access. In other cases, this bit is cleared to 0.
2	WR	This bit is set to 1 when a violation occurred in a write access, bit operation, or execution of the CAXI instruction. In other cases, this bit is cleared to 0.
1	RD	This bit is set to 1 when a violation occurred in a read access, bit operation, or execution of the CAXI instruction. In other cases, this bit is cleared to 0.
0	VD	This bit is set to 1 when a violation of peripheral device guard is detected by a program with the relevant right. Even if another violation of peripheral device guard is detected while this bit is 1, data of IPGECRUM and IPGADRUM are not updated and retained.

NOTE

When the IRE bit value of the IPGENUM register (described later) is 0 and violation of peripheral device guard by a program operating in user mode is an instruction fetch read access, no bit of this register is updated.

(b) IPGADRUM ─ Peripheral device guard violation access address register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EADR															
Value after Reset	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR															
Value after Reset	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: ×: Undefined (retained)

Table 3.71 IPGADRUM Register Contents

Bit Position	Bit Name	Function
31 to 0	EADR	Address of the access in which violation occurred is stored in these bits.

NOTE

When the IRE bit value of the IPGENUM register (described later) is 0 and violation of peripheral device guard by a program operating in user mode is an instruction fetch read access, no bit of this register is updated.

(c) IPGENUM ─ Peripheral device guard enable register

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IRE	E
Value after Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 3.72 IPGENUM Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	These bits are read as 0. The write value should be 0.
1	IRE	This bit sets whether to store the access information in the peripheral device guard violation access address register and the peripheral device guard violation access information register when a violation of peripheral device guard occurred in an instruction fetch access. 0: Instruction fetch access information is not stored. 1: Instruction fetch access information is stored. Caution: If you do not want to detect speculative instruction fetches (no instruction is executed in some cases), clear this bit to 0.
0	E	This bit enables or disables the peripheral devices protection function against accesses by the relevant access right. 0: The peripheral device protection function is disabled. 1: The peripheral device protection function is enabled.

(d) IPGPMTUM2 ─ Peripheral device guard setting register 2

Bit	7	6	5	4	3	2	1	0
	—	—	W1	R1	—	—	W0	R0
Value after Reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 3.73 IPGPMTUM2 Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	These bits are read as 0. The write value should be 0.
5	W1	This bit sets whether to enable write access to IPIR or MEV or TESTCOMP. *1 0: Write access to IPIR or MEV or TESTCOMP is treated as violation. 1: Write access to IPIR or MEV or TESTCOMP is not restricted.
4	R1	This bit sets whether to enable read access to IPIR, MEV, or TESTCOMP.*1 0: Read access to IPIR or MEV or TESTCOMP is treated as violation. 1: Read access to IPIR or MEV or TESTCOMP is not restricted.
3, 2	Reserved	These bits are read as 0. The write value should be 0.
1	W0	This bit sets whether to enable write access to INTC1. 0: Write access to INTC1 is treated as violation. 1: Write access to INTC1 is not restricted.
0	R0	This bit sets whether to enable read access to INTC1. 0: Read access to INTC1 is treated as violation. 1: Read access to INTC1 is not restricted.

Note 1. Access to TESTCOMP by CPU2 is not allowed. Access to IPIR or MEV or TESTCOMP by the SubCPU is not allowed.

(e) IPGPMTUM3 — Peripheral device guard setting register 3

Bit	7	6	5	4	3	2	1	0
	—	—	W1	R1	—	—	—	—
Value after Reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R

Table 3.74 IPGPMTUM3 Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	These bits are read as 0. The write value should be 0.
5	W1	This bit sets whether to enable write access to SysErrGen. 0: Write access to SysErrGen is treated as violation. 1: Write access to SysErrGen is not restricted.
4	R1	This bit sets whether to enable read access to SysErrGen. 0: Read access to SysErrGen is treated as violation. 1: Read access to SysErrGen is not restricted.
3 to 0	Reserved	These bits are read as 0. The write value should be 0.

(f) IPGPMTUM4 — Peripheral device guard setting register 4

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	W0	R0
Value after Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 3.75 IPGPMTUM4 Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	These bits are read as 0. The write value should be 0.
1	W0	This bit sets whether to enable write access to PEG. 0: Write access to PEG is treated as violation. 1: Write access to PEG is not restricted.
0	R0	This bit sets whether to enable read access to PEG. 0: Read access to PEG is treated as violation. 1: Read access to PEG is not restricted.

3.2.4.3 System Error Notification Control Function (SEG)

Errors due to an instruction fetch or data access can be the sources of system error exceptions. A SYSERR exception is an FE level exception from which return or recovery is not possible.

For source codes (FEIC) of the SYSERR exceptions and error handling, see **Table 3.78**.

SEG (SysErrGen) controls the notification and record of the error by the data access. Though errors by instruction fetch access are not conveyed to the SEG, instruction cache errors occurred on the RAM are notified to the SEG.

For details, see **Section 3.2.4.3 (2) (a) SEGCONT ─ Error notification control register** and **Section 3.2.4.3 (3) (c) Supplementary notes on SYSERR exception**.

Multiple error occurrence inputs are categorized according to error factors, and are processed sequentially from the highest-priority error factor, generating an FE-level asynchronous exception (SYSERR).

The priorities for error sources are based on their bit positions in the SEGFLAG register. Error factors of lower-order bits take precedence over error factors of higher-order bits.

Error information is recorded once regardless of error frequency.

The error with the highest priority of error factor (in case errors occurred simultaneously) is valid. Recorded error information is not overwritten by subsequent errors

(1) List of SEG function control registers

Table 3.76 Base Address of SEG Register: FFFE E980$_H$

Address Offset	Size (byte)	Register Name	Symbol	Right	R/W	Operable Bit Width				Value After Reset
						1	8	16	32	
+00$_H$	2	Error Notification Control Register	SEGCONT	—	R/W*1	—	—	√	—	0000$_H$
+02$_H$	2	Error Occurrence Retention Register	SEGFLAG	—	R/W*1	—	—	√	—	0000$_H$
+08$_H$	4	Error Source Retention Register (Address)	SEGADDR	—	R/W*1	—	—	√	√	Undefined (retained)

Note 1. Write access made in user mode is ignored.

NOTE

- If an access is made with an address offset or operable bit other than those specified above, an error response is returned.
- Write access is only possible in supervisor mode (UM = 0). Attempting to write in other modes leads to an error being returned.
- No restriction is provided for read accesses. Read accesses to ranges permitted by other protection facilities are enabled at any time.

(2) Register Set

(a) SEGCONT ─ Error notification control register

This register is used to enable (= 1) or disable (= 0) notification of SYSERR request in response to error flags that holds the data indicating the state of error occurrence by each factor.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PBPE	HBPE	GRME *1	LRME	CFBE	LSUE	—	ICCE	—	—
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R

Note 1. This is configured as a reserved bit for the SubCPU.

Table 3.77 SEGCONT Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 10	Reserved	These bits are read as 0. The write value should be 0.
9	PBPE	This bit notifies of a response to an error in P-Bus. The error includes the following: • Bus parity error
8	HBPE	This bit notifies of a response to an error in reading from the resources in the SubCPU by CPU1 or CPU2, or from the resources in CPU1 or CPU2 or from the global RAM by the SubCPU. The error includes the following: • Bus parity error
7	GRME*1	This bit notifies of a response to an error in data access to the global RAM. The error includes the following: • Uncorrectable ECC error (including the case where 1-bit correction is disabled) • Address parity error (including the case where redundant bit correction is disabled)
6	LRME	This bit notifies of a response to an error in data access to the local RAM of its own. The error includes the following: • Uncorrectable ECC error (including the case where 1-bit correction is disabled). • Address parity error (including the case where redundant bit correction is disabled)
5	CFBE	This bit notifies of a response to an error in access to the code flash. The error includes the following: • Uncorrectable ECC error (including the case where 1-bit correction is disabled). • Address parity error (including the case where redundant bit correction is disabled)
4	LSUE	This bit notifies of a response to a guard error or error response in data access. The guard error and error response includes the following: • Notification of a response to a P-Bus error (excluding errors in writing to P-Bus) – When an unimplemented area (FFFF 7900_H-FFFF $7EFF_H$) is accessed – PE-internal peripheral device guard (IPG) error – Access error to the local RAM of other PE (PEG error) – P-Bus guard error (for P-Bus guard for each register regarding the INTC2, PDMA, or the global RAM) • An access to an unimplemented area on the on-chip I/O register (self area) • Response to an occurrence of the following errors in the global RAM – An access guard violation error – An access to an unimplemented area • Response to an occurrence of the following error in the local RAM – An access to an unimplemented area • Response to an occurrence of the following error in the code flash – An error in store access • Response to an occurrence of the following error in the data flash*2 – Uncorrectable ECC error

Table 3.77 SEGCONT Register Contents (2/2)

Bit Position	Bit Name	Function
3	Reserved	This bit is read as 0. The write value should be 0.
2	ICCE	Instruction Cache Error Notification Enable The error occurred in the instruction cache is handled when the instruction cache system register, ICCTRL.ICHEMK, is set to 0 (whose value after reset is 1). For instruction cache errors, see **Section 3.2.1.2(6) (h) ICERR ─ Instruction cache error**.
1, 0	Reserved	These bits are read as 0. The write value should be 0.

Note 1. This is configured as a reserved bit for the SubCPU. It is read as 0 and the write value should be 0.

Note 2. This is the case where an uncorrectable ECC error occurs while conveyance of the ECC error signal is enabled.

Table 3.78 Error Source Codes and Handling of G3MH Core System Error Exceptions

Source Code	Error Description
10	Reserved
11	A guard error in instruction fetch and its response.
12	An error for which notification is enabled by ICCE, the second bit of SEGCONT.
13	The following errors in instruction fetch: • Bus parity error • Uncorrectable ECC error (including the case where 1-bit correction is disabled) • Address parity error (including the case where duplex bit correction is disabled)
14	An error for which notification is enabled by LSUE, the forth bit of SEGCONT.
15	An error for which notification is enabled by CFBE, the fifth bit of SEGCONT.
16	An error for which notification is enabled by LRME, the sixth bit of SEGCONT.
17	An error for which notification is enabled by GRME, the seventh bit of SEGCONT.*1
18	An error for which notification is enabled by HBPE, the eighth bit of SEGCONT.
19	An error for which notification is enabled by PBPE, the ninth bit of SEGCONT.
1A	Reserved
1B	Reserved
1C	Reserved
1D	Reserved
1E	Reserved
1F	Reserved

Note 1. This bit is reserved in the SubCPU.

(b) SEGFLAG ─ Error occurrence retention register

- This register contains the flags which indicate the state of error occurrence by each factor. Occurrence of an error sets this flag (= 1). These flags are not automatically cleared to 0.
- Writing to this register either sets or clears the values in this register.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PBPF	HBPF	GRMF*1	LRMF	CFBF	LSUF	—	ICCF	—	—
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R

Note 1. This is configured as a reserved bit for the SubCPU.

Table 3.79 SEGFLAG Register Contents

Bit Position	Bit Name	Function
15 to 10	Reserved	These bits are read as 0. The write value should be 0.
9	PBPF	The flag corresponding to bit 9 of the SEGCONT register.
8	HBPF	The flag corresponding to bit 8 of the SEGCONT register.
7	GRMF*1	The flag corresponding to bit 7 of the SEGCONT register.
6	LRMF	The flag corresponding to bit 6 of the SEGCONT register.
5	CFBF	The flag corresponding to bit 5 of the SEGCONT register.
4	LSUF	The flag corresponding to bit 4 of the SEGCONT register.
3	Reserved	This bit is read as 0. The write value should be 0.
2	ICCF	The flag corresponding to bit 2 of the SEGCONT register.
1, 0	Reserved	These bits are read as 0. The write value should be 0.

Note 1. This is configured as a reserved bit for the SubCPU. It is read as 0 and the write value should be 0.

(c) SEGADDR ─ Error source retention register (address)

This register records information of an error factor that has notified a SYSERR request (only one history is recorded). This register covers bit 4 to 9 of the SEGFLAG register, which are, PBPF, HBPF, GRMF*1, LRMF, CFBF, and LSUF. The content of this register cannot be modified while an error occurrence flag for which notification of a SYSERR request has been enabled remains set.

Note 1. GRMF is not included in the SubCPU.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Address[31:16]															
Value after Reset	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Address[15:0]															
Value after Reset	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note: ×: Undefined (retained)

Note 1. This bit cannot be modified while an error occurrence flag for which notification of a SYSERR request has been enabled remains set.

Table 3.80 SEGADDR Register Contents

Bit Position	Bit Name	Function
31 to 0	Address[31:0]	These bits retain the address at which a SYSERR source has occurred. 0 is stored if an instruction cache error (ICCE) has occurred. See the description of the ICERR register.

(3) SEG function

(a) SEG function: Notifying a SYSERR request in response to an error flag

- Setting an error flag takes precedence over clearing the same flag.
  - Simultaneous clearing operation is ignored.
- Priority of error sources to be notified
  - The priorities for error sources are based on their bit positions in the SEGFLAG register for which notification of SYSERR request has been enabled. Error factors of lower-order bits take precedence over error factors of higher-order bits.
  - The bit position of the error source is used as "SYSERR source code" to be notified.
- Conditions for starting notification of SYSERR requests
  - Setting the flags for which error notification is disabled does not start notification.
  - Setting the flags for which error notification is enabled immediately starts notification.
  - After clearing of a flag, notification is made depending on the flag state (re-arbitration).
- Stopping notification by responding to the SYSERR request
  - Stopping notification does not automatically clear the flag.
  - Notification will not be resumed until re-arbitration is performed by setting or clearing a flag.
  - If an error flag for the error source with the priority higher than that of the error source whose error flag is waiting for a response is set, the SYSERR source code to be notified may be replaced with the error source with the higher priority.

(b) SEG function: Recording error source information

- In response to an input of an error for which notification is enabled, the address at which the error has occurred is retained in the SEGADDR register.
  - No information is retained in the register by setting or clearing the error flag described in **Section 3.2.4.3(3)(a) SEG function: Notifying a SYSERR request in response to an error flag**.
  - When two or more errors have occurred at the same time, the address at which an error source with the highest priority occurred only is retained.
- While an error flag for which notification is enabled, described in **Section 3.2.4.3(3)(a) SEG function: Notifying a SYSERR request in response to an error flag**, is set, overwriting to the SEGADDR register is inhibited.
  - If inputs of error occurrence are generated sequentially, only the information of the first error source is retained.
  - To release inhibition of overwriting to the SEGADDR register, clear either SEGCONT or SEGFLAG register (or both).

(c) Supplementary notes on SYSERR exception

- Even when a SYSERR exception occurs, the value of the PSW.EBV bit is held, and the base address of the exception handler is not switched.
- Error detection in the instruction cache

  Even when an error is detected in the RAM in instruction cache, a rerun-typed SYSERR exception caused by instruction fetch does not occur. The instruction cache automatically invalidates the target entry in which an error occurred, and that entry is fetched again from the code flash. Thus execution of CPU instructions continues. Setting the ICCTRL.ICHEMK bit of the system register to 0 notifies the SEG of an error occurred in the instruction cache. For details on instruction cache errors, see **Section 3.2.1.2(6)(h), ICERR – Instruction cache error**.

3.2.4.4 Checker Core

CPU1 has the checker core for safety assurance, resulting in a highly reliable system. Monitoring the outputs from CPU1 and the checker core with the comparator all the time enables immediate detection of abnormal operations of CPU1. The CPU core, FPU, MPU, PEG, IPG, SEG, and INTC1 are duplicated by the checker core. CPU1 can also conduct a fault diagnosis test of its own comparator through a pseudo-error generated by the COMPTEST module. For details of the COMPTEST module, see **Section 29, Functional Safety**.

CAUTION

Reading of any register with an undefined value after a reset in a PE or writing to memory or a register outside the PE may cause a lockstep comparison error. The values of some program registers and system registers are undefined after a reset, so pay attention to this when saving register values on the stack in RAM.

3.2.5 Boot Control Function

This microcontroller has two startup modes for the CPUs: starting up both CPU1 (PE1) and CPU2 (PE2) simultaneously and starting up only CPU1 (PE1) while leaving CPU2 (PE2) in the non-operating state. A setting in an option byte selects the startup mode. See the description of the STARTUPPE bit of option byte 6 in **Section 35, Flash Memory**.

To start CPU2 (PE2) up from the non-operating state, use software running on CPU1 (PE1) to configure the boot control register.

Regardless of the startup mode of the CPUs, the SubCPU (PE3) in EMU3 will not be operating after release from a reset. It is started by software running on CPU1 (PE1) or CPU2 (PE2). For details, see **Section 25, Enhanced Motor Control Unit 3 (EMU3)**.

Table 3.81 Boot Control Register

Address	Size (byte)	Register Name	Symbol	Right	R/W	Operable Bit Width				Value After Reset
						1	8	16	32	
FFC5 8000$_H$	4	Boot Control Register	BOOTCTRL	—	R/W	—	√	√	√	*1

Note 1. 0000 003X$_H$ (This depends on the value set in the option byte register 6 (STARTUPPE)).

3.2.5.1 Boot Control Register ─ BOOTCTRL

This register controls startup of CPU1 (PE1) and CPU2 (PE2). Setting an effective bit of this register to 1 activates the corresponding CPU and the CPU starts fetching instructions. After a reset, the CPU waits without any operation if the corresponding bit is not set. The BC2 bit is invalid for the RH850/C1M-A1. A setting in an option byte determines the initial value of BC2. Writing 0 to these bits while the value is 1 is ignored.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	BC2	—	—
Value after Reset	0	0	0	0	0	0	0	0	0	0	1	1	1	X	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R

Table 3.82 Boot Control Register Contents

Bit Position	Bit Name	Function
31 to 6	Reserved	These bits are read as 0. The write value should be 0.
5 to 3	Reserved	These bits are read as 1. The write value should be 1.
2	BC2	Boot Control 2 Setting this bit to 1 starts up CPU2 (PE2). 0: CPU2 (PE2) is not stated. 1: CPU2 (PE2) is started. Writing 0 while this bit is 1 is ignored.
1	Reserved	This bit is read as 1. The write value should be 1.
0	Reserved	This bit is read as 0. The write value should be 0.

3.3 Inter-CPU Functions

3.3.1 Processor Element Identifier

The PEID, each processor element ID number, can be read from the PEID field in the HTCFG0 register. Which CPU core performs a specific program can be checked by referring to the PEID. The following shows the PEIDs of this product.

CPU Core	PEID
CPU1 (PE1)	001_B
CPU2 (PE2)	010_B
SubCPU (PE3) in EMU3	011_B

3.3.2 Inter-Processor Interrupt

CPU1 (PE1) and CPU2 (PE2) have the IPIR register as their own peripheral devices. Setting of the IPIR register enables an EI level interrupt request from a PE to another PE. For details, see **3.2.3 Inter-Processor Interrupts**.

The SubCPU in EMU3 is not provided with this function.

3.3.3 Exclusive Function

CPU1 and CPU2 can use the local RAM for their own, the global RAM, and the exclusive control registers (MEV) as a resource for exclusive control. Atomic operation instructions of LDL/STC, CAXI, SET1, CLR1, and NOT1 can be performed for the local RAM for each and the global RAM. For the exclusive control registers (MEV), the instructions of CAXI, SET1, CLR1, and NOT1 can be performed. Note that atomic operations are not performed for the LD and ST instructions even though they can access those resources.

CAUTIONS

1. Atomic guarantee is provided in access by the SubCPU in EMU3 to its own local RAM by the CAXI or bit operation instruction. However, atomic guarantee is not provided in the case of access by the SubCPU to the local RAM of CPU1 and CPU2 or to the global RAM.
2. Atomic operations succeeds in access by the SubCPU in EMU3 to its own local RAM by LDL or STC instruction. However, the operations always fail in access to the local RAM of CPU1 and CPU2 or the global RAM.
3. Atomic guarantee is not provided in access by CPU1 or CPU2 to the local RAM of the SubCPU by the CAXI instruction.
4. Atomic operation in access by CPU1 or CPU2 to the local RAM of the SubCPU by the LDL or STC instruction always fails.
5. The SubCPU in EMU3 does not have access to the MEV (mutual exclusion variable registers).

3.3.3.1 Exclusive Control Registers (MEV)

These registers support exclusive control for variables shared among PEs (common resources).
(MEV: Mutual Exclusion Variable Register)

- Thirty-two 32-bit MEV registers are included.
- Each register is accessible in 1-, 8-, 16-, and 32-bit units.
- These registers are accessible by CPU1 (PE1) and CPU2 (PE2). Accessing these registers by the SubCPU in EMU3 is not possible.
- Atomic operation instructions of CAXI, SET1, CLR1 and NOT1 are available.

CPU1 (PE1) and CPU2 (PE2) each has an independent access path for the MEV registers. Therefore, when CPU1 (PE1) and CPU2 (PE2) each accesses different MEV registers, they do not need to wait for access. When they access the same MEV register, however, waiting for access is required.

Table 3.83 List of Registers (Base Address: FFFE EC00$_H$)

Register Symbol	Register Name	R/W	Value After Reset	Access Size				Offset Address
				1	8	16	32	
G0MEV0	Exclusive control register 0	R/W	0000 0000$_H$	√	√	√	√	+00$_H$
G0MEV1	Exclusive control register 1	R/W	0000 0000$_H$	√	√	√	√	+04$_H$
G0MEV2	Exclusive control register 2	R/W	0000 0000$_H$	√	√	√	√	+08$_H$
G0MEV3	Exclusive control register 3	R/W	0000 0000$_H$	√	√	√	√	+0C$_H$
:	:	:	:	:	:	:	:	:
G0MEV31	Exclusive control register 31	R/W	0000 0000$_H$	√	√	√	√	+7C$_H$

3.3.3.2 Operations of the LDL.W and STC.W Instructions

The LDL.W and STC.W instructions can be used to correctly handle memory update in a multicore system by enabling atomic read-modify-write processing. The operations of the LDL.W and STC.W instructions are described below:

For the operations of LDL.W and STC.W instructions, refer to *RH850/G3MH User's Manual: Software*.

- Link generation
  The CPU can generate links for both the local RAM and global RAM. If the LDL.W instruction is executed to read the target, the link is generated with the link flag set. The following two types of link flags are provided:

(1) One for the own local RAM

(2) One for other than (1)

These link flags are generated independently to each other. If the CPU generates a link to its own local RAM, for example, the link will not disappear even if another link flag is generated in the global RAM, for example, by the LDL.W instruction.

- Successful store: Storing proceeds only for the links which has been generated and the applicable STC.W instruction has been executed to the link.
- Storing failure: Storing does not proceed while the link is lost even if the STC.W instruction is executed. Storing does not proceed as well when the STC.W instruction that does not correspond to the link is executed.
- Condition for successful storing: The STC.W instruction is determined that it corresponds to the link if the following condition is satisfied:
  - The address of the STC.W instruction is the same as the address of the LDL.W instruction that generated the link.
- Link lost: When either of the following occurs, the link flag is cleared, and the link is lost:

  When either of the following events occurs in the CPU that generated the link:
  - The CLL instruction is executed.
  - The STC.W instruction is executed. The corresponding link ((1) or (2) above) is lost regardless of whether the store succeeded or failed.
  - Occurrence of all EI level interrupts/exceptions
  - Occurrence of all FE level interrupts/exceptions
  - A restore instruction (FERET or EIRET) from the exception is executed. The link flags ((1) or (2) above) will be all cleared.
  - A EST instruction is executed.
  - Multiple LDL.W instructions are executed in succession for the same type of the link flag ((1) or (2) above). The link which has been generated by the preceding LDL.W instruction will be lost. Do not execute such processing.
  - A store operation other than the STC.W instruction is executed for the range of 32-byte order address containing the address*1 for which the link is generated. Do not execute such processing.
  - When a store operation (including execution of the STC.W instruction) is executed by another bus master for the range of 32-byte order address containing the address*1 for which the link is generated. The corresponding link is lost.

Note 1. This indicates the address that corresponds to 27 higher-order bits of the address of the link.

  - When the STC.W instruction is successful, an atomic read-modify-write processing has been executed by the LDL.W and STS.W instructions.

3.4 Notes on Usage

3.4.1 Synchronizing Completion of Store Instruction and Generation of Subsequent Instruction

When a control register is updated by a store instruction, there is a time lag from execution of the instruction by the CPU to actual updating of the control register. Therefore, appropriate processing for synchronization is required to ensure that control registers reflect updated values before execution of subsequent instructions. The processing for synchronization is shown below.

For details of the procedure regarding updating of the system registers by the LDSR instruction and synchronization with subsequent instructions, refer to *APPENDIX A, HAZARD RESOLUTION PROCEDURE FOR SYSTEM REGISTERS* in the *RH850G3MH User's Manual: Software*.

3.4.1.1 When updated results in the control registers are reflected in the implementation of a subsequent instruction:

Example 1: This includes the following case: an interrupt is enabled by implementation of an EI instruction after an interrupt request is cleared by access from the control register in the INTC2 and the peripheral circuits.

Proceed as follows in this case.

(1) Store instruction to update a control register (ST.W, etc.)

(2) Dummy reading of the above-mentioned control register (LD.W etc.)*1

(3) SYNCP

(4) Subsequent instruction (EI, etc.)

Example 2: If an access to another control register (control register B) is needed after a control register (control register A) has been completely updated, execute the similar processing. This includes the following cases: different peripheral devices are linked, or the interrupt mask for INTC is cleared after the peripheral device is set. Note that this processing is not required if the control registers A and B belong to the same peripheral group. For details on the peripheral groups, see **Section 3.1.2, Configuration of Peripheral Groups**.

(1) Store instruction to update the control register A (ST.W, etc.)

(2) Dummy read of the above-mentioned control register (LD.W etc.)*1

(3) SYNCP

(4) Store instruction to access the control register B (ST.W, LD.W, etc.)

The similar processing is also required when an access to a memory or control register to be protected is started after a safety function (such as some kind of memory protection and ECC) has been completely set up.

Note 1. Dummy reading of any register of the same peripheral group can be used instead.

3.4.1.2 When the updated results in the control registers and memories are reflected in the instruction fetch of a subsequent instruction:

(a) If you wish to write an instruction to the RAM and then branch to the RAM to execute the written instruction, handle this as follows.

(1) Store instruction to update a memory (ST.W, etc.)

(2) Dummy read of the above-mentioned memory (LD.W, etc.)

(3) SYNCP

(4) SYNCI

(5) Subsequent instruction (branch instruction, etc.)

(b) When branching to a target memory after waiting for the completion of updating the control registers for memory protection and ECC, handle this as follows.

(1) Store instruction to update a control register (ST.W, etc.)

(2) Dummy read of the control register (LD.W, etc.)

(3) SYNCP

(4) SYNCI

(5) Subsequent instruction (branch instruction, etc.)

3.4.1.3 When switching the code flash memory area:

In this case, refer to *(7) Updating the FCUFAREA Register, in section 9, Usage Notes, in the RH850/C1M-A Flash Memory User's Manual: Hardware Interface*.

3.4.2 Access to Registers by Bit-Manipulation Instructions

Processing of writing by bit-manipulation instructions consists of atomic read-modify-write processing in 8-bit units. Thus, access by a bit-manipulation instruction is only possible for registers for which reading and writing in 8-bit units is possible. If a register includes multiple flag bits, the read-modify-write operation may lead to the clearing of flags that were not actually targets for clearing.

3.4.3 Ensure Coherency after Rewriting the Code Flash

The CPU is equipped with the valid instruction cache and data buffer for the code flash area. Therefore, clear the instruction cache and data buffer to ensure coherency after rewriting the code flash by self-programming. The instruction cache is cleared by the ICCTRL register, and the data buffer is cleared by the CDBCR register.

3.4.4 Overwriting Context when Acknowledging Multiple Exceptions

Exceptions may be acknowledged regardless of the states of the ID or NP bit of the PSW register. This depends on the type of exception. When multiple exceptions occur, the contents of the system registers which hold the context information are overwritten. Regarding the conditions for acknowledging exceptions from each source and the possibility of return and recovery, see the list of exception sources in the *RH850G3MH User's Manual: Software*.

3.4.5 Usage Notes on Prefetching

CPU executes speculative instruction fetching from locations later than the current value of the program counter to maintain the throughput of instruction fetches. Reading from memory due to such prefetching may proceed even from locations to which instruction codes have not been assigned (note 1 in **Figure 3.6**). Please note the following. The CPU does not execute values read in such cases.

These notes apply to instruction fetching from memory in general.

- Occurrence of ECC errors due to values in memory being undefined

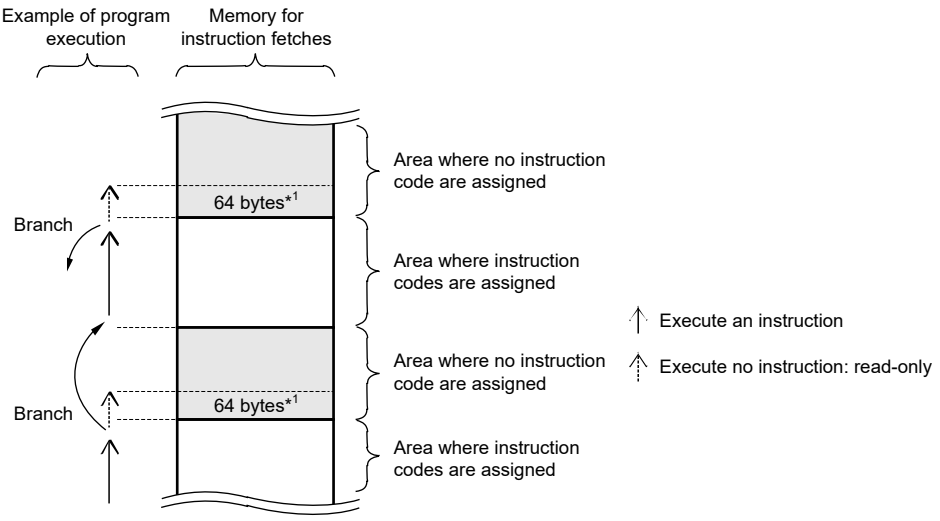
  This prefetching may lead to an ECC error in case of reading from the code flash memory after it has been erased or from the local RAM or global RAM before initialization. When instruction codes are assigned to memory, initialize said area with values as desired (note 1 in **Figure 3.6**).

- Detection of illegal access by the GRG or IPG

  The GRG or IPG may detect such prefetching as illegal access. To prevent prefetching being detected as an illegal access, do not allow any region of overlap area with said areas (note 1 in **Figure 3.6**) and areas to which access is prohibited by the GRG or IPG. Reading from an area protected by the MPU does not cause a memory protection exception.

- Access to Access Prohibited Area

  Assign instruction codes to memory without allowing any overlap between said area (note 1 in **Figure 3.6**) and an access-prohibited area.



Note 1. Area from final address of instruction code to final address + 64 bytes. Even if instruction codes are not placed in this area, requests for prefetching may lead to reading from memory at the time of branching.

Figure 3.6 Area that Requires Attention Regarding Prefetching

Section 4 Address Space

4.1 RH850/C1M-A Address Space

4.1.1 Address Space (C1M-A2)

Table 4.1 shows the address space of the RH850/C1M-A2.

Do not access an address with which no register is mapped in the on-chip I/O register space. Do not access any address that is not specified in **Table 4.1** and any reserved area. If an unspecified address or reserved area is accessed, operation is not guaranteed.

Table 4.1 Address Space (C1M-A2)

Address	Address Space Type	Size
0000 0000$_H$ to 001F FFFF$_H$ (0001 7000$_H$ to 0001 7FFF$_H$)	On-chip ROM (user area read, bank A) (FCU firmware area (Map is switched by the FCUFAREA register))*3	2.0 MB (4 KB)
0020 0000$_H$ to 003F FFFF$_H$	On-chip ROM (user area read, bank B)	2.0 MB
0040 0000$_H$ to 00FF FFFF$_H$	Reserved area	—
0100 0000$_H$ to 0100 7FFF$_H$	On-chip ROM (user boot area read)	32 KB
0100 8000$_H$ to 0103 6FFF$_H$	Reserved area	—
0103 7000$_H$ to 0103 7FFF$_H$	FCU firmware area*3	4 KB
0103 8000$_H$ to FBFF FFFF$_H$	Reserved area	—
FC00 0000$_H$ to FC07 FFFF$_H$	EMU on-chip I/O register*4	512 KB
FC08 0000$_H$ to FE7E FFFF$_H$	Reserved area	—
FE7F 0000$_H$ to FE7F FFFF$_H$	On-chip RAM (local RAM EMU SubCPU area)	64 KB
FE80 0000$_H$ to FE9E FFFF$_H$	Reserved area	—
FE9F 0000$_H$ to FE9F FFFF$_H$	On-chip RAM (local RAM CPU2 area)	64 KB
FEA0 0000$_H$ to FEBE FFFF$_H$	Reserved area	—
FEBF 0000$_H$ to FEBF FFFF$_H$	On-chip RAM (local RAM CPU1 area)	64 KB
FEC0 0000$_H$ to FEDE FFFF$_H$	Reserved area	—
FEDF 0000$_H$ to FEDF FFFF$_H$	On-chip RAM (local RAM self area*1)	64 KB
FEE0 0000$_H$ to FEEE FFFF$_H$	Reserved area	—
FEEF 0000$_H$ to FEEF FFFF$_H$	On-chip RAM (global RAM area bank A)	64 KB
FEF0 0000$_H$ to FEF0 FFFF$_H$	On-chip RAM (global RAM area bank B)	64 KB
FEF1 0000$_H$ to FEFF FFFF$_H$	Reserved area	—
FF00 0000$_H$ to FFFD FFFF$_H$ (FF20 0000$_H$ to FF20 FFFF$_H$) (FFA1 2000$_H$ to FFA1 2FFF$_H$)	On-chip I/O register (Data Flash (read/write)) (FCU RAM area)	16 MB to 128 KB (64 KB) (4 KB)
FFFE 0000$_H$ to FFFE DFFF$_H$	Reserved area	—
FFFE E000$_H$ to FFFE FFFF$_H$	On-chip I/O register (self area*2)	8 KB
FFFF 0000$_H$ to FFFF 4FFF$_H$	Reserved area	
FFFF 5000$_H$ to FFFF FFFF$_H$	On-chip I/O register	44 KB

Note 1. The local RAM self area is an address area for access by each of the CPUs (CPU1, CPU2, and the SubCPU) to its own local RAM.

Note 2. The internal I/O register area (“self area”) is an address range for access by each of the CPUs (CPU1, CPU2, and the SubCPU) to its own internal I/O registers (registers for peripherals of the given CPU).

Note 3. For details, see *RH850/C1M-A Flash Memory User’s Manual: Hardware Interface.*

Note 4. This area is accessible only by the SubCPU of EMU3.

4.1.2 Address Space (C1M-A1)

Table 4.2 shows the address space of the RH850/C1M-A1.

Do not access an address with which no register is mapped in the on-chip I/O register space. Do not access any address that is not specified in **Table 4.2** and any reserved area. If an unspecified address or reserved area is accessed, operation is not guaranteed.

Table 4.2 Address Space (C1M-A1)

Address	Address Space Type	Size
0000 0000$_H$ to 001F FFFF$_H$ (0001 7000$_H$ to 0001 7FFF$_H$)	On-chip ROM (user area read, bank A) (FCU firmware area (Map is switched by the FCUFAREA register))*3	2.0 MB (4 KB)
0020 0000$_H$ to 00FF FFFF$_H$	Reserved area	—
0100 0000$_H$ to 0100 7FFF$_H$	On-chip ROM (user boot area read)	32 KB
0100 8000$_H$ to 0103 6FFF$_H$	Reserved area	—
0103 7000$_H$ to 0103 7FFF$_H$	FCU firmware area*3	4 KB
0103 8000$_H$ to FBFF FFFF$_H$	Reserved area	—
FC00 0000$_H$ to FC07 FFFF$_H$	EMU on-chip I/O register*4	512 KB
FC08 0000$_H$ to FE7E FFFF$_H$	Reserved area	—
FE7F 0000$_H$ to FE7F FFFF$_H$	On-chip RAM (local RAM EMU SubCPU area)	64 KB
FE80 0000$_H$ to FEBE FFFF$_H$	Reserved area	—
FEBF 0000$_H$ to FEBF FFFF$_H$	On-chip RAM (local RAM CPU1 area)	64 KB
FEC0 0000$_H$ to FEDE FFFF$_H$	Reserved area	—
FEDF 0000$_H$ to FEDF FFFF$_H$	On-chip RAM (local RAM self area*1)	64 KB
FEE0 0000$_H$ to FEEE FFFF$_H$	Reserved area	—
FEEF 0000$_H$ to FEEF FFFF$_H$	On-chip RAM (global RAM area bank A)	64 KB
FEF00000$_H$ to FEFF FFFF$_H$	Reserved area	—
FF00 0000$_H$ to FFFD FFFF$_H$ (FF20 0000$_H$ to FF20 FFFF$_H$) (FFA1 2000$_H$ to FFA1 2FFF$_H$)	On-chip I/O register (Data Flash (read/write)) (FCU RAM area)	16 MB to 128 KB (64 KB) (4 KB)
FFFE 0000$_H$ to FFFE DFFF$_H$	Reserved area	—
FFFE E000$_H$ to FFFE FFFF$_H$	On-chip I/O register (self area*2)	8 KB
FFFF 0000$_H$ to FFFF 4FFF$_H$	Reserved area	
FFFF 5000$_H$ to FFFF FFFF$_H$	On-chip I/O register	44 KB

Note 1. The local RAM self area is an address area for access by each of the CPUs (CPU1 and the SubCPU) to its own local RAM.

Note 2. The internal I/O register area (“self area”) is an address range for access by each of the CPUs (CPU1 and the SubCPU) to its own internal I/O registers (registers for peripherals of the given CPU).

Note 3. For details, see *RH850/C1M-A Flash Memory User’s Manual: Hardware Interface.*

Note 4. This area is accessible only by the SubCPU of EMU3.

4.1.3 Address Space Viewed from Each Bus Master

Figure 4.1 (C1M-A2) and **Figure 4.2** (C1M-A1) show address spaces viewed from each bus master.

4.1.3.1 Space in which Instructions can be Fetched

1. Instructions of CPU1, CPU2 (C1M-A2 only), and the SubCPU of EMU can be fetched from the on-chip ROM and on-chip RAM (local RAM and global RAM areas).
2. The reset vector (the initial value of RBASE) of CPU1, CPU2 (C1M-A2 only), and the SubCPU of EMU are as follows:
   - When the startup area is the user boot area, its head address is 0100 0000_H.
   - When the startup area is the user area, its head address is 0000 0000_H.

4.1.3.2 Data Space Accessible by CPU1

See **Figure 4.1** and **Figure 4.2** for the accessible spaces from CPU1.

4.1.3.3 Data Space Accessible by CPU2 (C1M-A2 only)

See **Figure 4.1** and **Figure 4.2** for the accessible spaces from CPU2 (applicable to C1M-A2 only).

4.1.3.4 Data Space Accessible by the SubCPU of EMU

See **Figure 4.1** and **Figure 4.2** for the accessible spaces from the SubCPU of EMU.

4.1.3.5 Data Space Accessible by DMA (DMAC, DTS)

See **Figure 4.1** and **Figure 4.2** for the accessible spaces from the DMA.

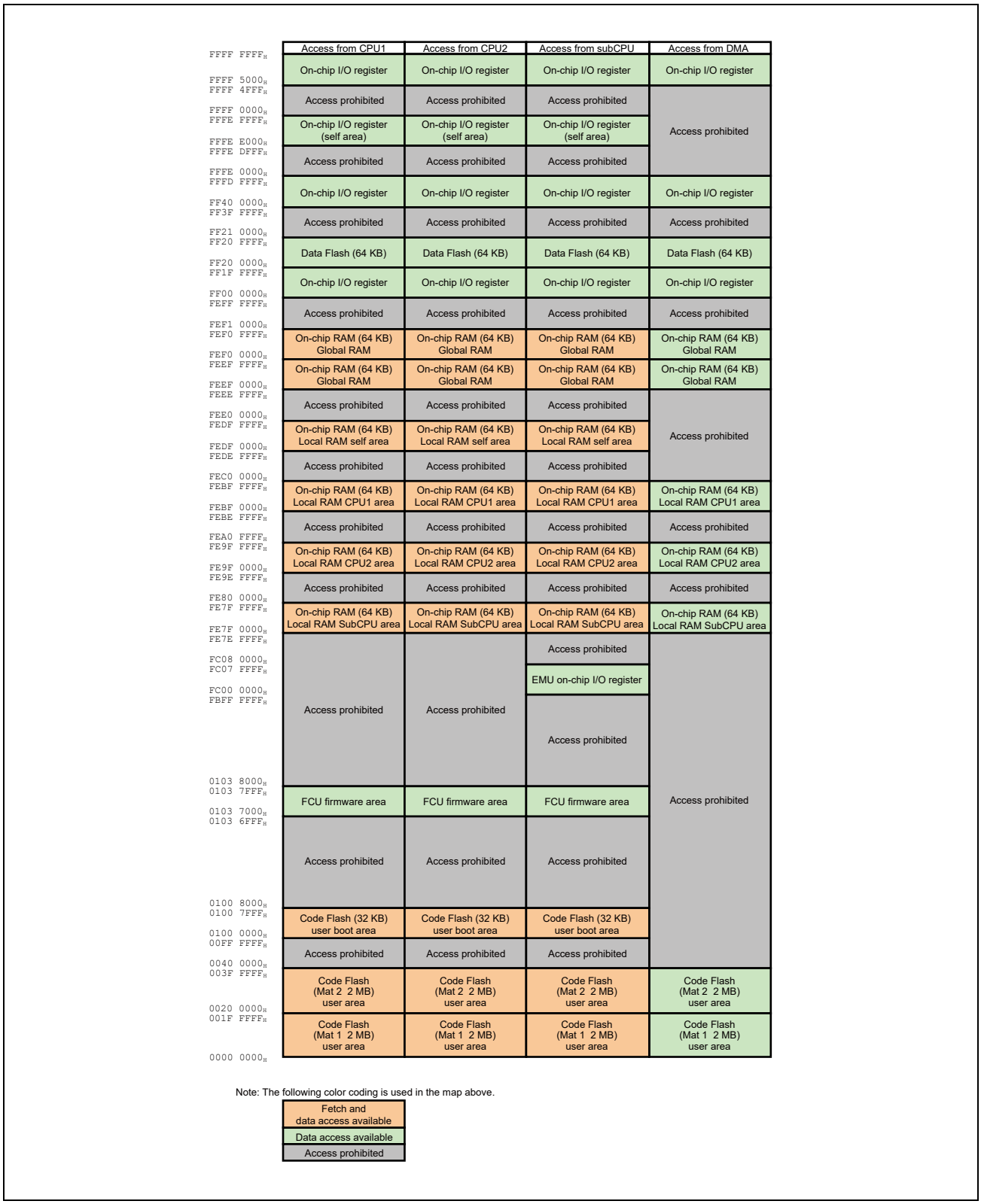


Figure 4.1 Address Space Viewed from Each Bus Master (C1M-A2)

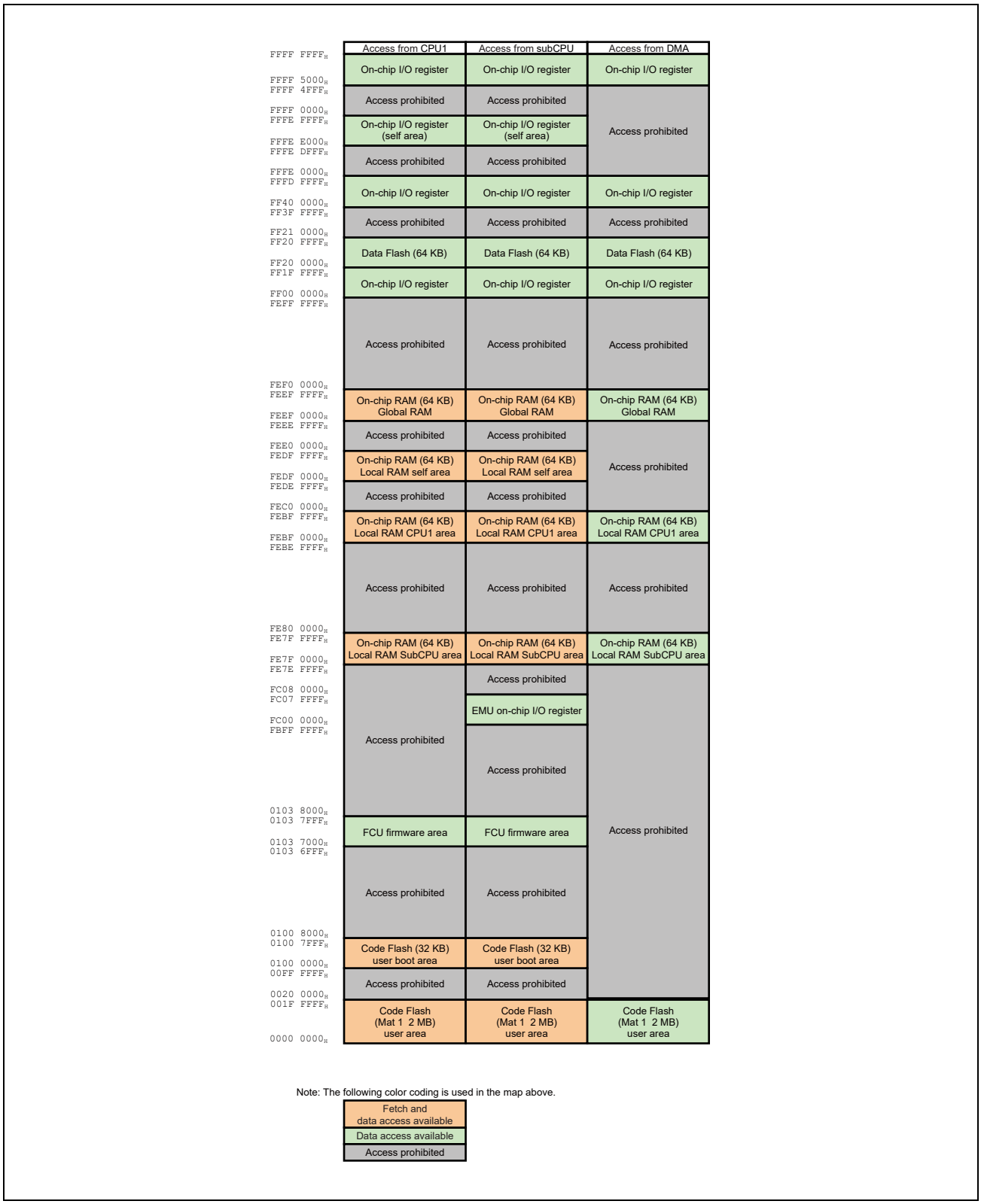


Figure 4.2 Address Space Viewed from Each Bus Master (C1M-A1)

4.1.4 Global RAM Area

Figure 4.3 and **Figure 4.4** show the address map of the global RAM of C1M-A2 and C1M-A1, respectively. The global RAM of C1M-A2 is divided into two, bank A and bank B. Different banks can be accessed concurrently.

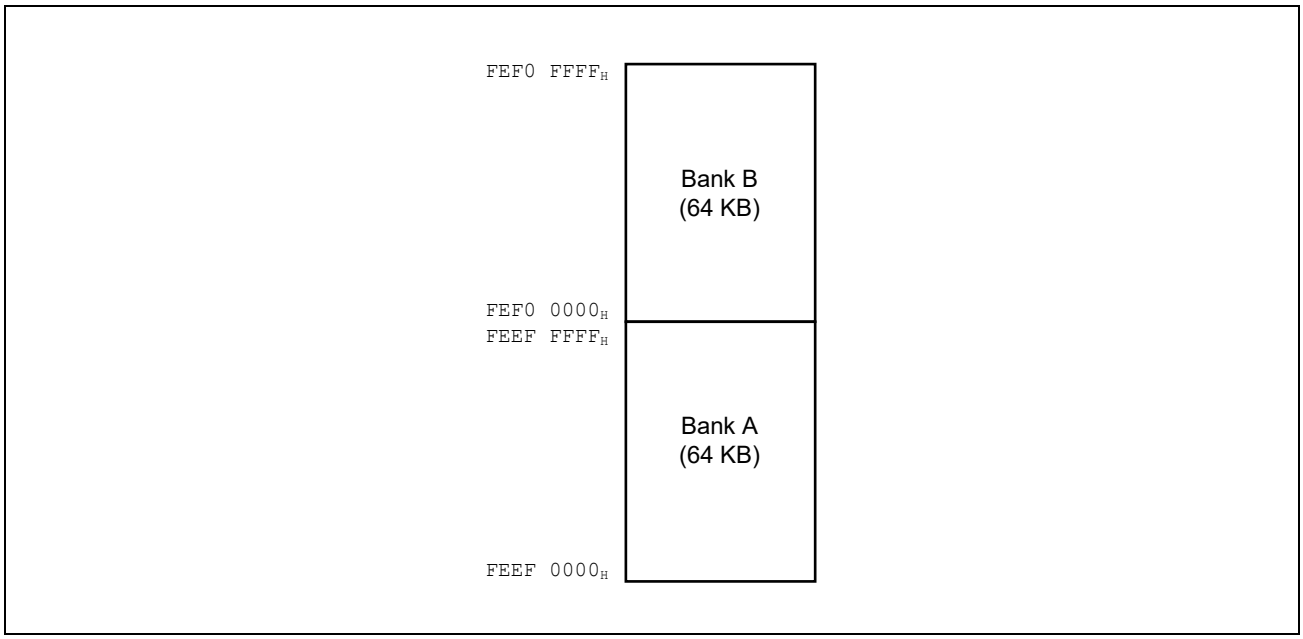


Figure 4.3 GRAM Bank Area (C1M-A2)

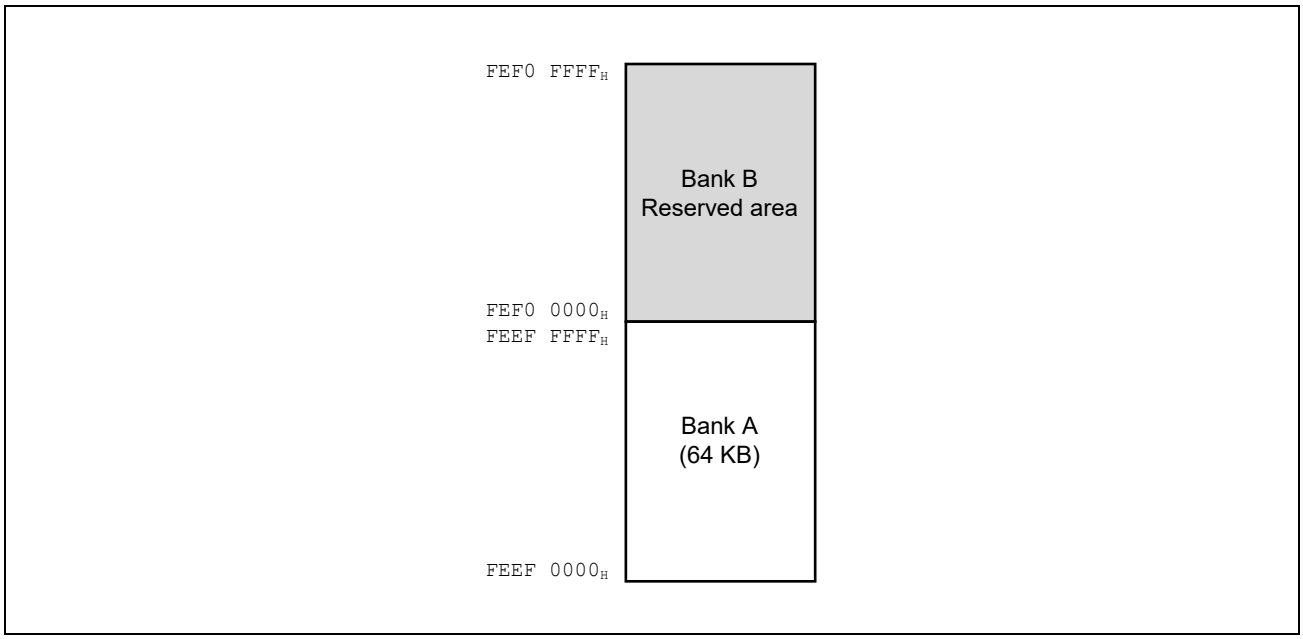
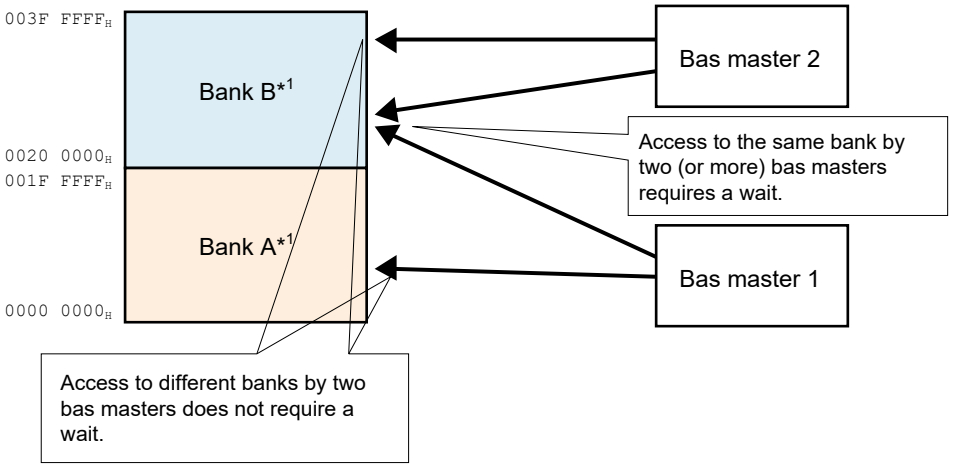


Figure 4.4 GRAM Bank Area (C1M-A1)

4.2 Code Flash Space

4.2.1 Parallel Access to the User Area (C1M-A2)

The user area of the code flash memory is in a two-mat configuration.These two mats are respectively assigned to as bank A and bank B. If one bus master accesses to bank A and another accesses bank B, they can do so without a wait for arbitration of contention for one access path. When multiple bus masters access the same bank, bank A or bank B, access is arbitrated in round-robin fashion.



Note 1. See **Table 4.1** and **Table 4.2** for the memory size of each bank.

Figure 4.5 Access Competition between Bank A and Bank B

Section 5 Operating Mode

5.1 Features

This LSI has three mode pins to determine the operating mode (MD1, MD0, and FLMODE).

5.2 Operating Mode

This LSI has multiple operating modes, which can be selected with the three pins (MD1, MD0, and FLMODE) and the setting of STMSEL1/STMSEL0 in option byte 0. For the procedures to set STMSEL1/STMSEL0, see **Section 35, Flash Memory**. **Table 5.1** shows the list of the operating modes.

Table 5.1 Selection of Operating Mode

Value Set in the Pin			Value Set in the Option Byte Register 0		Operating Mode	Startup Area	Type of IF*1	Remark
MD1	MD0	FLMODE	STMSEL1	STMSEL0				
0	0	0	0	0	User boot mode	User area	The interface can be selected by OPBT2 in the option byte area. For details, see **Section 35.9.2 OPBT2 — Option Byte 2 Register**.	On-chip debug is available.
0	0	0	0	1	User boot mode	User boot area		
0	0	0	1	X	Serial programming mode	Boot area	Writer I/F (2-line UART)	Serial programming is available.
0	0	1	X	X	Boundary scan mode	—	JTAG	Boundary scan is available.
0	1	0	X	X	Serial programming mode	Boot area	Writer I/F (2-line UART)	Serial programming is available.
0	1	1	X	X	Serial programming mode	Boot area	Writer I/F (3-line clock synchronous connection)	Serial programming is available.

Note: X = Don't care

Note 1. See **Section 2.4.3, Pin State** for the functions and states of pins when each interface is selected.

5.2.1 User Boot Mode

After release from the reset state, instruction fetch is carried out from the user boot area or the user area.

5.2.2 Serial Programming Mode

After release from the reset state, the LSI boots up from the on-chip boot program and starts connection in the specified transmission method. For details, see **Section 35, Flash Memory**.

5.2.3 Boundary Scan Mode

This is a mode to use the boundary scan function in accordance with the standard of IEEE1149.1. For details, see **Section 38, Boundary Scan**.

Section 6 Interrupt

The interrupt controllers (INTC) judge the priority levels of interrupt sources and control interrupt requests for the CPU. The INTC has registers for setting the order of priority for all of the interrupts, and processing of the interrupts is in accord with the order of priority set by the user in these registers.

6.1 Overview

- Distribution of interrupt requests from one source to more than one core is supported.
  - Interrupt requests from one source are distributable to more than one CPU core (specifically, to CPU1 and CPU2)
  - Target interrupt sources: 1 FE level interrupt and EI level interrupt C1M-A1: 21 sources/C1M-A2: 26 sources
  - Some interrupt sources can be distributed to the SubCPU in EMU3 simultaneously.
- Interrupt sources
  - FE level interrupts
    ECM interrupt (FEINT) (1 source)
  - EI level interrupts (maskable) (EIINT)
    High-speed interrupts (EIINT0 to EIINT31) C1M-A1: 21 sources/C1M-A2: 26 sources
    - Inter-processor interrupts
    - ECM interrupts
    - External interrupts (IRQ)
    - Software interrupts (SINT)
    - Interrupts from the fixed-period timer (OSTM)
    - DMA error notification interrupts
  - Low-speed interrupts (EIINT32 to EIINT383) C1M-A1: 221 sources/C1M-A2: 287 sources
    - Interrupts from timers
    - Transfer-related interrupts
    - Interrupts from the ADC
    - Interrupts from the DMAC and DTS
    - ICU-S interrupt
- Able to set up the order of priority for interrupts
  384 interrupt control registers are provided so that one of 16 priority levels is assignable to the IRQ (external) and other maskable interrupts thus placing each of the interrupt requests in an order of priority.
- Sensing of external interrupts (IRQ)For IRQ sources, the low level, the high level, falling edges, and rising edges are selectable.
- Two ways to specify the addresses of the interrupt handlers:
  through direct branching or with reference to a table.
- Inter-processor interrupts
  High-speed interrupts between CPU1 and CPU2 are available.
- Software interrupts (SINT)
  The software interrupt registers are accessible from programs and can be used to generate changes to the order of priority as desired.
- Sharing interrupt sources
  Exception handler addresses are reduced by merging multiple interrupt sources.

Interrupts are controlled by the following interrupt controllers.

- INTC1
  CPU1, CPU2, and the SubCPU in EMU3 have their own interrupt controllers.
  Each CPU accesses to the INTC1 register that corresponds to respective CPUs.
  INTC1 controls high-speed interrupts, and has the following functions:
  - Priority order setting
  - Interrupt mask setting
- INTC2
  INTC2 is a common interrupt controller that CPU1, CPU2, and the SubCPU in EMU3 share.
  INTC2 controls low-speed interrupts, and has the following functions:
  - Priority order setting
  - Interrupt mask setting
  - Binding setting

6.2 Register Specifications

The INTC contains the registers listed in the table below. These registers are used to specify the interrupt priority and control the detection of external interrupt input signals.

6.2.1 Configuration of Registers

Table 6.1 Interrupt Control

Module Name	Address	Register Symbol	Register Name	R/W	Value after Reset
INTC1 (EIC0 to EIC31)	FFFE EA00$_H$ to FFFE EA3E$_H$ (EIC0 to EIC31)	EICn*1	EI level interrupt control register	R/W	008F$_H$*2 808F$_H$*3
INTC2 (EIC32 to EIC383)	FFFF B040$_H$ to FFFF B2FE$_H$ (EIC32 to EIC383)				
INTC1 (IMR0)	FFFE EAF0$_H$ (IMR0)	IMRn*4	EI level interrupt mask register	R/W	FFFF FFFF$_H$
INTC2 (IMR1 to IMR11)	FFFF B404$_H$ to FFFF B42C$_H$ (IMR1 to IMR11)				
INTC1 (EIBD0 to EIBD031)	FFFE EB00$_H$ to FFFE EB7C$_H$ (EIBD0 to EIBD31)	EIBDn*5	EI level interrupt bind register	R/W	*6
INTC2 (EIBD32 to EIBD383)	FFFF B880$_H$ to FFFF BDFC$_H$ (EIBD32 to EIBD383)				

Note 1. n = 0 to 383

Note 2. This is the case when an edge is detected.

Note 3. This is the case when a level signal is detected.

Note 4. n = 0 to 11

Note 5. n = 0 to 383

Note 6. n = 0 to 31: Same as the PEID bits
n = 32 to 383: 0000 0001$_H$

Among the registers shown in **Table 6.1**, the EIC0 to EIC31, IMR0, and EIBD0 to EIBD31 are located in INTC1 of the CPU Peripheral area included in each CPU. Each register of these only can be accessed from CPU1, CPU2, or the SubCPU in EMU3 which includes it. Writing is only possible in supervisor mode (PSW.UM = 0).
Of the registers listed in **Table 6.1**, EIC32 to EIC383, IMR1 to IMR11, and EIBD32 to EIBD383 are located in INTC2 in Peripheral Group 0. Writing to these registers is only possible in supervisor mode for CPU bound to EIBDn (n = 32 to 383). When writing to IMR1 to IMR11, only the bits corresponding to the conditions described above are overwritten; other bits are not updated.

In the register areas listed in **Table 6.1**, the values of those listed as reserved for the given channel numbers in **Table 6.11** must retain their initial values.

Table 6.2 External Interrupts and Software Interrupts

Module Name	Address	Register Symbol	Register Name	R/W	Value after Reset
EINT	FFC0 0010$_H$	EXINTCTL	External interrupt control register	R/W	0000$_H$
EINT	FFC0 0014$_H$	EXINTSTR	External interrupt status register	R	00$_H$
EINT	FFC0 0018$_H$	EXINTSTC	External interrupt status clear register	W	00$_H$
EINT	FFC0 0020$_H$	SINTR0	Software interrupt register	R/W	00$_H$
EINT	FFC0 0024$_H$	SINTR1	Software interrupt register 1	R/W	00$_H$
EINT	FFC0 0028$_H$	SINTR2	Software interrupt register 2	R/W	00$_H$
EINT	FFC0 002C$_H$	SINTR3	Software interrupt register 3	R/W	00$_H$
EINT	FFC0 0030$_H$	SINTR4	Software interrupt register 4	R/W	00$_H$
EINT	FFC0 0034$_H$	SINTR5	Software interrupt register 5	R/W	00$_H$
EINT	FFC0 0038$_H$	SINTR6	Software interrupt register 6	R/W	00$_H$
EINT	FFC0 003C$_H$	SINTR7	Software interrupt register 7	R/W	00$_H$
EMUEINT	FFC0 1020$_H$	ESINTR0	EMU software interrupt register	R/W	00$_H$
EMUEINT	FFC0 1024$_H$	ESINTR1	EMU software interrupt register 1	R/W	00$_H$
EMUEINT	FFC0 1028$_H$	ESINTR2	EMU software interrupt register 2	R/W	00$_H$
EMUEINT	FFC0 102C$_H$	ESINTR3	EMU software interrupt register 3	R/W	00$_H$
EMUEINT	FFC0 1030$_H$	ESINTR4	EMU software interrupt register 4	R/W	00$_H$
EMUEINT	FFC0 1034$_H$	ESINTR5	EMU software interrupt register 5	R/W	00$_H$
EMUEINT	FFC0 1038$_H$	ESINTR6	EMU software interrupt register 6	R/W	00$_H$
EMUEINT	FFC0 103C$_H$	ESINTR7	EMU software interrupt register 7	R/W	00$_H$

Table 6.3 Interrupt Merge Functions

Module Name	Address	Register Symbol	Register Name	R/W	Value after reset
INTIF	FFF9 8000$_H$	PINT0	Peripheral interrupt status register 0	R	0000 0000$_H$
INTIF	FFF9 8004$_H$	PINT1	Peripheral interrupt status register 1	R	0000 0000$_H$
INTIF	FFF9 8008$_H$	PINT2	Peripheral interrupt status register 2	R	0000 0000$_H$
INTIF	FFF9 800C$_H$	PINT3	Peripheral interrupt status register 3	R	0000 0000$_H$
INTIF	FFF9 8010$_H$	PINT4	Peripheral interrupt status register 4	R	0000 0000$_H$
INTIF	FFF9 8014$_H$	PINT5	Peripheral interrupt status register 5	R	0000 0000$_H$
INTIF	FFF9 8018$_H$	PINT6	Peripheral interrupt status register 6	R	0000 0000$_H$
INTIF	FFF9 801C$_H$	PINT7	Peripheral interrupt status register 7	R	0000 0000$_H$
INTIF	FFF9 8020$_H$	PINTCLR0	Peripheral interrupt status clear register 0	W	0000 0000$_H$
INTIF	FFF9 8024$_H$	PINTCLR1	Peripheral interrupt status clear register 1	W	0000 0000$_H$
INTIF	FFF9 8028$_H$	PINTCLR2	Peripheral interrupt status clear register 2	W	0000 0000$_H$
INTIF	FFF9 802C$_H$	PINTCLR3	Peripheral interrupt status clear register 3	W	0000 0000$_H$
INTIF	FFF9 8030$_H$	PINTCLR4	Peripheral interrupt status clear register 4	W	0000 0000$_H$
INTIF	FFF9 8034$_H$	PINTCLR5	Peripheral interrupt status clear register 5	W	0000 0000$_H$
INTIF	FFF9 8038$_H$	PINTCLR6	Peripheral interrupt status clear register 6	W	0000 0000$_H$
INTIF	FFF9 803C$_H$	PINTCLR7	Peripheral interrupt status clear register 7	W	0000 0000$_H$

6.2.2 EI Level Interrupt Control Registers 0 to 383 (EIC0 to EIC383)

One of these registers is provided for every EI level interrupt source. The registers are used to configure the conditions of control for each source. For each source, see **Table 6.11** and **Table 6.12**.

CAUTION

If 0 is written to the EIRFn bit immediately after a peripheral module generates the corresponding interrupt request upon detection of an edge (before an interrupt is accepted by the CPU), the request may be lost.

Writing 1 to the EIRFn bit immediately after an interrupt is accepted by the CPU may lead to re-issuing of the request.

This includes the use of bit-manipulation instructions (set1, clr1, and not1) for writing to any of these registers. For bit-manipulation instructions, see also **Section 3.4.2, Access to Registers by Bit-Manipulation Instructions.**

Executing a bit-manipulation instruction to the lower bytes including the EIMKn bit has no effect on the EIRFn bit.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EICTn	—	—	EIRFn	—	—	—	—	EIMKn	EITBn	—	—	EIP3n	EIP2n	EIP1n	EIP0n
Value after reset	*1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1
R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Note 1. 0: Edge detection
1: Level detection

Table 6.4 EIC0-EIC383 EIC Register Contents (1/2)

Bit Position	Bit Name	Function
15	EICTn	This is the interrupt channel type bit. The values listed below are read in accord with the interrupt input interface. The bit is read-only. 0: Edge detection 1: Level detection When writing to this bit, write the value after reset.
14, 13	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
12	EIRFn	This is an interrupt request flag. The operation of the flag depends on the interrupt input interface. 0: No interrupt request is made (initial value) 1: Interrupt request is made. • Edge detection When the CPU core has accepted an interrupt request on its own channel, the request is automatically cleared. This bit may be set and cleared by software. • Level detection Setting and clearing of the bit by software is not possible. That is, the bit is read-only in this case.
11 to 8	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Table 6.4 EIC0-EIC383 EIC Register Contents (2/2)

Bit Position	Bit Name	Function
7	EIMKn	This is an interrupt mask bit. Setting this bit masks interrupt requests set in the interrupt request flag (EIRFn), i.e. it may be used to obstruct interrupt requests from the given channel to the CPU core. Furthermore, clearing of the PMEI bit in the ICSR does not proceed in response to notification of the presence of an interrupt that has not been processed on the channel for which this bit has been set. This bit does not mask input from an interrupt input pin, so the interrupt request flag still gets set even if this bit is set. The setting of this bit also reflects the setting in the interrupt mask register (IMR). 0: Interrupt processing is enabled. 1: Interrupt processing is disabled (initial value)
6	EITBn	This bit is used to select the method of detection for the interrupt. 0: Direct jumping to an address determined from the level of priority 1: Reference to a table
5, 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3 to 0	EIPn	These bits specify the interrupt priority as one of 16 levels, with 0 as the highest and 15 as the lowest. When multiple EI level-interrupt requests are made simultaneously, the interrupt from the source with the highest priority setting in these bits is selected and conveyed to the CPU core. When these bits specify the same priority level for simultaneously occurring interrupt requests, the source with the lower channel number takes priority. This order is fixed.

Note: n = 0 to 383

Leave registers corresponding to channel numbers listed as reserved in **Table 6.11** and **Table 6.12** at their values following a reset.

6.2.3 EI Level Interrupt Mask Registers 0 to 11 (IMR0 to IMR11)

These registers are a collection of the EIMK bits of the EIC registers. Each bit of IMRn reflects the setting of the corresponding EIMK bit. Setting IMRn is reflected in the corresponding EIMK bit.

IMR0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR0H	EIMK 31	EIMK 30	EIMK 29	EIMK 28	EIMK 27	EIMK 26	EIMK 25	EIMK 24	EIMK 23	EIMK 22	EIMK 21	EIMK 20	EIMK 19	EIMK 18	EIMK 17	EIMK 16
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR0L	EIMK 15	EIMK 14	EIMK 13	EIMK 12	EIMK 11	EIMK 10	EIMK9	EIMK8	EIMK7	EIMK6	EIMK5	EIMK4	EIMK3	EIMK2	EIMK1	EIMK0
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IMR1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR1H	EIMK 63	EIMK 62	EIMK 61	EIMK 60	EIMK 59	EIMK 58	EIMK 57	EIMK 56	EIMK 55	EIMK 54	EIMK 53	EIMK 52	EIMK 51	EIMK 50	EIMK 49	EIMK 48
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR1L	EIMK 47	EIMK 46	EIMK 45	EIMK 44	EIMK 43	EIMK 42	EIMK 41	EIMK 40	EIMK 39	EIMK 38	EIMK 37	EIMK 36	EIMK 35	EIMK 34	EIMK 33	EIMK 32
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

:

:

IMR11

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR11H	EIMK 383	EIMK 382	EIMK 381	EIMK 380	EIMK 379	EIMK 378	EIMK 377	EIMK 376	EIMK 375	EIMK 374	EIMK 373	EIMK 372	EIMK 371	EIMK 370	EIMK 369	EIMK 368
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR11L	EIMK 367	EIMK 366	EIMK 365	EIMK 364	EIMK 363	EIMK 362	EIMK 361	EIMK 360	EIMK 359	EIMK 358	EIMK 357	EIMK 356	EIMK 355	EIMK 354	EIMK 353	EIMK 352
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: EIMK bits corresponding to channel numbers listed as reserved in **Table 6.11** and **Table 6.12** must be set to 1.

6.2.4 EI Level Interrupt Bind Registers 0 to 383 (EIBD0 to EIBD383)

An EIBD register is provided per EI level INT source and establishes association between the sources and CPUs. For each source, see **Table 6.11** and **Table 6.12.**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EIBDnH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GPID	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIBDnL	—	—	—	—	—	—	—	—	—	—	—	—	—	PEID		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0		*1	
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Note 1. EIBD32 to 383: 001

Table 6.5 EIBD0-EIBD383 Register Contents

Bit Position	Bit Name	Function
31 to 18	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
17, 16	GPID	These bits are only present in EIBD32 to EIBD383. Used them to check the PEID settings as listed below. 00: The PEID bits are selecting CPU1 as the target for binding. 01: The PEID bits are selecting CPU2 as the target for binding. 02: The PEID bits are selecting the SubCPU in EMU3 as the target for binding. These bits are reserved in registers EIBD0 to EIBD31. When writing, set the value for these bits to 0. When read, the bits are always read as 0.
15 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2 to 0	PEID	These bits specify the destination to which the interrupt request is bound. However, the PEID bits in the EIBD0 to EIBD31 registers are fixed, and so the values of these bits cannot be changed in those registers. 001: The interrupt is bound to CPU1. 010: The interrupt is bound to CPU2. 011: The interrupt is bound to the SubCPU in EMU3. Always make either of the above settings if interrupts from the corresponding source are to be executed.

Note: Leave registers corresponding to channel numbers listed as reserved in **Table 6.11** at their values following a reset.

CAUTION

Changing the value of the corresponding EIBDn register during the processing of an EIINT request is prohibited.

6.2.5 EXINTCTL — External Interrupt Control Register

This 16-bit register is used to specify the low level, the high level, falling edges, or rising edges as the sense for detection of the individual external interrupts input through the IRQ0 to IRQ7 pins.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ7S		IRQ6S		IRQ5S		IRQ4S		IRQ3S		IRQ2S		IRQ1S		IRQ0S	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 6.6 EXINTCTL Register Contents

Bit Position	Bit Name	Function
15, 14	IRQ7S	External Interrupt Sense Select Each pair of bits is used to select the low level, the high level, falling edges, or rising edges as the sense for detection of interrupt signals on a pin from among IRQ0 to IRQ7. 00: Interrupt requests are detected as the low level on the IRQn input.*1 01: Interrupt requests are detected as the high level on the IRQn input.*1 10: Interrupt requests are detected as falling edges on the IRQn input. 11: Interrupt requests are detected as rising edges on the IRQn input.
13, 12	IRQ6S	
11, 10	IRQ5S	
9, 8	IRQ4S	
7, 6	IRQ3S	
5, 4	IRQ2S	
3, 2	IRQ1S	
1, 0	IRQ0S	

Note 1. When the sense setting is for level detection, the active level should be retained until an interrupt request is accepted.

6.2.6 EXINTSTR — External Interrupt Status Register

EXINTSTR is an eight-bit register that indicates interrupt requests via the external interrupt input pins (IRQ0 to IRQ7). When the sense setting for a given interrupt from among IRQ0 to IRQ7 is for edge detection, register EXINTSTC can be used to clear the corresponding flag in this register.

Bit	7	6	5	4	3	2	1	0
	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 6.7 EXINTSTR Register Contents

Bit Position	Bit Name	Function
7	IRQ7F	External Interrupt Request These bits form a table that indicates the states of interrupt requests from IRQ0 to IRQ7. • When level detection is selected 0: An interrupt request is not present on IRQn. 1: An interrupt request is present on IRQn. [Clearing condition] The level selected by the IRQn1S and IRQn0S bits in the EXINTCTL register not being on the corresponding IRQn. [Setting condition] The level selected by the IRQn1S and IRQn0S bits in the EXINTCTL register being on the corresponding IRQn. • When edge detection is selected 0: An IRQn interrupt request has not been detected. 1: An IRQn interrupt request has been detected. [Clearing condition] 1 is written to an IRQnC bit in the EXINTSTC register. [Setting condition] The edge selected by the IRQn1S and IRQn0S bits in the EXINTCTL register having been generated.
6	IRQ6F	
5	IRQ5F	
4	IRQ4F	
3	IRQ3F	
2	IRQ2F	
1	IRQ1F	
0	IRQ0F	

Note: n = 0 to 7

6.2.7 EXINTSTC — External Interrupt Status Clear Register

Register EXINTSTC is an eight-bit register that is used to clear IRQnF bits in EXINTSTR for which edge detection has been selected as the method of sensing. When this is the case, writing 1 to an IRQnC bit clears the corresponding IRQnF in EXINTSTR if the interrupt has been detected.

Bit	7	6	5	4	3	2	1	0
	IRQ7C	IRQ6C	IRQ5C	IRQ4C	IRQ3C	IRQ2C	IRQ1C	IRQ0C
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Table 6.8 EXINTSTC Register Contents

Bit Position	Bit Name	Function
7	IRQ7C	External Interrupt Request Clear When edge detection has been selected as the method of sensing for an IRQ0 to IRQ7, the corresponding bit is used to clear the status flag that indicates the interrupt request. • When level detection is selected The bits have no function. • When edge detection is selected Writing 1 to an IRQnC bit clears the corresponding IRQnF in EXINTSTR.
6	IRQ6C	
5	IRQ5C	
4	IRQ4C	
3	IRQ3C	
2	IRQ2C	
1	IRQ1C	
0	IRQ0C	

Note: n = 0 to 7

6.2.8 SINTR0 to SINTR7 — Software Interrupt Register

These eight-bit registers are used to control software interrupts 0 to 7 (SINT0 to SINT7).

Writing 01_H to one of these registers increments the value in the counter; writing 00_H decrements it. If the resulting counter value is one or more, the corresponding interrupt from among software interrupts 0 to 7 (SINT0 to SINT7) is generated.

Reading produces the current value of the counter.

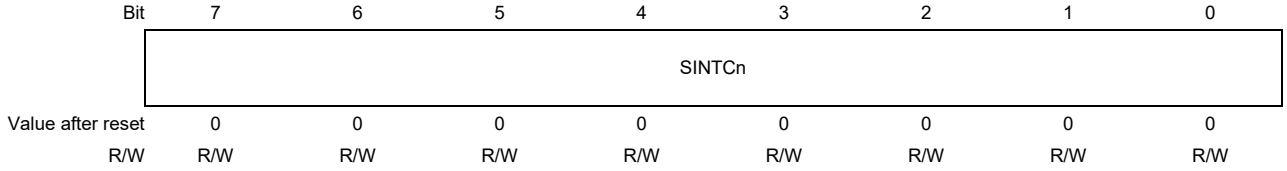


Table 6.9 SINTR0 to SINTR7 Register Contents

Bit Position	Bit Name	Function
7 to 0	SINTCn[7:0]	Software Interrupt Request These bits are used to generate software interrupt requests. [Reading] The value counted number of times counted by the SINTn interrupt request counter is read out. [Writing] Writing 01_H increments the counter.*1 Writing 00_H decrements the counter.*2

Note 1. When 01_H is written to the register while the value of the counter is FF_H, the counter is not incremented and its value remains FF_H.

Note 2. When 00_H is written to the register while the value of the counter is 00_H, the counter is not decremented and its value remains 00_H.

6.2.9 ESINTR0 to ESINTR7 — EMU Software Interrupt Register

These eight-bit registers are used to control EMU software interrupts 0 to 7 (ESINT0 to ESINT7).

Writing 01_H to one of these registers increments the value in the counter; writing 00_H decrements it. If the resulting counter value is one or more, the corresponding interrupt from among EMU software interrupts 0 to 7 (ESINT0 to ESINT7) is generated.

Reading produces the current value of the counter.

Bit	7	6	5	4	3	2	1	0
	ESINTCn7	ESINTCn6	ESINTCn5	ESINTCn4	ESINTCn3	ESINTCn2	ESINTCn1	ESINTCn0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 6.10 ESINTR0 to ESINTR7 Register Contents

Bit Position	Bit Name	Function
7 to 0	ESINTCn[7:0]	EMU Software Interrupt Request These bits are used to generate EMU software interrupt requests. [Reading] The value counted number of times counted by the ESINTn interrupt request counter is read out. [Writing] Writing 01_H increments the counter.*1 Writing 00_H decrements the counter.*2

Note 1. When 01_H is written to the register while the value of the counter is FF_H, the counter is not incremented and its value remains FF_H.

Note 2. When 00_H is written to the register while the value of the counter is 00_H, the counter is not decremented and its value remains 00_H.

6.2.10 PINT0 to PINT7, PINTCLR0 to PINTCLR7 — Peripheral Interrupt Status Registers and Peripheral Interrupt Status Clear Registers

The DTS transfer completion interrupts and transfer count match interrupts are merged in 32-channel units. PINT0 to PINT7 contain the interrupt status flags to indicate the originating channels for the actual interrupts. When multiple interrupt sources are generated within the same register among PINT0 to PINT7, only the single bit on the lower-order side is set.

Interrupt request flags that have been set are cleared from within the interrupt handlers by reading the value of a peripheral interrupt status register (from PINT0 to PINT7) and writing that value to the interrupt clear register (from PINTCLR0 to PINTCLR7) that corresponds to the interrupt channel.

PINT n + x (n = 0 to 3, x = 0)

Bit	31	30	29	28	27	26	25	24
	INTDTS [31+32×n]	INTDTS [30+32×n]	INTDTS [29+32×n]	INTDTS [28+32×n]	INTDTS [27+32×n]	INTDTS [26+32×n]	INTDTS [25+32×n]	INTDTS [24+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	23	22	21	20	19	18	17	16
	INTDTS [23+32×n]	INTDTS [22+32×n]	INTDTS [21+32×n]	INTDTS [20+32×n]	INTDTS [19+32×n]	INTDTS [18+32×n]	INTDTS [17+32×n]	INTDTS [16+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8
	INTDTS [15+32×n]	INTDTS [14+32×n]	INTDTS [13+32×n]	INTDTS [12+32×n]	INTDTS [11+32×n]	INTDTS [10+32×n]	INTDTS [9+32×n]	INTDTS [8+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	7	6	5	4	3	2	1	0
	INTDTS [7+32×n]	INTDTS [6+32×n]	INTDTS [5+32×n]	INTDTS [4+32×n]	INTDTS [3+32×n]	INTDTS [2+32×n]	INTDTS [1+32×n]	INTDTS [0+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

PINT n + x (n = 0 to 3, x = 4)

Bit	31	30	29	28	27	26	25	24
	INTCTDTS [31+32×n]	INTCTDTS [30+32×n]	INTCTDTS [29+32×n]	INTCTDTS [28+32×n]	INTCTDTS [27+32×n]	INTCTDTS [26+32×n]	INTCTDTS [25+32×n]	INTCTDTS [24+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	23	22	21	20	19	18	17	16
	INTCTDTS [23+32×n]	INTCTDTS [22+32×n]	INTCTDTS [21+32×n]	INTCTDTS [20+32×n]	INTCTDTS [19+32×n]	INTCTDTS [18+32×n]	INTCTDTS [17+32×n]	INTCTDTS [16+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8
	INTCTDTS [15+32×n]	INTCTDTS [14+32×n]	INTCTDTS [13+32×n]	INTCTDTS [12+32×n]	INTCTDTS [11+32×n]	INTCTDTS [10+32×n]	INTCTDTS [9+32×n]	INTCTDTS [8+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	7	6	5	4	3	2	1	0
	INTCTDTS [7+32×n]	INTCTDTS [6+32×n]	INTCTDTS [5+32×n]	INTCTDTS [4+32×n]	INTCTDTS [3+32×n]	INTCTDTS [2+32×n]	INTCTDTS [1+32×n]	INTCTDTS [0+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

PINTCLR n + x (n = 0 to 3, x = 0)

Bit	31	30	29	28	27	26	25	24
	INTCLR [31+32×n]	INTCLR [30+32×n]	INTCLR [29+32×n]	INTCLR [28+32×n]	INTCLR [27+32×n]	INTCLR [26+32×n]	INTCLR [25+32×n]	INTCLR [24+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	23	22	21	20	19	18	17	16
	INTCLR [23+32×n]	INTCLR [22+32×n]	INTCLR [21+32×n]	INTCLR [20+32×n]	INTCLR [19+32×n]	INTCLR [18+32×n]	INTCLR [17+32×n]	INTCLR [16+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8
	INTCLR [15+32×n]	INTCLR [14+32×n]	INTCLR [13+32×n]	INTCLR [12+32×n]	INTCLR [11+32×n]	INTCLR [10+32×n]	INTCLR [9+32×n]	INTCLR [8+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	7	6	5	4	3	2	1	0
	INTCLR [7+32×n]	INTCLR [6+32×n]	INTCLR [5+32×n]	INTCLR [4+32×n]	INTCLR [3+32×n]	INTCLR [2+32×n]	INTCLR [1+32×n]	INTCLR [0+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

PINTCLR n + x (n = 0 to 3, x = 4)

Bit	31	30	29	28	27	26	25	24
	INTCTCLR [31+32×n]	INTCTCLR [30+32×n]	INTCTCLR [29+32×n]	INTCTCLR [28+32×n]	INTCTCLR [27+32×n]	INTCTCLR [26+32×n]	INTCTCLR [25+32×n]	INTCTCLR [24+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	23	22	21	20	19	18	17	16
	INTCTCLR [23+32×n]	INTCTCLR [22+32×n]	INTCTCLR [21+32×n]	INTCTCLR [20+32×n]	INTCTCLR [19+32×n]	INTCTCLR [18+32×n]	INTCTCLR [17+32×n]	INTCTCLR [16+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8
	INTCTCLR [15+32×n]	INTCTCLR [14+32×n]	INTCTCLR [13+32×n]	INTCTCLR [12+32×n]	INTCTCLR [11+32×n]	INTCTCLR [10+32×n]	INTCTCLR [9+32×n]	INTCTCLR [8+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	7	6	5	4	3	2	1	0
	INTCTCLR [7+32×n]	INTCTCLR [6+32×n]	INTCTCLR [5+32×n]	INTCTCLR [4+32×n]	INTCTCLR [3+32×n]	INTCTCLR [2+32×n]	INTCTCLR [1+32×n]	INTCTCLR [0+32×n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

PINT0

Bit Position	Bit Name	Function
31 to 0	INTDTS[31:0]	States of DTS transfer completion interrupts from channel 31 to channel 0

PINT1

Bit Position	Bit Name	Function
31 to 0	INTDTS[63:32]	States of DTS transfer completion interrupts from channel 63 to channel 32

PINT2

Bit Position	Bit Name	Function
31 to 0	INTDTS[95:64]	States of DTS transfer completion interrupts from channel 95 to channel 64

PINT3

Bit Position	Bit Name	Function
31 to 0	INTDTS[127:96]	States of DTS transfer completion interrupts from channel 127 to channel 96

PINT4

Bit Position	Bit Name	Function
31 to 0	INTCTDTS[31:0]	States of DTS transfer count match interrupts from channel 31 to channel 0

PINT5

Bit Position	Bit Name	Function
31 to 0	INTCTDTS[63:32]	States of DTS transfer count match interrupts from channel 63 to channel 32

PINT6

Bit Position	Bit Name	Function
31 to 0	INTCTDTS[95:64]	States of DTS transfer count match interrupts from channel 95 to channel 64

PINT7

Bit Position	Bit Name	Function
31 to 0	INTCTDTS[127:96]	States of DTS transfer count match interrupts from channel 127 to channel 96

PINTCLR0

Bit Position	Bit Name	Function
31 to 0	INTCLR[31:0]	The respective bits clear the corresponding flags for the DTS channel 31 to channel 0 transfer completion interrupts. Write the value read from PINT0 to this register from within the interrupt handler.

PINTCLR1

Bit Position	Bit Name	Function
31 to 0	INTCLR[63:32]	The respective bits clear the corresponding flags for the DTS channel 63 to channel 32 transfer completion interrupts. Write the value read from PINT1 to this register from within the interrupt handler.

PINTCLR2

Bit Position	Bit Name	Function
31 to 0	INTCLR[95:64]	The respective bits clear the corresponding flags for the DTS channel 95 to channel 64 transfer completion interrupts. Write the value read from PINT2 to this register from within the interrupt handler.

PINTCLR3

Bit Position	Bit Name	Function
31 to 0	INTCLR[127:96]	The respective bits clear the corresponding flags for the DTS channel 127 to channel 96 transfer completion interrupts. Write the value read from PINT3 to this register from within the interrupt handler.

PINTCLR4

Bit Position	Bit Name	Function
31 to 0	INTCTCLR[31:0]	The respective bits clear the corresponding flags for the DTS channel 31 to channel 0 transfer count match interrupts. Write the value read from PINT4 to this register from within the interrupt handler.

PINTCLR5

Bit Position	Bit Name	Function
31 to 0	INTCTCLR[63:32]	The respective bits clear the corresponding flags for the DTS channel 63 to channel 32 transfer count match interrupts. Write the value read from PINT5 to this register from within the interrupt handler.

PINTCLR6

Bit Position	Bit Name	Function
31 to 0	INTCTCLR[95:64]	The respective bits clear the corresponding flags for the DTS channel 95 to channel 64 transfer count match interrupts. Write the value read from PINT6 to this register from within the interrupt handler.

PINTCLR7

Bit Position	Bit Name	Function
31 to 0	INTCTCLR[127:96]	The respective bits clear the corresponding flags for the DTS channel 127 to channel 96 transfer count match interrupts. Write the value read from PINT7 to this register from within the interrupt handler.

6.3 Interrupt Sources

The six types of interrupt source are external interrupts (IRQ), ECM interrupts, inter-processor interrupts, software interrupts, EMU software interrupts, and peripheral module interrupts.

6.3.1 IRQ Interrupts

IRQ interrupts refers to interrupt requests input via pins IRQ0 to IRQ7. The settings of the external interrupt sense select bits (IRQ7S to IRQ0S) in the external interrupt control (EXINTCTL) register select the low level, the high level, falling edges, or rising edges for detection as the IRQ interrupt on each of the corresponding pins. Furthermore, the interrupt control registers set the priority up to 16 levels for each interrupt source.

When the low level is selected for detection as an IRQ interrupt, an interrupt request signal is conveyed to the interrupt controller (INTC) while the low level is on the corresponding pin from among IRQ0 to IRQ7. The request signal is not conveyed when the level becomes high. The active level should be retained until acceptance of the interrupt request. Checking for requests can be done by reading the corresponding IRQ interrupt request bit (IRQ0F to IRQ7F) in the external interrupt status register (EXINTSTR).

When edge detection is selected for an IRQ interrupt, an interrupt request signal is conveyed to the interrupt controller (INTC) when the corresponding change in the level on the given pin from among IRQ0 to IRQ7 occurs.

Reading bits EXINTSTR.IRQ7F to EXINTSTR.IRQ0F checks whether the corresponding IRQ interrupt request is detected. In addition, the IRQ interrupt request can be cleared by writing 1 to the bit corresponding to EXINTSTC when an edge is detected.

When returning from the IRQ interrupt exception handler, clear the external interrupt status register (EXINTSTR) to prevent re-acceptance by mistake, and then issue an instruction to return from interrupt.

6.3.2 ECM Interrupts

The Error Control Module (ECM) generates ECM interrupt requests by merging error interrupts from multiple sources. For details, refer to **Section 30, Error Control Module (ECM)**.

6.3.3 Inter-Processor Interrupts

Four registers (IPIR_CHn) for conveying interrupts between CPU1 and CPU2 are provided. IPIR_CH0 to IPIR_CH3 are assigned to CH0 to CH3 of user interrupt (EIINT). An interrupt for specific CPUs (including own CPU) can be requested by manipulating bits corresponding to respective CPUs.

The order of priority can be set for each source in 16 levels by using the interrupt control registers.

6.3.4 Software Interrupts

The software interrupt (SINT) is generated by setting the SINTR0 to SINTR7 registers. Multiple interrupt requests can be queued.

The order of priority for software interrupts is set for each source in 16 levels by using interrupt control registers.

6.3.5 EMU Software Interrupts

The EMU software interrupt (ESINT) is generated by setting the ESINTR0 to ESINTR7 registers. It is an interrupt source used specifically for the SubCPU in EMU3. Multiple interrupt requests can be queued.

The order of priority for software interrupts is set for each source in 16 levels by using interrupt control registers.

6.3.6 Internal Peripheral Module Interrupts

Internal peripheral module interrupts are interrupts from the internal peripheral modules listed below.

- Code flash and data flash
- Serial communications interface 3 (SCI3)
- OS Timer (OSTM)
- Window watchdog timer (WDTA)
- Motor control timer (TSG3)
- A/D converter (ADCC)
- Clocked Serial Interface H (CSIH)
- CAN interface (RS-CANFD)
- LIN/UART interface (RLIN3)
- Direct memory access controller (DMAC, DTS)
- Enhanced motor control unit (EMU3)
- R/D converter (RDC3A)
- Timer array unit J (TAUJ)
- Timer array unit D (TAUD)
- Encoder timer (ENCA)
- Timer option (TAPA)
- Timer pattern buffer (TPBA)
- Peripheral interconnection 1 (PIC1B)
- A/D converter option (ADPA)
- Intelligent cryptographic unit (ICU-S)

Since each source is allocated a different interrupt vector, there is no need to determine the source from the interrupt exception handling routines. The order of priority for interrupts is set for each source in 16 levels.

6.4 Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority

Interrupt sources, the corresponding source codes, address offsets of the handlers, and interrupt priority levels for CPU1 and CPU2 are listed in **Table 6.11.** EIINT interrupt channel numbers 0 to 7 for the SubCPU in EMU3 differ from those for CPU1 and CPU2. Interrupt sources, the corresponding source codes, address offsets of the handlers, and interrupt priority levels for channel numbers 0 to 7 of the SubCPU in EMU3 are listed in **Table 6.12.** The other channel numbers for CPU1 and CPU2 are the same as those for the SubCPU in EMU3.

Along with the standard specification for the addresses of handlers, where these are determined by the settings of the PSW.EBV bits and the RBASE and EBASE registers within the CPU core, an extended specification where fixed addresses of the handlers for each of the interrupts are set per channel is also available.

In the standard specification, the address of each exception handler is the offset address plus the base address (set in the RBASE and EBASE registers) within the CPU core. Interrupt offsets are determined through the two methods described below. Put briefly, the offset address is specified or determined by the interrupt channel.

- In one method, the address is determined as the priority level set for each channel (0 to 15) plus a value in the range from 100_H to $1F0_H$ regardless of the interrupt channel number (see note 1 under **Table 6.11** and **Table 6.12**)
- In the other method, the addresses are evenly spread across the range of offsets up to 100_H regardless of the priority level. This method has the advantage of reducing the amount of memory taken up for exception handling (see note 2 under **Table 6.11** and **Table 6.12**)

In the extended specification, a table of the addresses of the exception handlers for each of the interrupt channels is kept, and the addresses of the handlers are extracted from this table. The locations for reference in the table are obtained from the formula below (see note 3 under **Table 6.11** and **Table 6.12**). INTBP is the value in the register of that name within the CPU core.

Location for reading the address of the exception handler
= value in the INTBP register + the channel number × 4 (in bytes)

For information about RH850G3MH exceptions, see the *RH850G3MH Series User's Manual: Software*.

For system error exceptions, see **Section 3, CPU System**.

The orders of priority are set per channel to values. When the same priority level is set for two sources, the channel numbers are used to provide fixed priority levels for the sources, and that with the smaller channel number is selected.

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (1/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3				
					RINT=0*1	RINT=1*2			C1M-A2	C1M-A1	
FE level interrupt	Error control module (ECM) FE level interrupts		(FEINT)	$F0_H$	$+0F0_H$	$+0F0_H$	—		√	√	*6
Inter-processor interrupts	IPIR_CH0		0	1000	*4	*5	$+000_H$	0 to 15 (15)	√	×	*6
	IPIR_CH1		1	1001	*4	*5	$+004_H$	0 to 15 (15)	√	×	*6
	IPIR_CH2		2	1002	*4	*5	$+008_H$	0 to 15 (15)	√	×	*6
	IPIR_CH3		3	1003	*4	*5	$+00C_H$	0 to 15 (15)	√	×	*6
	Reserved		4	1004	*4	*5			×	×	*6
	Reserved		5	1005	*4	*5			×	×	*6
	Reserved		6	1006	*4	*5			×	×	*6
	Reserved		7	1007	*4	*5			×	×	*6
Error control module	Error control module (ECM) maskable interrupt		8	1008	*4	*5	$+020_H$	0 to 15 (15)	√	√	*6
IRQ (external interrupts)	IRQ0 interrupt	√	9	1009	*4	*5	$+024_H$	0 to 15 (15)	√	√	*6
	IRQ1 interrupt	√	10	100A	*4	*5	$+028_H$	0 to 15 (15)	√	√	*6
	IRQ2 interrupt	√	11	100B	*4	*5	$+02C_H$	0 to 15 (15)	√	√	*6
	IRQ3 interrupt	√	12	100C	*4	*5	$+030_H$	0 to 15 (15)	√	√	*6
	IRQ4 interrupt	√	13	100D	*4	*5	$+034_H$	0 to 15 (15)	√	√	*6
	IRQ5 interrupt	√	14	100E	*4	*5	$+038_H$	0 to 15 (15)	√	√	*6
	IRQ6 interrupt	√	15	100F	*4	*5	$+03C_H$	0 to 15 (15)	√	√	*6
	IRQ7 interrupt	√	16	1010	*4	*5	$+040_H$	0 to 15 (15)	√	√	*6
SINT (software interrupts)	SINT0 interrupt	√	17	1011	*4	*5	$+044_H$	0 to 15 (15)	√	√	*6
	SINT1 interrupt	√	18	1012	*4	*5	$+048_H$	0 to 15 (15)	√	√	*6
	SINT2 interrupt	√	19	1013	*4	*5	$+04C_H$	0 to 15 (15)	√	√	*6
	SINT3 interrupt	√	20	1014	*4	*5	$+050_H$	0 to 15 (15)	√	√	*6
	SINT4 interrupt	√	21	1015	*4	*5	$+054_H$	0 to 15 (15)	√	√	*6
	SINT5 interrupt	√	22	1016	*4	*5	$+058_H$	0 to 15 (15)	√	√	*6
	SINT6 interrupt	√	23	1017	*4	*5	$+05C_H$	0 to 15 (15)	√	√	*6
	SINT7 interrupt	√	24	1018	*4	*5	$+060_H$	0 to 15 (15)	√	√	*6
OSTM	OSTM0 interrupt (OSTM0TINT)		25	1019	*4	*5	$+064_H$	0 to 15 (15)	√	√	*6
	OSTM1 interrupt (OSTM1TINT)		26	101A	*4	*5	$+068_H$	0 to 15 (15)	√	√	*6
	OSTM2 interrupt (OSTM2TINT)		27	101B	*4	*5	$+06C_H$	0 to 15 (15)	√	√	*6
	OSTM3 interrupt (OSTM3TINT)		28	101C	*4	*5	$+070_H$	0 to 15 (15)	√	×	*6
DMA	DMA transfer error (DMAERR)		29	101D	*4	*5	$+074_H$	0 to 15 (15)	√	√	*6
	Reserved		30	101E	*4	*5	$+078_H$	0 to 15 (15)	×	×	*6
	Reserved		31	101F	*4	*5	$+07C_H$	0 to 15 (15)	×	×	*6
	Reserved	√	32	1020	*4	*5	$+080_H$	0 to 15 (15)	×	×	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (2/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3				
					RINT=0*1	RINT=1*2			C1M-A2	C1M-A1	
Code Flash/Data Flash	Flash sequencer processing completion interrupt*7		33	1021	*4	*5	+084H	0 to 15 (15)	√	√	*6
	Reserved	√	34	1022	*4	*5	+088H	0 to 15 (15)	×	×	*6
	Reserved		35	1023	*4	*5	+08CH	0 to 15 (15)	×	×	*6
	Reserved		36	1024	*4	*5	+090H	0 to 15 (15)	×	×	*6
	Reserved	√	37	1025	*4	*5	+094H	0 to 15 (15)	×	×	*6
DMAC	Channel 0 transfer completion interrupt/channel 0 transfer count match interrupt		38	1026	*4	*5	+098H	0 to 15 (15)	√	√	*6
	Channel 1 transfer completion interrupt/channel 1 transfer count match interrupt		39	1027	*4	*5	+09CH	0 to 15 (15)	√	√	*6
	Channel 2 transfer completion interrupt/ channel 2 transfer count match interrupt		40	1028	*4	*5	+0A0H	0 to 15 (15)	√	√	*6
	Channel 3 transfer completion interrupt/ channel 3 transfer count match interrupt		41	1029	*4	*5	+0A4H	0 to 15 (15)	√	√	*6
	Channel 4 transfer completion interrupt/ channel 4 transfer count match interrupt		42	102A	*4	*5	+0A8H	0 to 15 (15)	√	√	*6
	Channel 5 transfer completion interrupt/ channel 5 transfer count match interrupt		43	102B	*4	*5	+0ACH	0 to 15 (15)	√	√	*6
	Channel 6 transfer completion interrupt/ channel 6 transfer count match interrupt		44	102C	*4	*5	+0B0H	0 to 15 (15)	√	√	*6
	Channel 7 transfer completion interrupt/ channel 7 transfer count match interrupt		45	102D	*4	*5	+0B4H	0 to 15 (15)	√	√	*6
	Channel 8 transfer completion interrupt/ channel 8 transfer count match interrupt		46	102E	*4	*5	+0B8H	0 to 15 (15)	√	√	*6
	Channel 9 transfer completion interrupt/ channel 9 transfer count match interrupt		47	102F	*4	*5	+0BCH	0 to 15 (15)	√	√	*6
	Channel 10 transfer completion interrupt/ channel 10 transfer count match interrupt		48	1030	*4	*5	+0C0H	0 to 15 (15)	√	√	*6
	Channel 11 transfer completion interrupt/ channel 11 transfer count match interrupt		49	1031	*4	*5	+0C4H	0 to 15 (15)	√	√	*6
	Channel 12 transfer completion interrupt/ channel 12 transfer count match interrupt		50	1032	*4	*5	+0C8H	0 to 15 (15)	√	√	*6
	Channel 13 transfer completion interrupt/ channel 13 transfer count match interrupt		51	1033	*4	*5	+0CCH	0 to 15 (15)	√	√	*6
	Channel 14 transfer completion interrupt/ channel 14 transfer count match interrupt		52	1034	*4	*5	+0D0H	0 to 15 (15)	√	√	*6
	Channel 15 transfer completion interrupt/ channel 15 transfer count match interrupt		53	1035	*4	*5	+0D4H	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (3/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3		C1M-A2	C1M-A1	
					RINT=0*1	RINT=1*2					
WDTA	WDTA0TIT interval timer interrupt (75 % interrupt)		54	1036	*4	*5	+0D8H	0 to 15 (15)	√	√	*6
	WDTA1TIT interval timer interrupt (75 % interrupt)		55	1037	*4	*5	+0DCH	0 to 15 (15)	√	×	*6
EMU3	EMU30 interrupt 0		56	1038	*4	*5	+0E0H	0 to 15 (15)	√	√	*6
	EMU30 interrupt 1		57	1039	*4	*5	+0E4H	0 to 15 (15)	√	√	*6
	EMU30 interrupt 2		58	103A	*4	*5	+0E8H	0 to 15 (15)	√	√	*6
	EMU30 interrupt 3		59	103B	*4	*5	+0ECH	0 to 15 (15)	√	√	*6
	EMU30 interrupt 4		60	103C	*4	*5	+0F0H	0 to 15 (15)	√	√	*6
	EMU30 interrupt 5		61	103D	*4	*5	+0F4H	0 to 15 (15)	√	√	*6
	EMU30 interrupt 6		62	103E	*4	*5	+0F8H	0 to 15 (15)	√	√	*6
	EMU30 interrupt 7		63	103F	*4	*5	+0FCH	0 to 15 (15)	√	√	*6
	Reserved		64	1040	*4	*5	+100H	0 to 15 (15)	×	×	*6
	Reserved		65	1041	*4	*5	+104H	0 to 15 (15)	×	×	*6
	EMU31 interrupt 0		66	1042	*4	*5	+108H	0 to 15 (15)	√	√	*6
	EMU31 interrupt 1		67	1043	*4	*5	+10CH	0 to 15 (15)	√	√	*6
	EMU31 interrupt 2		68	1044	*4	*5	+110H	0 to 15 (15)	√	√	*6
	EMU31 interrupt 3		69	1045	*4	*5	+114H	0 to 15 (15)	√	√	*6
	EMU31 interrupt 4		70	1046	*4	*5	+118H	0 to 15 (15)	√	√	*6
	EMU31 interrupt 5		71	1047	*4	*5	+11CH	0 to 15 (15)	√	√	*6
	EMU31 interrupt 6		72	1048	*4	*5	+120H	0 to 15 (15)	√	√	*6
	EMU31 interrupt 7		73	1049	*4	*5	+124H	0 to 15 (15)	√	√	*6
	Reserved		74	104A	*4	*5	+128H	0 to 15 (15)	×	×	*6
	Reserved		75	104B	*4	*5	+12CH	0 to 15 (15)	×	×	*6
RDC_0	RDC3A0 Z-phase interrupt		76	104C	*4	*5	+130H	0 to 15 (15)	√	√	*6
	RDC3A0 RDC error interrupt		77	104D	*4	*5	+134H	0 to 15 (15)	√	√	*6
	RDC3A0 compare 0 match interrupt		78	104E	*4	*5	+138H	0 to 15 (15)	√	√	*6
	RDC3A0 compare 1 match interrupt		79	104F	*4	*5	+13CH	0 to 15 (15)	√	√	*6
	RDC3A0 compare 2 match interrupt		80	1050	*4	*5	+140H	0 to 15 (15)	√	√	*6
	RDC3A0 excitation timer (ET) interrupt		81	1051	*4	*5	+144H	0 to 15 (15)	√	√	*6
	RDC3A0 BIST completion interrupt		82	1052	*4	*5	+148H	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (4/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3		C1M-A2	C1M-A1	
					RINT=0*1	RINT=1*2					
RDC_1	RDC3A1 Z-phase interrupt		83	1053	*4	*5	+14C$_H$	0 to 15 (15)	√	×	*6
	RDC3A1 RDC error interrupt		84	1054	*4	*5	+150$_H$	0 to 15 (15)	√	×	*6
	RDC3A1 compare 0 interrupt		85	1055	*4	*5	+154$_H$	0 to 15 (15)	√	×	*6
	RDC3A1 compare 1 interrupt		86	1056	*4	*5	+158$_H$	0 to 15 (15)	√	×	*6
	RDC3A1 compare 2 interrupt		87	1057	*4	*5	+15C$_H$	0 to 15 (15)	√	×	*6
	RDC3A1 excitation timer (ET) interrupt		88	1058	*4	*5	+160$_H$	0 to 15 (15)	√	×	*6
	RDC3A1 BIST completion interrupt		89	1059	*4	*5	+164$_H$	0 to 15 (15)	√	×	*6
TAUJ_0	INTTAUJ0I0 interrupt		90	105A	*4	*5	+168$_H$	0 to 15 (15)	√	√	*6
	INTTAUJ0I1 interrupt		91	105B	*4	*5	+16C$_H$	0 to 15 (15)	√	√	*6
	INTTAUJ0I2 interrupt		92	105C	*4	*5	+170$_H$	0 to 15 (15)	√	√	*6
	INTTAUJ0I3 interrupt		93	105D	*4	*5	+174$_H$	0 to 15 (15)	√	√	*6
TAUJ_1	INTTAUJ1I0 interrupt		94	105E	*4	*5	+178$_H$	0 to 15 (15)	√	×	*6
	INTTAUJ1I1 interrupt		95	105F	*4	*5	+17C$_H$	0 to 15 (15)	√	×	*6
	INTTAUJ1I2 interrupt		96	1060	*4	*5	+180$_H$	0 to 15 (15)	√	×	*6
	INTTAUJ1I3 interrupt		97	1061	*4	*5	+184$_H$	0 to 15 (15)	√	×	*6
TAUD_0	INTTAUD0I0 interrupt		98	1062	*4	*5	+188$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I1 interrupt		99	1063	*4	*5	+18C$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I2 interrupt		100	1064	*4	*5	+190$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I3 interrupt		101	1065	*4	*5	+194$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I4 interrupt		102	1066	*4	*5	+198$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I5 interrupt		103	1067	*4	*5	+19C$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I6 interrupt		104	1068	*4	*5	+1A0$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I7 interrupt		105	1069	*4	*5	+1A4$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I8 interrupt		106	106A	*4	*5	+1A8$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I9 interrupt		107	106B	*4	*5	+1AC$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I10 interrupt		108	106C	*4	*5	+1B0$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I11 interrupt		109	106D	*4	*5	+1B4$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I12 interrupt		110	106E	*4	*5	+1B8$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I13 interrupt		111	106F	*4	*5	+1BC$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I14 interrupt		112	1070	*4	*5	+1C0$_H$	0 to 15 (15)	√	√	*6
	INTTAUD0I15 interrupt		113	1071	*4	*5	+1C4$_H$	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (5/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3		C1M-A2	C1M-A1	
					RINT=0*1	RINT=1*2					
TAUD_1	INTTAUD1I0 interrupt		114	1072	*4	*5	+1C8H	0 to 15 (15)	√	√	*6
	INTTAUD1I1 interrupt		115	1073	*4	*5	+1CCH	0 to 15 (15)	√	√	*6
	INTTAUD1I2 interrupt		116	1074	*4	*5	+1D0H	0 to 15 (15)	√	√	*6
	INTTAUD1I3 interrupt		117	1075	*4	*5	+1D4H	0 to 15 (15)	√	√	*6
	INTTAUD1I4 interrupt		118	1076	*4	*5	+1D8H	0 to 15 (15)	√	√	*6
	INTTAUD1I5 interrupt		119	1077	*4	*5	+1DCH	0 to 15 (15)	√	√	*6
	INTTAUD1I6 interrupt		120	1078	*4	*5	+1E0H	0 to 15 (15)	√	√	*6
	INTTAUD1I7 interrupt		121	1079	*4	*5	+1E4H	0 to 15 (15)	√	√	*6
	INTTAUD1I8 interrupt		122	107A	*4	*5	+1E8H	0 to 15 (15)	√	√	*6
	INTTAUD1I9 interrupt		123	107B	*4	*5	+1ECH	0 to 15 (15)	√	√	*6
	INTTAUD1I10 interrupt		124	107C	*4	*5	+1F0H	0 to 15 (15)	√	√	*6
	INTTAUD1I11 interrupt		125	107D	*4	*5	+1F4H	0 to 15 (15)	√	√	*6
	INTTAUD1I12 interrupt		126	107E	*4	*5	+1F8H	0 to 15 (15)	√	√	*6
	INTTAUD1I13 interrupt		127	107F	*4	*5	+1FCH	0 to 15 (15)	√	√	*6
	INTTAUD1I14 interrupt		128	1080	*4	*5	+200H	0 to 15 (15)	√	√	*6
	INTTAUD1I15 interrupt		129	1081	*4	*5	+204H	0 to 15 (15)	√	√	*6
TAUD_2	INTTAUD2I0 interrupt		130	1082	*4	*5	+208H	0 to 15 (15)	√	×	*6
	INTTAUD2I1 interrupt		131	1083	*4	*5	+20CH	0 to 15 (15)	√	×	*6
	INTTAUD2I2 interrupt		132	1084	*4	*5	+210H	0 to 15 (15)	√	×	*6
	INTTAUD2I3 interrupt		133	1085	*4	*5	+214H	0 to 15 (15)	√	×	*6
	INTTAUD2I4 interrupt		134	1086	*4	*5	+218H	0 to 15 (15)	√	×	*6
	INTTAUD2I5 interrupt		135	1087	*4	*5	+21CH	0 to 15 (15)	√	×	*6
	INTTAUD2I6 interrupt		136	1088	*4	*5	+220H	0 to 15 (15)	√	×	*6
	INTTAUD2I7 interrupt		137	1089	*4	*5	+224H	0 to 15 (15)	√	×	*6
	INTTAUD2I8 interrupt		138	108A	*4	*5	+228H	0 to 15 (15)	√	×	*6
	INTTAUD2I9 interrupt		139	108B	*4	*5	+22CH	0 to 15 (15)	√	×	*6
	INTTAUD2I10 interrupt		140	108C	*4	*5	+230H	0 to 15 (15)	√	×	*6
	INTTAUD2I11 interrupt		141	108D	*4	*5	+234H	0 to 15 (15)	√	×	*6
	INTTAUD2I12 interrupt		142	108E	*4	*5	+238H	0 to 15 (15)	√	×	*6
	INTTAUD2I13 interrupt		143	108F	*4	*5	+23CH	0 to 15 (15)	√	×	*6
	INTTAUD2I14 interrupt		144	1090	*4	*5	+240H	0 to 15 (15)	√	×	*6
	INTTAUD2I15 interrupt		145	1091	*4	*5	+244H	0 to 15 (15)	√	×	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (6/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3		C1M-A2	C1M-A1	
					RINT=0*1	RINT=1*2					
TAUD_3	INTTAUD3I0 interrupt		146	1092	*4	*5	+248H	0 to 15 (15)	√	×	*6
	INTTAUD3I1 interrupt		147	1093	*4	*5	+24CH	0 to 15 (15)	√	×	*6
	INTTAUD3I2 interrupt		148	1094	*4	*5	+250H	0 to 15 (15)	√	×	*6
	INTTAUD3I3 interrupt		149	1095	*4	*5	+254H	0 to 15 (15)	√	×	*6
	INTTAUD3I4 interrupt		150	1096	*4	*5	+258H	0 to 15 (15)	√	×	*6
	INTTAUD3I5 interrupt		151	1097	*4	*5	+25CH	0 to 15 (15)	√	×	*6
	INTTAUD3I6 interrupt		152	1098	*4	*5	+260H	0 to 15 (15)	√	×	*6
	INTTAUD3I7 interrupt		153	1099	*4	*5	+264H	0 to 15 (15)	√	×	*6
	INTTAUD3I8 interrupt		154	109A	*4	*5	+268H	0 to 15 (15)	√	×	*6
	INTTAUD3I9 interrupt		155	109B	*4	*5	+26CH	0 to 15 (15)	√	×	*6
	INTTAUD3I10 interrupt		156	109C	*4	*5	+270H	0 to 15 (15)	√	×	*6
	INTTAUD3I11 interrupt		157	109D	*4	*5	+274H	0 to 15 (15)	√	×	*6
	INTTAUD3I12 interrupt		158	109E	*4	*5	+278H	0 to 15 (15)	√	×	*6
	INTTAUD3I13 interrupt		159	109F	*4	*5	+27CH	0 to 15 (15)	√	×	*6
	INTTAUD3I14 interrupt		160	10A0	*4	*5	+280H	0 to 15 (15)	√	×	*6
	INTTAUD3I15 interrupt		161	10A1	*4	*5	+284H	0 to 15 (15)	√	×	*6
	Reserved		162	10A2	*4	*5	+288H	0 to 15 (15)	×	×	*6
	Reserved		163	10A3	*4	*5	+28CH	0 to 15 (15)	×	×	*6
PIC1B	ENCA0 compare 0 match or capture 0 interrupt		164	10A4	*4	*5	+290H	0 to 15 (15)	√	√	*6
	ENCA1 compare 0 match or capture 0 interrupt		165	10A5	*4	*5	+294H	0 to 15 (15)	√	√	*6
ENCA_0	Overflow interrupt		166	10A6	*4	*5	+298H	0 to 15 (15)	√	√	*6
	Compare 1 match or capture 1 interrupt		167	10A7	*4	*5	+29CH	0 to 15 (15)	√	√	*6
	Underflow interrupt		168	10A8	*4	*5	+2A0H	0 to 15 (15)	√	√	*6
	Interrupt to indicate clearing due to clearing input from the encoder		169	10A9	*4	*5	+2A4H	0 to 15 (15)	√	√	*6
ENCA_1	Overflow interrupt		170	10AA	*4	*5	+2A8H	0 to 15 (15)	√	√	*6
	Compare 1 match or capture 1 interrupt		171	10AB	*4	*5	+2ACH	0 to 15 (15)	√	√	*6
	Underflow interrupt		172	10AC	*4	*5	+2B0H	0 to 15 (15)	√	√	*6
	Interrupt to indicate clearing due to clearing input from the encoder		173	10AD	*4	*5	+2B4H	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (7/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3				
					RINT=0*1	RINT=1*2			C1M-A2	C1M-A1	
	Reserved		174	10AE	*4	*5	+2B8H	0 to 15 (15)	×	×	*6
	Reserved		175	10AF	*4	*5	+2BCH	0 to 15 (15)	×	×	*6
TAPA_0	TAPA0 peak interrupt 0		176	10B0	*4	*5	+2C0H	0 to 15 (15)	√	√	*6
	TAPA0 trough interrupt		177	10B1	*4	*5	+2C4H	0 to 15 (15)	√	√	*6
TAPA_1	TAPA1 peak interrupt 0		178	10B2	*4	*5	+2C8H	0 to 15 (15)	√	√	*6
	TAPA1 trough interrupt 0		179	10B3	*4	*5	+2CCH	0 to 15 (15)	√	√	*6
TAPA_2	TAPA2 peak interrupt 0		180	10B4	*4	*5	+2D0H	0 to 15 (15)	√	×	*6
	TAPA2 trough interrupt 0		181	10B5	*4	*5	+2D4H	0 to 15 (15)	√	×	*6
	Reserved		182	10B6	*4	*5	+2D8H	0 to 15 (15)	×	×	*6
	Reserved		183	10B7	*4	*5	+2DCH	0 to 15 (15)	×	×	*6
TPBA_0	Cycle match detection interrupt		184	10B8	*4	*5	+2E0H	0 to 15 (15)	√	√	*6
	Duty match detection interrupt		185	10B9	*4	*5	+2E4H	0 to 15 (15)	√	√	*6
	Pattern match detection interrupt		186	10BA	*4	*5	+2E8H	0 to 15 (15)	√	√	*6
TPBA_1	Cycle match detection interrupt		187	10BB	*4	*5	+2ECH	0 to 15 (15)	√	×	*6
	Duty match detection interrupt		188	10BC	*4	*5	+2F0H	0 to 15 (15)	√	×	*6
	Pattern match detection interrupt		189	10BD	*4	*5	+2F4H	0 to 15 (15)	√	×	*6
TSG3_0	TSG30CMP1E compare match interrupt		190	10BE	*4	*5	+2F8H	0 to 15 (15)	√	√	*6
	TSG30CMP2E compare match interrupt		191	10BF	*4	*5	+2FCH	0 to 15 (15)	√	√	*6
	TSG30CMP3E compare match interrupt		192	10C0	*4	*5	+300H	0 to 15 (15)	√	√	*6
	TSG30CMP4E compare match interrupt		193	10C1	*4	*5	+304H	0 to 15 (15)	√	√	*6
	TSG30CMP5E compare match interrupt		194	10C2	*4	*5	+308H	0 to 15 (15)	√	√	*6
	TSG30CMP6E compare match interrupt		195	10C3	*4	*5	+30CH	0 to 15 (15)	√	√	*6
	TSG30CMP7E compare match interrupt		196	10C4	*4	*5	+310H	0 to 15 (15)	√	√	*6
	TSG30CMP8E compare match interrupt		197	10C5	*4	*5	+314H	0 to 15 (15)	√	√	*6
	TSG30CMP9E compare match interrupt		198	10C6	*4	*5	+318H	0 to 15 (15)	√	√	*6
	TSG30CMP10E compare match interrupt		199	10C7	*4	*5	+31CH	0 to 15 (15)	√	√	*6
	TSG30CMP11E compare match interrupt		200	10C8	*4	*5	+320H	0 to 15 (15)	√	√	*6
	TSG30CMP12E compare match interrupt		201	10C9	*4	*5	+324H	0 to 15 (15)	√	√	*6
	Error interrupt		202	10CA	*4	*5	+328H	0 to 15 (15)	√	√	*6
	Warning interrupt		203	10CB	*4	*5	+32CH	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (8/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3				
					RINT=0*1	RINT=1*2			C1M-A2	C1M-A1	
TSG3_1	TSG31CMP1E compare match interrupt		204	10CC	*4	*5	+330H	0 to 15 (15)	√	√	*6
	TSG31CMP2E compare match interrupt		205	10CD	*4	*5	+334H	0 to 15 (15)	√	√	*6
	TSG31CMP3E compare match interrupt		206	10CE	*4	*5	+338H	0 to 15 (15)	√	√	*6
	TSG31CMP4E compare match interrupt		207	10CF	*4	*5	+33CH	0 to 15 (15)	√	√	*6
	TSG31CMP5E compare match interrupt		208	10D0	*4	*5	+340H	0 to 15 (15)	√	√	*6
	TSG31CMP6E compare match interrupt		209	10D1	*4	*5	+344H	0 to 15 (15)	√	√	*6
	TSG31CMP7E compare match interrupt		210	10D2	*4	*5	+348H	0 to 15 (15)	√	√	*6
	TSG31CMP8E compare match interrupt		211	10D3	*4	*5	+34CH	0 to 15 (15)	√	√	*6
	TSG31CMP9E compare match interrupt		212	10D4	*4	*5	+350H	0 to 15 (15)	√	√	*6
	TSG31CMP10E compare match interrupt		213	10D5	*4	*5	+354H	0 to 15 (15)	√	√	*6
	TSG31CMP11E compare match interrupt		214	10D6	*4	*5	+358H	0 to 15 (15)	√	√	*6
	TSG31CMP12E compare match interrupt		215	10D7	*4	*5	+35CH	0 to 15 (15)	√	√	*6
	Error interrupt		216	10D8	*4	*5	+360H	0 to 15 (15)	√	√	*6
	Warning interrupt		217	10D9	*4	*5	+364H	0 to 15 (15)	√	√	*6
TSG3_2	TSG32CMP1E compare match interrupt		218	10DA	*4	*5	+368H	0 to 15 (15)	√	×	*6
	TSG32CMP2E compare match interrupt		219	10DB	*4	*5	+36CH	0 to 15 (15)	√	×	*6
	TSG32CMP3E compare match interrupt		220	10DC	*4	*5	+370H	0 to 15 (15)	√	×	*6
	TSG32CMP4E compare match interrupt		221	10DD	*4	*5	+374H	0 to 15 (15)	√	×	*6
	TSG32CMP5E compare match interrupt		222	10DE	*4	*5	+378H	0 to 15 (15)	√	×	*6
	TSG32CMP6E compare match interrupt		223	10DF	*4	*5	+37CH	0 to 15 (15)	√	×	*6
	TSG32CMP7E compare match interrupt		224	10E0	*4	*5	+380H	0 to 15 (15)	√	×	*6
	TSG32CMP8E compare match interrupt		225	10E1	*4	*5	+384H	0 to 15 (15)	√	×	*6
	TSG32CMP9E compare match interrupt		226	10E2	*4	*5	+388H	0 to 15 (15)	√	×	*6
	TSG32CMP10E compare match interrupt		227	10E3	*4	*5	+38CH	0 to 15 (15)	√	×	*6
	TSG32CMP11E compare match interrupt		228	10E4	*4	*5	+390H	0 to 15 (15)	√	×	*6
	TSG32CMP12E compare match interrupt		229	10E5	*4	*5	+394H	0 to 15 (15)	√	×	*6
	Error interrupt		230	10E6	*4	*5	+398H	0 to 15 (15)	√	×	*6
	Warning interrupt		231	10E7	*4	*5	+39CH	0 to 15 (15)	√	×	*6
TSG3_0	TSG30CMP0E compare match interrupt		232	10E8	*4	*5	+3A0H	0 to 15 (15)	√	√	*6
	Peak interrupt		233	10E9	*4	*5	+3A4H	0 to 15 (15)	√	√	*6
	Trough interrupt		234	10EA	*4	*5	+3A8H	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (9/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3		C1M-A2	C1M-A1	
					RINT=0*1	RINT=1*2					
TSG3_1	TSG31CMP0E compare match interrupt		235	10EB	*4	*5	+3AC$_H$	0 to 15 (15)	√	√	*6
	Peak interrupt		236	10EC	*4	*5	+3B0$_H$	0 to 15 (15)	√	√	*6
	Trough interrupt		237	10ED	*4	*5	+3B4$_H$	0 to 15 (15)	√	√	*6
TSG3_2	TSG32CMP0E compare match interrupt		238	10EE	*4	*5	+3B8$_H$	0 to 15 (15)	√	×	*6
	Peak interrupt		239	10EF	*4	*5	+3BC$_H$	0 to 15 (15)	√	×	*6
	Trough interrupt		240	10F0	*4	*5	+3C0$_H$	0 to 15 (15)	√	×	*6
ICUSE	Block data write request interrupt		241	10F1	*4	*5	+3C4$_H$	0 to 15 (15)	√	√	*6
	Block data read request interrupt/Command completion notification interrupt		242	10F2	*4	*5	+3C8$_H$	0 to 15 (15)	√	√	*6
ADC	ADI00 ADCC0 scan group 0 completion interrupt		243	10F3	*4	*5	+3CC$_H$	0 to 15 (15)	√	√	*6
	ADI01 ADCC0 scan group 1 completion interrupt		244	10F4	*4	*5	+3D0$_H$	0 to 15 (15)	√	√	*6
	ADI02 ADCC0 scan group 2 completion interrupt		245	10F5	*4	*5	+3D4$_H$	0 to 15 (15)	√	√	*6
	ADI03 ADCC0 scan group 3 completion interrupt		246	10F6	*4	*5	+3D8$_H$	0 to 15 (15)	√	√	*6
	ADI04 ADCC0 scan group 4 completion interrupt		247	10F7	*4	*5	+3DC$_H$	0 to 15 (15)	√	√	*6
	ADI10 ADCC1 scan group 0 completion interrupt		248	10F8	*4	*5	+3E0$_H$	0 to 15 (15)	√	√	*6
	ADI11 ADCC1 scan group 1 completion interrupt		249	10F9	*4	*5	+3E4$_H$	0 to 15 (15)	√	√	*6
	ADI12 ADCC1 scan group 2 completion interrupt		250	10FA	*4	*5	+3E8$_H$	0 to 15 (15)	√	√	*6
	ADI13 ADCC1 scan group 3 completion interrupt		251	10FB	*4	*5	+3EC$_H$	0 to 15 (15)	√	√	*6
	ADI14 ADCC1 scan group 4 completion interrupt		252	10FC	*4	*5	+3F0$_H$	0 to 15 (15)	√	√	*6
	ADI20 ADCC2 scan group 0 completion interrupt		253	10FD	*4	*5	+3F4$_H$	0 to 15 (15)	√	√	*6
	ADI21 ADCC2 scan group 1 completion interrupt		254	10FE	*4	*5	+3F8$_H$	0 to 15 (15)	√	√	*6
	ADI22 ADCC2 scan group 2 completion interrupt		255	10FF	*4	*5	+3FC$_H$	0 to 15 (15)	√	√	*6
	ADI23 ADCC2 scan group 3 completion interrupt		256	1100	*4	*5	+400$_H$	0 to 15 (15)	√	√	*6
	ADI24 ADCC2 scan group 4 completion interrupt		257	1101	*4	*5	+404$_H$	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (10/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3				
					RINT=0*1	RINT=1*2			C1M-A2	C1M-A1	
ADC	ADE0 ADC0AD error interrupt		258	1102	*4	*5	+408H	0 to 15 (15)	√	√	*6
	ADE1 ADC1AD error interrupt		259	1103	*4	*5	+40CH	0 to 15 (15)	√	√	*6
	ADE2 ADC2AD error interrupt		260	1104	*4	*5	+410H	0 to 15 (15)	√	√	*6
	Reserved		261	1105	*4	*5	+414H	0 to 15 (15)	×	×	*6
	Reserved		262	1106	*4	*5	+418H	0 to 15 (15)	×	×	*6
RS-CANFD	CAN receive FIFO interrupt	√	263	1107	*4	*5	+41CH	0 to 15 (15)	√	√	*6
	CAN global error interrupt	√	264	1108	*4	*5	+420H	0 to 15 (15)	√	√	*6
RS-CANFD	CAN0 transmit/receive FIFO reception completion interrupt	√	265	1109	*4	*5	+424H	0 to 15 (15)	√	√	*6
ch0	CAN0 error interrupt	√	266	110A	*4	*5	+428H	0 to 15 (15)	√	√	*6
	CAN0 transmit interrupt	√	267	110B	*4	*5	+42CH	0 to 15 (15)	√	√	*6
RS-CANFD	CAN1 transmit/receive FIFO reception completion interrupt	√	268	110C	*4	*5	+430H	0 to 15 (15)	√	√	*6
ch1	CAN1 error interrupt	√	269	110D	*4	*5	+434H	0 to 15 (15)	√	√	*6
	CAN1 transmit interrupt	√	270	110E	*4	*5	+438H	0 to 15 (15)	√	√	*6
RS-CANFD	CAN2 transmit/receive FIFO reception completion interrupt	√	271	110F	*4	*5	+43CH	0 to 15 (15)	√	√	*6
ch2	CAN2 error interrupt	√	272	1110	*4	*5	+440H	0 to 15 (15)	√	√	*6
	CAN2 transmit interrupt	√	273	1111	*4	*5	+444H	0 to 15 (15)	√	√	*6
RS-CANFD	CAN3 transmit/receive FIFO reception completion interrupt	√	274	1112	*4	*5	+448H	0 to 15 (15)	√	√	*6
ch3	CAN3 error interrupt	√	275	1113	*4	*5	+44CH	0 to 15 (15)	√	√	*6
	CAN3 transmit interrupt	√	276	1114	*4	*5	+450H	0 to 15 (15)	√	√	*6
RLIN3_0	RLIN30 interrupt		277	1115	*4	*5	+454H	0 to 15 (15)	√	√	*6
	RLIN30 transmit interrupt		278	1116	*4	*5	+458H	0 to 15 (15)	√	√	*6
	RLIN30 reception completion interrupt		279	1117	*4	*5	+45CH	0 to 15 (15)	√	√	*6
	RLIN30 status interrupt		280	1118	*4	*5	+460H	0 to 15 (15)	√	√	*6
RLIN3_1	RLIN31 interrupt		281	1119	*4	*5	+464H	0 to 15 (15)	√	√	*6
	RLIN31 transmit interrupt		282	111A	*4	*5	+468H	0 to 15 (15)	√	√	*6
	RLIN31 reception completion interrupt		283	111B	*4	*5	+46CH	0 to 15 (15)	√	√	*6
	RLIN31 status interrupt		284	111C	*4	*5	+470H	0 to 15 (15)	√	√	*6
RLIN3_2	RLIN32 interrupt		285	111D	*4	*5	+474H	0 to 15 (15)	√	√	*6
	RLIN32 transmit interrupt		286	111E	*4	*5	+478H	0 to 15 (15)	√	√	*6
	RLIN32 reception completion interrupt		287	111F	*4	*5	+47CH	0 to 15 (15)	√	√	*6
	RLIN32 status interrupt		288	1120	*4	*5	+480H	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (11/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3				
					RINT=0*1	RINT=1*2			C1M-A2	C1M-A1	
SCI_0	ERI (receive error)	√	289	1121	*4	*5	+484H	0 to 15 (15)	√	√	*6
	RXI (receive data full)		290	1122	*4	*5	+488H	0 to 15 (15)	√	√	*6
	TXI (transmit data empty)		291	1123	*4	*5	+48CH	0 to 15 (15)	√	√	*6
	TEI (transmission completion)	√	292	1124	*4	*5	+490H	0 to 15 (15)	√	√	*6
SCI_1	ERI (receive error)	√	293	1125	*4	*5	+494H	0 to 15 (15)	√	√	*6
	RXI (receive data full)		294	1126	*4	*5	+498H	0 to 15 (15)	√	√	*6
	TXI (transmit data empty)		295	1127	*4	*5	+49CH	0 to 15 (15)	√	√	*6
	TEI (transmission completion)	√	296	1128	*4	*5	+4A0H	0 to 15 (15)	√	√	*6
SCI_2	ERI (receive error)	√	297	1129	*4	*5	+4A4H	0 to 15 (15)	√	√	*6
	RXI (receive data full)		298	112A	*4	*5	+4A8H	0 to 15 (15)	√	√	*6
	TXI (transmit data empty)		299	112B	*4	*5	+4ACH	0 to 15 (15)	√	√	*6
	TEI (transmission completion)	√	300	112C	*4	*5	+4B0H	0 to 15 (15)	√	√	*6
CSIH_0	Communication status interrupt		301	112D	*4	*5	+4B4H	0 to 15 (15)	√	√	*6
	Receive status interrupt		302	112E	*4	*5	+4B8H	0 to 15 (15)	√	√	*6
	Communication error interrupt		303	112F	*4	*5	+4BCH	0 to 15 (15)	√	√	*6
	Job completion interrupt		304	1130	*4	*5	+4C0H	0 to 15 (15)	√	√	*6
CSIH_1	Communication status interrupt		305	1131	*4	*5	+4C4H	0 to 15 (15)	√	√	*6
	Receive status interrupt		306	1132	*4	*5	+4C8H	0 to 15 (15)	√	√	*6
	Communication error interrupt		307	1133	*4	*5	+4CCH	0 to 15 (15)	√	√	*6
	Job completion interrupt		308	1134	*4	*5	+4D0H	0 to 15 (15)	√	√	*6
CSIH_2	Communication status interrupt		309	1135	*4	*5	+4D4H	0 to 15 (15)	√	√	*6
	Receive status interrupt		310	1136	*4	*5	+4D8H	0 to 15 (15)	√	√	*6
	Communication error interrupt		311	1137	*4	*5	+4DCH	0 to 15 (15)	√	√	*6
	Job completion interrupt		312	1138	*4	*5	+4E0H	0 to 15 (15)	√	√	*6
DTS	ch0-31 transfer completion interrupt	√	313	1139	*4	*5	+4E4H	0 to 15 (15)	√	√	*6
	ch32-63 transfer completion interrupt	√	314	113A	*4	*5	+4E8H	0 to 15 (15)	√	√	*6
	ch64-95 transfer completion interrupt	√	315	113B	*4	*5	+4ECH	0 to 15 (15)	√	√	*6
	ch96-127 transfer completion interrupt	√	316	113C	*4	*5	+4F0H	0 to 15 (15)	√	√	*6
	ch0-31 transfer count match interrupt	√	317	113D	*4	*5	+4F4H	0 to 15 (15)	√	√	*6
	ch32-63 transfer count match interrupt	√	318	113E	*4	*5	+4F8H	0 to 15 (15)	√	√	*6
	ch64-95 transfer count match interrupt	√	319	113F	*4	*5	+4FCH	0 to 15 (15)	√	√	*6
	ch96-127 transfer count match interrupt	√	320	1140	*4	*5	+500H	0 to 15 (15)	√	√	*6
RSENT_0	RSENT receive interrupt		321	1141	*4	*5	+504H	0 to 15 (15)	√	√	*6
	RSENT status interrupt	√	322	1142	*4	*5	+508H	0 to 15 (15)	√	√	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (12/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3				
					RINT=0*1	RINT=1*2			C1M-A2	C1M-A1	
RSENT_1	RSENT receive interrupt		323	1143	*4	*5	+50C$_H$	0 to 15 (15)	√	√	*6
	RSENT status interrupt	√	324	1144	*4	*5	+510$_H$	0 to 15 (15)	√	√	*6
RSENT_2	RSENT receive interrupt		325	1145	*4	*5	+514$_H$	0 to 15 (15)	√	√	*6
	RSENT status interrupt	√	326	1146	*4	*5	+518$_H$	0 to 15 (15)	√	√	*6
RSENT_3	RSENT receive interrupt		327	1147	*4	*5	+51C$_H$	0 to 15 (15)	√	√	*6
	RSENT status interrupt	√	328	1148	*4	*5	+520$_H$	0 to 15 (15)	√	√	*6
	Reserved		329	1149	*4	*5	+524$_H$	0 to 15 (15)	×	×	*6
	Reserved		330	114A	*4	*5	+528$_H$	0 to 15 (15)	×	×	*6
	Reserved		331	114B	*4	*5	+52C$_H$	0 to 15 (15)	×	×	*6
	Reserved		332	114C	*4	*5	+530$_H$	0 to 15 (15)	×	×	*6
ADPA	ADPA control notification interrupt 0		333	114D	*4	*5	+534$_H$	0 to 15 (15)	√	√	*6
	ADPA control notification interrupt 1		334	114E	*4	*5	+538$_H$	0 to 15 (15)	√	√	*6
	ADPA control notification interrupt 2		335	114F	*4	*5	+53C$_H$	0 to 15 (15)	√	√	*6
	Reserved		336	1150	*4	*5	+540$_H$	0 to 15 (15)	×	×	*6
	Reserved		337	1151	*4	*5	+544$_H$	0 to 15 (15)	×	×	*6
	Reserved		338	1152	*4	*5	+548$_H$	0 to 15 (15)	×	×	*6
	Reserved		339	1153	*4	*5	+54C$_H$	0 to 15 (15)	×	×	*6
	Reserved		340	1154	*4	*5	+550$_H$	0 to 15 (15)	×	×	*6
	Reserved		341	1155	*4	*5	+554$_H$	0 to 15 (15)	×	×	*6
	Reserved		342	1156	*4	*5	+558$_H$	0 to 15 (15)	×	×	*6
	Reserved		343	1157	*4	*5	+55C$_H$	0 to 15 (15)	×	×	*6
	Reserved		344	1158	*4	*5	+560$_H$	0 to 15 (15)	×	×	*6
	Reserved		345	1159	*4	*5	+564$_H$	0 to 15 (15)	×	×	*6
	Reserved		346	115A	*4	*5	+568$_H$	0 to 15 (15)	×	×	*6
	Reserved		347	115B	*4	*5	+56C$_H$	0 to 15 (15)	×	×	*6
	Reserved		348	115C	*4	*5	+570$_H$	0 to 15 (15)	×	×	*6
	Reserved		349	115D	*4	*5	+574$_H$	0 to 15 (15)	×	×	*6
	Reserved		350	115E	*4	*5	+578$_H$	0 to 15 (15)	×	×	*6
	Reserved		351	115F	*4	*5	+57C$_H$	0 to 15 (15)	×	×	*6
	Reserved		352	1160	*4	*5	+580$_H$	0 to 15 (15)	×	×	*6
	Reserved		353	1161	*4	*5	+584$_H$	0 to 15 (15)	×	×	*6
	Reserved		354	1162	*4	*5	+588$_H$	0 to 15 (15)	×	×	*6
	Reserved		355	1163	*4	*5	+58C$_H$	0 to 15 (15)	×	×	*6
	Reserved		356	1164	*4	*5	+590$_H$	0 to 15 (15)	×	×	*6
	Reserved		357	1165	*4	*5	+594$_H$	0 to 15 (15)	×	×	*6
	Reserved		358	1166	*4	*5	+598$_H$	0 to 15 (15)	×	×	*6
	Reserved		359	1167	*4	*5	+59C$_H$	0 to 15 (15)	×	×	*6

Table 6.11 Interrupt Exception Handlers and Orders of Priority (CPU1, CPU2) (13/13)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel No.	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Target Device		Default Priority Order
					Direct Branching	Direct Branching	Reference to a Table*3		C1M-A2	C1M-A1	
					RINT=0*1	RINT=1*2					
	Reserved		360	1168	*4	*5	+5A0H	0 to 15 (15)	×	×	*6
	Reserved		361	1169	*4	*5	+5A4H	0 to 15 (15)	×	×	*6
	Reserved		362	116A	*4	*5	+5A8H	0 to 15 (15)	×	×	*6
	Reserved		363	116B	*4	*5	+5ACH	0 to 15 (15)	×	×	*6
	Reserved		364	116C	*4	*5	+5B0H	0 to 15 (15)	×	×	*6
	Reserved		365	116D	*4	*5	+5B4H	0 to 15 (15)	×	×	*6
	Reserved		366	116E	*4	*5	+5B8H	0 to 15 (15)	×	×	*6
	Reserved		367	116F	*4	*5	+5BCH	0 to 15 (15)	×	×	*6
	Reserved		368	1170	*4	*5	+5C0H	0 to 15 (15)	×	×	*6
	Reserved		369	1171	*4	*5	+5C4H	0 to 15 (15)	×	×	*6
	Reserved		370	1172	*4	*5	+5C8H	0 to 15 (15)	×	×	*6
	Reserved		371	1173	*4	*5	+5CCH	0 to 15 (15)	×	×	*6
	Reserved		372	1174	*4	*5	+5D0H	0 to 15 (15)	×	×	*6
	Reserved		373	1175	*4	*5	+5D4H	0 to 15 (15)	×	×	*6
	Reserved		374	1176	*4	*5	+5D8H	0 to 15 (15)	×	×	*6
	Reserved		375	1177	*4	*5	+5DCH	0 to 15 (15)	×	×	*6
	Reserved		376	1178	*4	*5	+5E0H	0 to 15 (15)	×	×	*6
	Reserved		377	1179	*4	*5	+5E4H	0 to 15 (15)	×	×	*6
	Reserved		378	117A	*4	*5	+5E8H	0 to 15 (15)	×	×	*6
	Reserved		379	117B	*4	*5	+5ECH	0 to 15 (15)	×	×	*6
	Reserved		380	117C	*4	*5	+5F0H	0 to 15 (15)	×	×	*6
	Reserved		381	117D	*4	*5	+5F4H	0 to 15 (15)	×	×	*6
	Reserved		382	117E	*4	*5	+5F8H	0 to 15 (15)	×	×	*6
	Reserved		383	117F	*4	*5	+5FCH	0 to 15 (15)	×	×	*6

Note: Withdrawing an interrupt request in the form of a level interrupt requires clearing the given bit in the status register of the corresponding module from within the interrupt processing routing. For information about clearing the status register, see **Section 3.4.1, Synchronizing Completion of Store Instruction and Generation of Subsequent Instruction**. The EICn.EICTn bit is 1. Software cannot clear the EICN.EIRFn bit.

Note 1. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.

Note 2. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.

Note 3. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.

Note 4. The offset address does not vary depending on the channel. It should be selected from addresses $+100_H$ to $+1F0_H$ based on the order of priority.

Note 5. All the offset addresses are $+100_H$ regardless of the order of priority because of offset address reduction.

Note 6. Items are in descending order of priority from top to bottom.

Note 7. See the *RH850/C1M-A Flash Memory User's Manual: Hardware Interface*.

Table 6.12 Interrupt Exception Handlers and Orders of Priority (SubCPU in EMU3)

Function/ Module	Interrupt Source Name	Level Interrupt*	EIINT Interrupt Channel	Source Code	Offset Address			Interrupt Priority Order (Initial Value)	Default Priority Order
					Direct Branching RINT=0*1	Direct Branching RINT=1*2	Reference to a Table*3		
ESINT (EMU software interrupt)	ESINT0 interrupt	√	0	1000	*4	*5	$+000_H$	0 to 15 (15)	*6
	ESINT1 interrupt	√	1	1001	*4	*5	$+004_H$	0 to 15 (15)	*6
	ESINT2 interrupt	√	2	1002	*4	*5	$+008_H$	0 to 15 (15)	*6
	ESINT3 interrupt	√	3	1003	*4	*5	$+00C_H$	0 to 15 (15)	*6
	ESINT4 interrupt	√	4	1004	*4	*5	$+010_H$	0 to 15 (15)	*6
	ESINT5 interrupt	√	5	1005	*4	*5	$+014_H$	0 to 15 (15)	*6
	ESINT6 interrupt	√	6	1006	*4	*5	$+018_H$	0 to 15 (15)	*6
	ESINT7 interrupt	√	7	1007	*4	*5	$+01C_H$	0 to 15 (15)	*6

Remark: For information about channels other than EIINT interrupt channels 0 to 7, see **Table 6.11**.

Note: Withdrawing an interrupt request in the form of a level interrupt requires clearing the given bit in the status register of the corresponding module from within the interrupt processing routing. For information about clearing the status register, see **Section 3.4.1, Synchronizing Completion of Store Instruction and Generation of Subsequent Instruction**. The EICn.EICTn bit is 1. Software cannot clear the EICN.EIRFn bit.

Note 1. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.

Note 2. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.

Note 3. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.

Note 4. The offset address does not vary depending on the channel. It should be selected from addresses $+100_H$ to $+1F0_H$ based on the order of priority.

Note 5. All the offset addresses are $+100_H$ regardless of the order of priority because of offset address reduction.

Note 6. Items are in descending order of priority from top to bottom.

6.5 Operation

6.5.1 External Interrupts(IRQ)

Externally input interrupts are IRQ. Four sensing methods are available for IRQ. Regarding the flow of interrupt, see **Section 6.5.6, Flow of Interrupt Processing**.

6.5.2 Inter-Processor Interrupts

For operational description, see **Section 3.2.3.1, Inter-Processor Interrupt Control Registers** and **Section 6.5.6.2, Inter-Processor Interrupt Flow**.

6.5.3 Software Interrupts

Regarding operations in relation to these interrupts, see **Section 6.2.8, SINTR0 to SINTR7 — Software Interrupt Register** for a description of the registers, and **Section 6.5.6, Flow of Interrupt Processing**.

6.5.4 EMU Software Interrupts

Regarding operations in relation to these interrupts, see **Section 6.2.9, ESINTR0 to ESINTR7 — EMU Software Interrupt Register** and **Section 6.5.6, Flow of Interrupt Processing**.

6.5.5 Merging of DTS Interrupts

Each set of 32 of the 128 transfer completion interrupts and 128 transfer count match interrupts from the DTS is gathered to produce a single interrupt signal.

When multiple interrupt source conditions are satisfied, only the single bit on the lower-order side of the corresponding bits in the status registers (PINT0 to PINT7) is set to judge which interrupt is accepted. For the flow of interrupts in the merging of DTS interrupts, see **Section 6.5.6, Flow of Interrupt Processing**.

Table 6.13 Interrupt Related Registers of the DTS

Interrupt Sources	Channel	Status Register	Clear Register
DTS transfer completion interrupt	0 to 31	PINT0	PINTCLR0
	32 to 63	PINT1	PINTCLR1
	64 to 95	PINT2	PINTCLR2
	96 to 127	PINT3	PINTCLR3
DTS transfer count match interrupt	0 to 31	PINT4	PINTCLR4
	32 to 63	PINT5	PINTCLR5
	64 to 95	PINT6	PINTCLR6
	96 to 127	PINT7	PINTCLR7

6.5.6 Flow of Interrupt Processing

6.5.6.1 Flow of Processing for External Interrupts

Figure 6.1 shows an example of the flow of IRQ (external interrupt) processing.

- Set the EXINTCTL register to select the method of detection (edge detection or level detection) for the IRQ.
- After detection of an IRQ, an interrupt request is issued to the INTC.
- After recovery on completion of interrupt processing within the INTC in the case of level detection, confirm that the IRQn pin is negated before issuing the instruction to return from the interrupt.
- After recovery on completion of interrupt processing within the INTC in the case of edge detection, clear the interrupt request bit in the EXINTSTR register, and then issue the instruction to return from the interrupt.

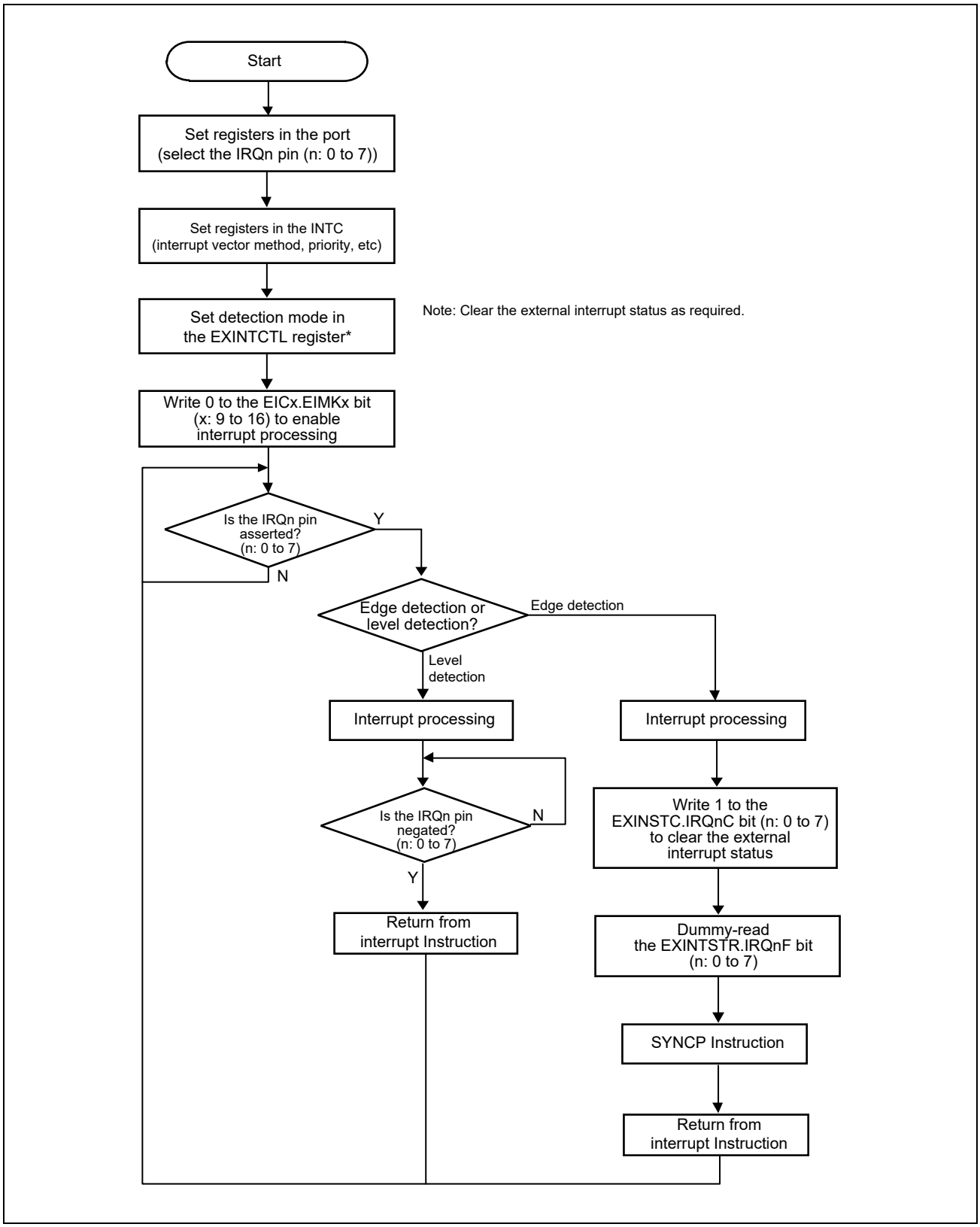


Figure 6.1 Example of External Interrupt Processing Flow

6.5.6.2 Inter-Processor Interrupt Flow

Figure 6.2 shows an example of the flow of inter-processor interrupt processing.

- Inter-processor interrupting generates an interrupt request by writing 1 to applicable bits of CPU to which interrupts of the inter-PE interrupt registers (IPIR0 to IPIR3) are requested.
- The settings of interrupt request of the inter-PE interrupt registers (IPIR0 to IPIR3) are automatically cleared to 0 after notification of an interrupt request is complete.

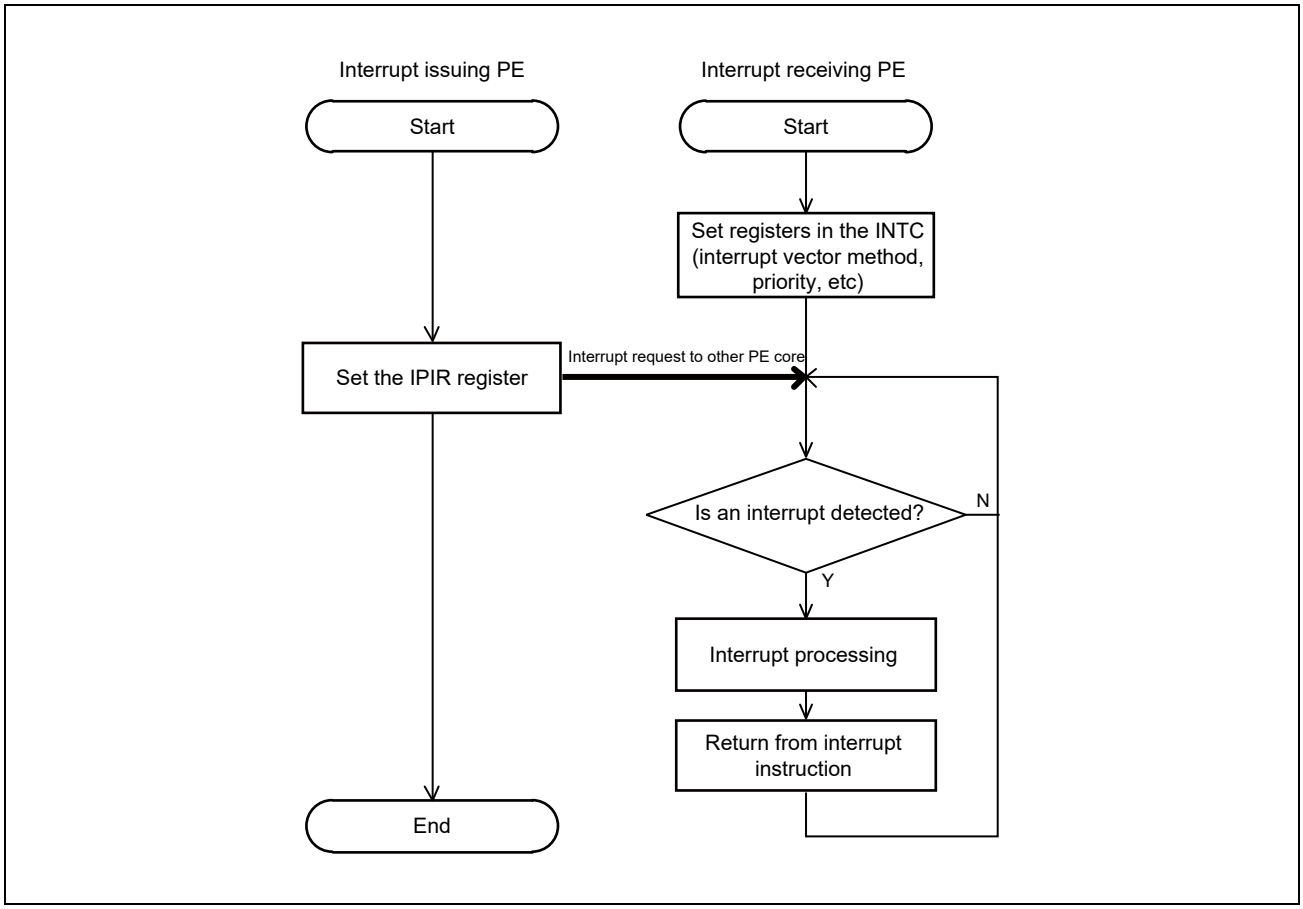


Figure 6.2 Example of Inter-Processor Interrupt Processing Flow

6.5.6.3 Software Interrupt Processing Flow

Figure 6.3 shows an example of the flow of software interrupt processing.

- Software interrupt requests are controlled by writing 00_H or 01_H to the counter registers (SINTR0 to SINTR7).
- Writing 00_H leads to the counter's value being decremented by 1.
- Writing 01_H leads to the counter's value being incremented by 1.
- If the incremented counter value is 1 or greater, an interrupt request for the INTC is generated.
- Decrement the counter by 1 during interrupt processing in the INTC, and if SINTRn is 00_H after issuing the instruction to return from the interrupt, wait for access to SINTRn. Wait for the writing of 01_H to SINTRn.
- The operations above are also performed for EMU software interrupt processing.

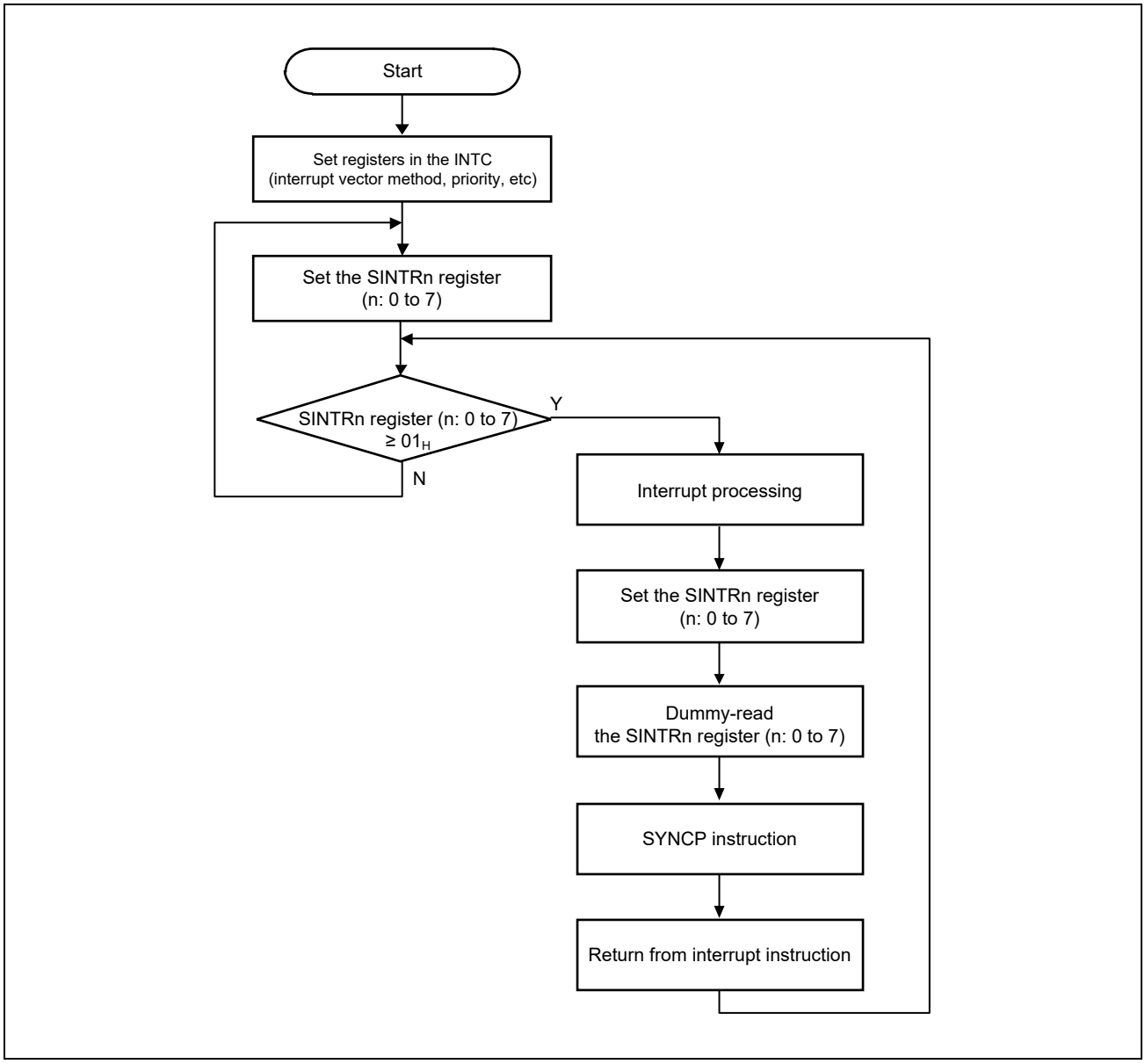


Figure 6.3 Example of Software Interrupt Processing Flow

6.5.6.4 Flow of Processing for DTS Interrupts

Figure 6.4 shows an example of the flow of DTS interrupt processing.

- In the case of a single interrupt request from a set of 32 actual sources
  - After the bit corresponding to the interrupt request in a PINTn register is set to 1, the interrupt request is output.
  - On completion of interrupt processing, 1 is written to the corresponding bit in a clearing register (PINTCLRn), the interrupt request is cleared before issuing the return from interrupt instruction, then waiting for a next interrupt commences.
- In the case of multiple interrupt requests from a set of 32 actual sources
  - The bit on the lower-order side among the multiple bits for the interrupt requests is extracted, only the corresponding bit in a PINTn register is set to 1, and the interrupt request is output.
  - On completion of interrupt processing, 1 is written to the corresponding bit in a clearing register (PINTCLRn), clearing the interrupt request bits, then issuing the return from interrupt instruction.
  - After clearing the interrupt request bits for which interrupt processing has proceeded, the bit on the lower-order side of the PINTn register that corresponds to an interrupt request is extracted and an interrupt request is issued in the same way as described for the previous operation.
  - This process is repeated until none of the 32 bits for the set of interrupt sources remains set.

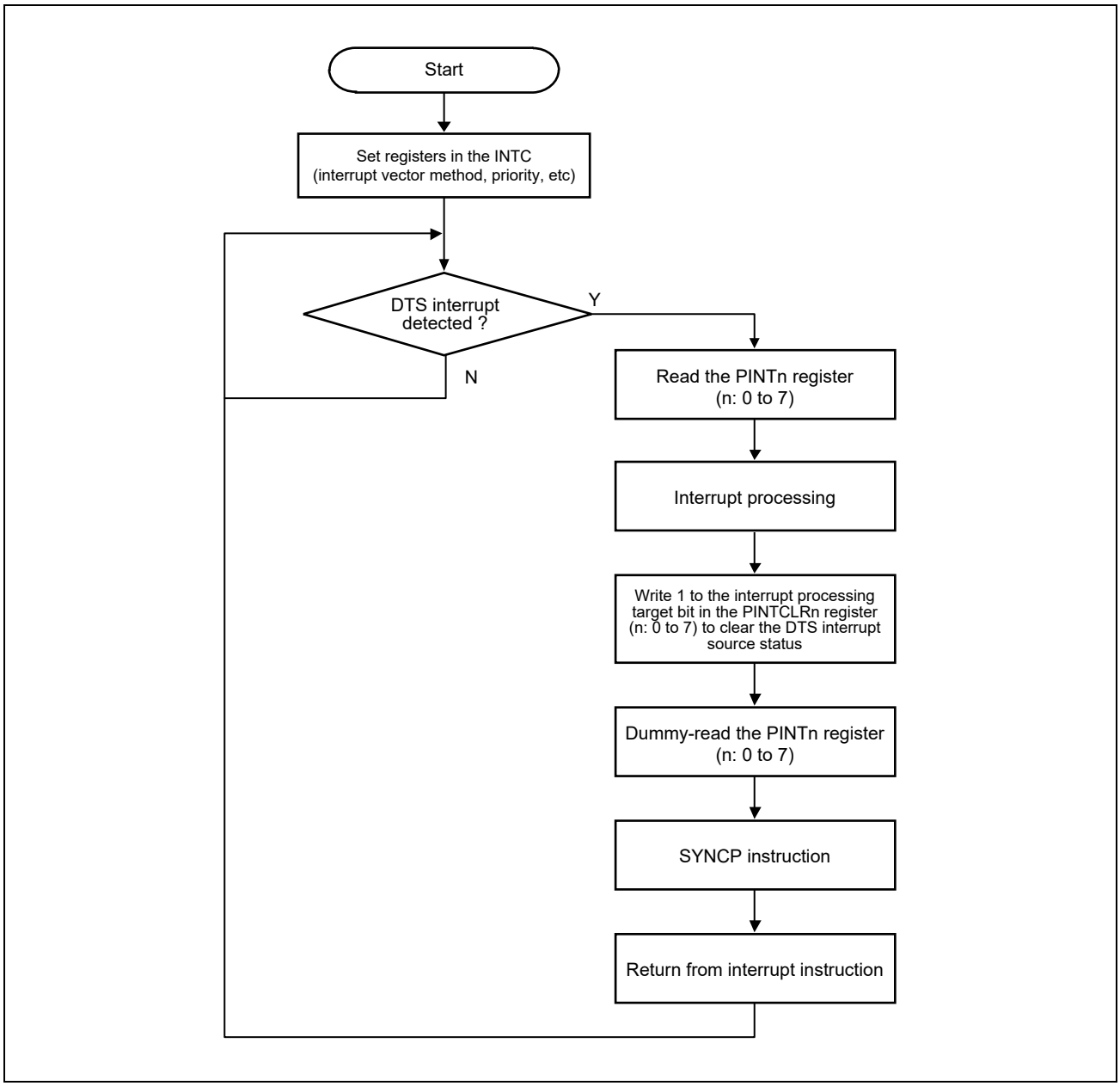


Figure 6.4 Example of DTS Interrupt Processing Flow

6.6 Interrupt Response Times

6.6.1 Interrupt Response Times

Table 6.14 Interrupt Response Times (Min)

Target	Interrupt Request Source		Number of Cycles for Processing				
	INTC Connection	Operating Clock	Synchro -nization	INTC2	INTC1	In CPU1/ CPU2/ SubCPU	Total (in the case of edge detection by fixed vector method a)
CPU1/ CPU2/ SubCPU	Directly input to INTC1	High-speed/Low-speed peripheral clock	0	—	2 × Iϕ <1 × Iϕ>	See the description under “In CPU1/CPU2/SubCPU” below.	7 × Iϕ
		Unmodulated High-speed peripheral clock	—				—
		Unmodulated Low-speed peripheral clock	5 × Pϕ <2 × Pϕ>				5 × Pϕ + 7 × Iϕ
	Input via INTC2	High-speed/Low-speed peripheral clock	0	3 × Pϕ + 1 × Iϕ <2 × Pϕ + 1 × Iϕ>	—		3 × Pϕ + 6 × Iϕ
		Unmodulated High-speed peripheral clock	4 × Pϕ <2 × Pϕ>				7 × Pϕ + 6 × Iϕ
		Unmodulated Low-speed peripheral clock	5 × Pϕ <2 × Pϕ>				5 × Pϕ + 6 × Iϕ

Note: The numbers in < > indicate the numbers of cycles in the case of level detection.

Note 1. Pϕ = 4Iϕ at 320 MHz, Pϕ = 3Iϕ at 240 MHz

Iϕ: CPU clock (CPU1, CPU2: CLK_CPU, SubCPU: CLK_EMU_H), Pϕ: High-speed peripheral clock (CLK_HSB)

Vector Method	Cache HIT/MISS	In CPU1/CPU2/SubCPU	Vector method
Fixed vector method	a) ISR ENTRY I$ HIT	5 × Iϕ	a) Fixed vector method
	b) ISR ENTRY I$ MISS	9 × Iϕ	
Vector table reference method	c) Vector code flash assigned, ISR ENTRY I$ HIT	13 × Iϕ	b) Vector table reference method Code flash assigned
	d) Vector code flash assigned, ISR ENTRY I$ MISS	18 × Iϕ	

Note: Pϕ = 4Iϕ at 320 MHz, Pϕ = 3Iϕ at 240 MHz

6.7 Data Transfer in Response to Interrupt Request Signals

An interrupt request signal can be used to activate the DMAC or DTS for handling data transfer. For details, see **Section 7, DMA Controller**.

Section 7 DMA Controller

7.1 Overview

7.1.1 Overview

The direct memory access (DMA) controller is used to move data without using the CPU.

DMA consists of two types of DMA transfer modules: DMAC and DTS. A DMAC includes registers for storing transfer information, and a DTS stores transfer information in the dedicated RAM (DTSRAM). DMA has two 8-channel DMAC modules and one 128-channel DTS module.

In this manual, DTFR denotes the function to select among hardware DMA transfer sources for a DMAC and retain the DMA transfer request, and DTSFSL denotes the function to retain a DMA transfer request for each DTS channel. The DTFR and DTSFSL can respectively handle 128 types of hardware DMA transfer sources.

The address space that can be used for DMA transfer is a 4 GB address space represented by a 32-bit address. For information about which resource is assigned to a particular area in the 4 GB address space and which area is accessible from DMA, see **Section 4, Address Space**.

7.1.2 Term Definition

Table 7.1 shows the terms used in this section.

Table 7.1 List of Term Definitions

Term	Meaning
DMA transfer	A general term for data transfer carried out by DMA.
DMA cycle	A series of actions that consist of reading an amount of data specified by the transfer size (8/16/32/64/128 bits) from the address specified by the source address and writing it to the address specified by the destination address. The first half of the DMA cycle (reading part) is called a read cycle, and the second half (writing part) is called a write cycle.
Hardware DMA transfer source	A trigger for a DMA transfer request given by an internal peripheral device.
Hardware DMA transfer request	A DMA transfer request generated by a hardware DMA transfer source
Software DMA transfer request	A DMA transfer request generated by software writing to a register.
DMA transfer request	A trigger to start DMA transfer with a DMAC and DTS.
Transfer information (TI)	The information required for DMA transfer, including the source address, destination address, transfer data size, and transfer count. The transfer information for a DTS is specifically termed as TI.
DTSRAM	RAM used by a DTS to store the transfer information.
Single transfer	DMA transfer consisting of one DMA cycle started by one DMA transfer request.
Block transfer 1	DMA transfer consisting of a number of DMA cycles specified by the transfer count in the transfer information, started by one DMA transfer request.
Block transfer 2	DMA transfer consisting of a number of DMA cycles specified by the address reload count in the transfer information, started by one DMA transfer request.
Block transfer	A general term for both block transfer 1 and block transfer 2.
Last transfer	The DMA cycle carried out when the transfer count in the transfer information is 1.
Address reload transfer	The DMA cycle carried out when the address reload count in the transfer information is 1 if the reload function 2 is used.
Suspension	An action of pausing DMA transfer during block transfer. You can resume DMA transfer after suspension.
Resume	An action of resuming suspended DMA transfer.
Transfer abort	An action of aborting DMA transfer in the middle. You cannot resume DMA transfer after that.

7.2 DMA Function

7.2.1 Basic Operation of DMA Transfer

7.2.1.1 Transfer Mode

DMA has three transfer modes.

Single Transfer

One DMA cycle is executed when a DMA transfer request is accepted.

Block Transfer 1

A number of DMA cycles specified in the transfer count register are executed when a DMA transfer request is accepted.

Block Transfer 2

A number of DMA cycles specified by the address reload count are executed when a DMA transfer request is accepted. If the address reload count is larger than the value in the transfer count register, a number of DMA cycles specified in the transfer count register are executed.

7.2.1.2 Executing a DMA Cycle

DMA always executes a write cycle after a read cycle is complete. For example, if the transfer data size is 128 bits, a write cycle is executed after a read cycle for the 128-bit data is finished. A write cycle never starts in the middle of a read cycle.

7.2.1.3 Updating Transfer Information

When a DMA cycle is executed, DMA updates transfer information as follows.

Source Address and Destination Address

Transfer information will be updated as described in **Table 7.2** according to the settings for the source address and destination address and the settings in the transfer control register such as the direction of counting for source and destination addresses and transfer data size.

Table 7.2 Updating the Source Address and the Destination Address

Direction of Counting	Transfer Data Size	Address after Update
Up	8 Bits	(address before update) + 0000_0001_H
	16 Bits	(address before update) + 0000_0002_H
	32 Bits	(address before update) + 0000_0004_H
	64 Bits	(address before update) + 0000_0008_H
	128 Bits	(address before update) + 0000_0010_H
Down	8 Bits	(address before update) – 0000_0001_H
	16 Bits	(address before update) – 0000_0002_H
	32 Bits	(address before update) – 0000_0004_H
	64 Bits	(address before update) – 0000_0008_H
	128 Bits	(address before update) – 0000_0010_H
Fixed	–	Same as the address before update.

When you use the reload function, a special update rule is applied other than the one described in **Table 7.2** for the last transfer and the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

Transfer Count and Address Reload Count

The transfer count is decremented by one for every DMA cycle.

The address reload count is decremented by one for every DMA cycle when the reload function 2 or block transfer 2 is used. When the reload function 2 or block transfer 2 is not used, it is not updated.

When you use the reload function, a special update rule is applied for the last transfer and the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

Other transfer information

Other transfer information is not updated during execution of a DMA cycle.

7.2.1.4 Last Transfer and Address Reload Transfer

The last transfer means a DMA cycle executed when the value in the transfer count register, which shows the remaining number of transfers, is one. The last transfer differs in operation compared to other DMA cycles as follows.

- The transfer completion flag (DCSTn.TC) is set when the last transfer is complete. (Only applicable for a DMAC)
- The channel operation enable (DCENn.DTE) bit is cleared when the last transfer is complete.(Only applicable for a DMAC. When the continuous transfer is disabled.)
- When the transfer completion interrupt output enable is set, a transfer completion interrupt is output when the last transfer is complete.
- When the reload function 1 is enabled, the reload function 1 is executed at the timing of the last transfer. For details, see **Section 7.2.3, Reload Function**.

The address reload transfer means a DMA cycle executed when the reload function 2 is enabled and the address reload count is one. The reload function 2 is executed during the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

7.2.1.5 Transfer Completion Interrupt and Transfer Count Match Interrupt Outputs

DMA can output a transfer completion interrupt and a transfer count match interrupt.

Transfer Completion Interrupt Output

When the transfer completion interrupt output enable (DTCTn.TCE) is set in the transfer control register, a DMAC requests a DMAC transfer completion interrupt when the last transfer is complete.

When the transfer completion interrupt output enable (DTTCTn.TCE) is set in the transfer control register, a DTS requests a DTS transfer completion interrupt when the last transfer is complete.

Transfer Count Match Interrupt Output

When the transfer count match interrupt enable (DTCTn.CCE) is set in the transfer control register, a DMAC requests a DMAC transfer count match interrupt when the DMA cycle in which the transfer count compare register and the transfer count have the same value is complete.

When the transfer count match interrupt enable (DTTCTn.CCE) is set in the transfer control register, a DTS requests a DTS transfer count match interrupt at the completion of the DMA cycle in which the transfer count compare register and the transfer count have the same value.

Figure 7.1 shows the operation of the transfer completion interrupt and the transfer count match interrupt.

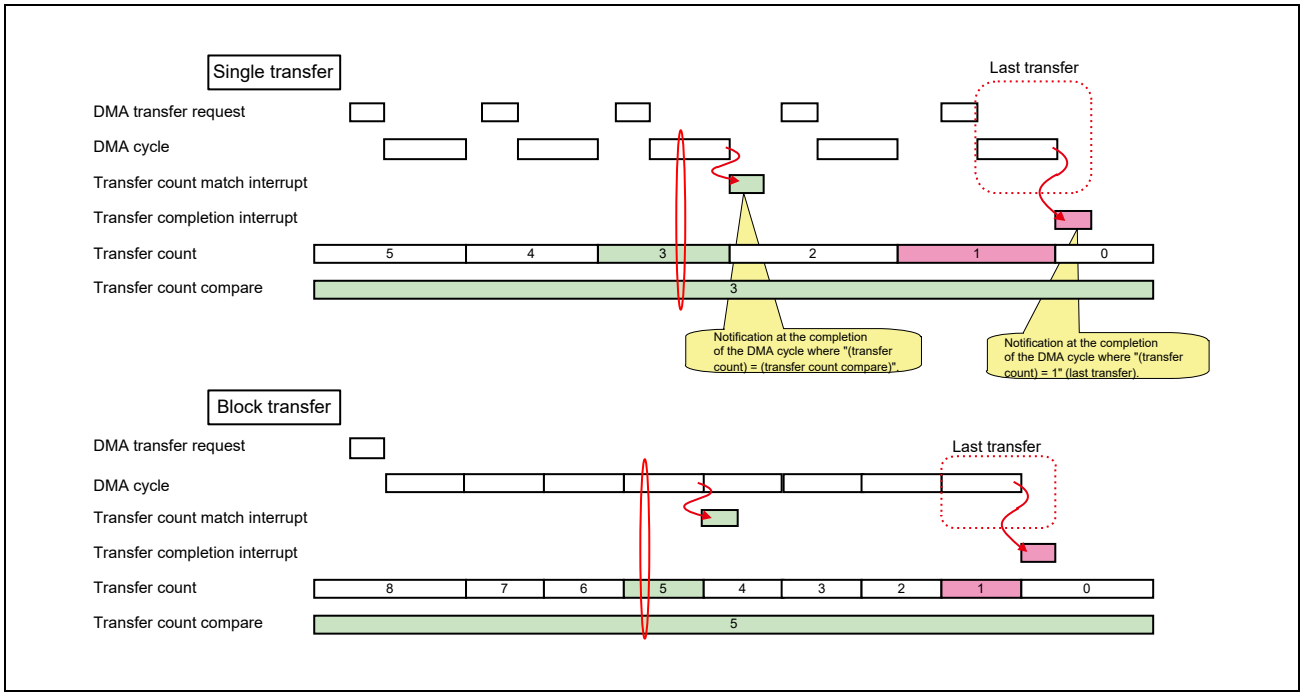


Figure 7.1 Transfer Completion Interrupt and Transfer Count Match Interrupt

7.2.1.6 Continuous Transfer

If the continuous transfer is not used, a DMAC sets the transfer completion flag (DCSTn.TC) and clears the channel operation enable (DCENn.DTE) bit when the last transfer is complete. In this case, a DMA transfer request is not accepted when the request is generated after the completion of the last transfer.

If the continuous transfer is used, the channel operation enable (DCENn.DTE) bit is not cleared when the last transfer is complete, and a DMA transfer request can be accepted even when the transfer completion flag is set. If DMA is used for a case where a specified number of DMA transfers are executed repetitively, software overhead associated with clearing the transfer completion flag and setting the channel operation enable bit after the completion of the last transfer can be reduced by using the continuous transfer.

The continuous transfer is enabled by setting the continuous transfer enable (DTCTn.MLE) in the DMAC transfer control register.

The continuous transfer is designed to work with the reload function 1. The continuous transfer function itself does not update the source address register, destination address register, and transfer count register. If, after the last transfer is complete, you want to restore the source address register, destination address register, and transfer count register to the state before the DMA transfer starts, use the reload function 1 and set the values of the source address register, destination address register, and the transfer count register before the DMA transfer starts to the reload source address register, reload destination address register, and reload transfer count register respectively.

A DTS does not have a setting corresponding to the continuous transfer enable (DTCTn.MLE) for a DMAC. This is because a DTS does not have bits like the transfer completion flag (DCSTn.TC) and the channel operation enable (DCENn.DTE) a DMAC has.

A DTS does not start DMA transfer when a DMA transfer request is generated while the transfer count is 0. (This corresponds to the case for a DMAC where the continuous transfer is not used.)

If the reload function 1 is used for a DTS and the value other than 0 is reloaded to the transfer count when the last transfer is complete, DMA transfer can start when the next DMA transfer request is accepted. (This corresponds to the case for a DMAC where the continuous transfer is used.)

Figure 7.2 shows an operation of continuous transfer by a DMAC.

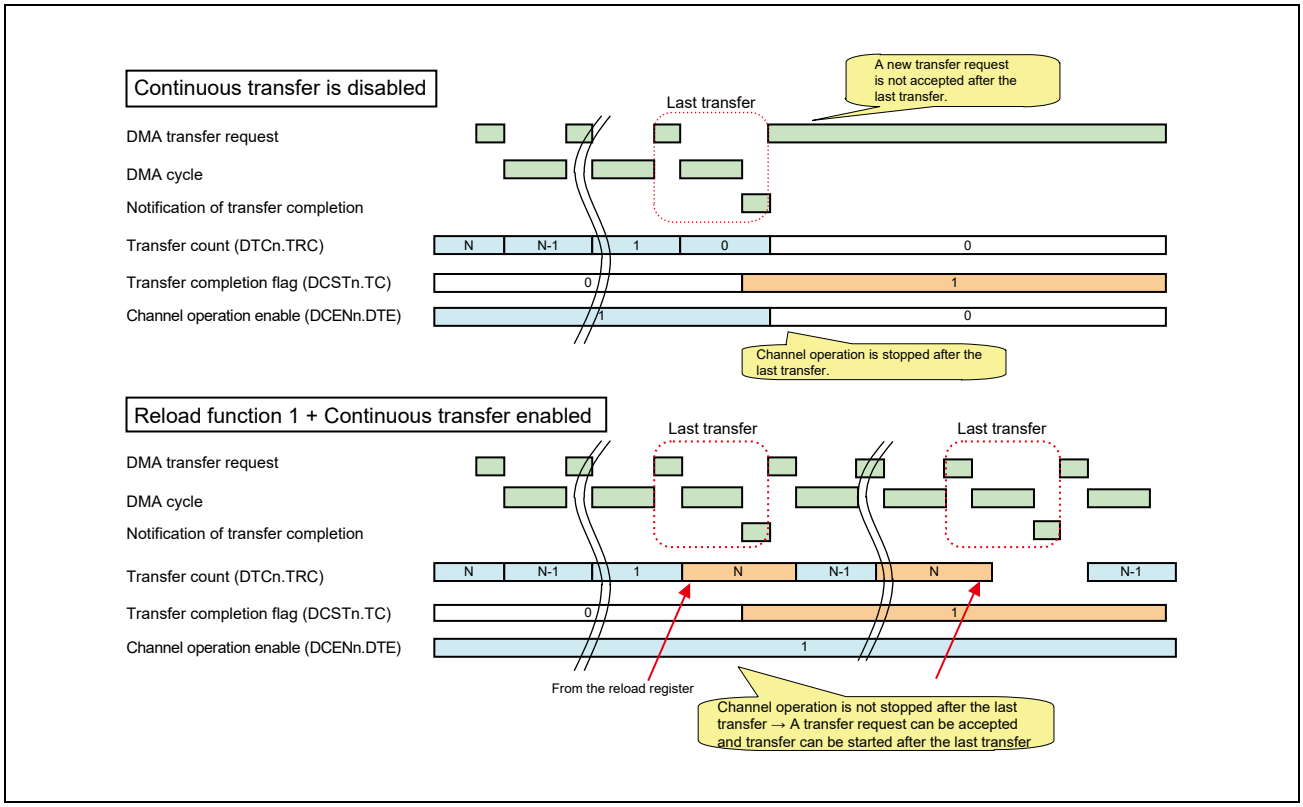


Figure 7.2 Operation of Continuous Transfer by a DMAC

7.2.2 Channel Priority Order

This subsection explains arbitration between multiple DMA channels.

7.2.2.1 DMAC Channel Arbitration

A DMAC select one channel out of eight channels with arbitration. Arbitration is done according to the fixed priority order. The priority order is “channel 0 > channel 1 > channel 2 > channel 3 > channel 4 > channel 5 > channel 6 > channel 7” for DMAC0, and “channel 8 > channel 9 > channel 10 > channel 11 > channel 12 > channel 13 > channel 14 > channel 15” for DMAC1.

Arbitration is done for every DMA cycle. No arbitration occurs between the read and write of a DMA cycle.

If, at the timing when one DMA cycle in the middle of a block transfer of a channel is complete, there is a DMA transfer request from a channel with a higher priority than the channel, a DMA cycle of the channel with the higher priority will be executed next as the result of arbitration.
If a DMAC executes the block transfer 1 or block transfer 2, DMAC channel arbitration is done for every DMA cycle, and possibly a DMA cycle of another DMAC channel with a higher priority can cut in.

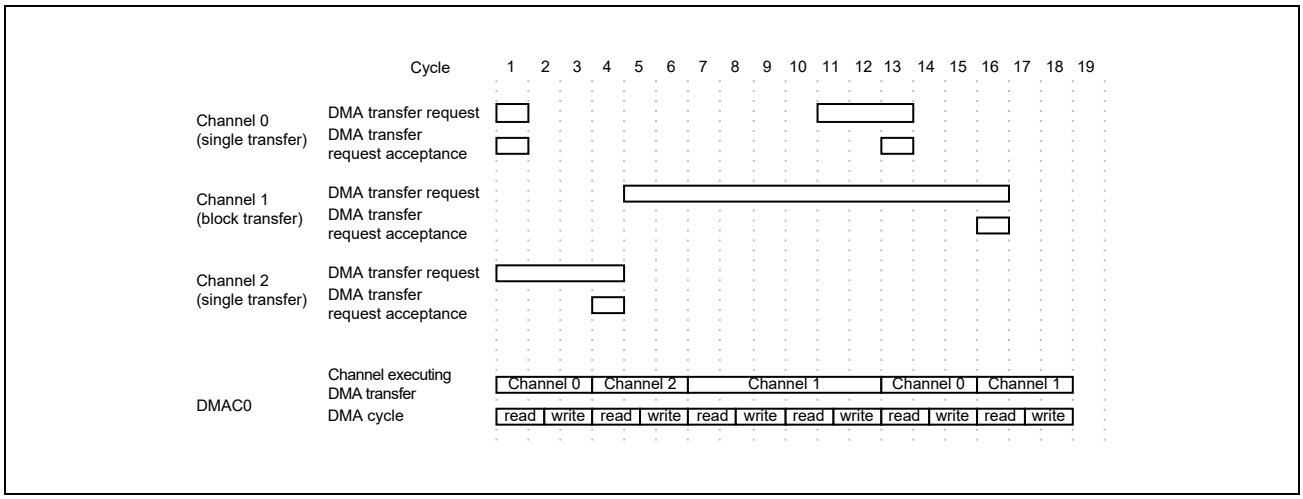


Figure 7.3 DMAC Channel Arbitration

Cycle numbers shown in **Figure 7.3** are for explanation purpose only. They do not indicate an actual number of cycles necessary for executing DMA transfer.

In **Figure 7.3**, DMA transfer requests for channels 0 and 2 are generated at Cycle 1. As a result of arbitration, a DMA cycle for channel 0 starts because its priority is higher. At Cycle 4, a DMA cycle for channel 2 starts. At Cycle 5, a DMA transfer request for channel 1 is generated. The DMA cycle for channel 2 is still ongoing and no arbitration is done at this point. At Cycle 7, a DMA cycle for channel 1 starts. Channel 1 uses block transfer. Another DMA cycle continues at Cycle 10 because there are no DMA transfer requests from other channels. At Cycle 11, a DMA transfer request for channel 0 is generated. The DMA cycle for channel 1 is still ongoing and no arbitration is done at this point. At Cycle 12, the DMA cycle for channel 1 is complete. At Cycle 13, a DMA cycle for channel 0 starts as a result of arbitration between DMA channels 0 and 1.

It should be noted that, even though a block transfer of channel 1 has been already started, a DMA cycle of not channel 1 but channel 0 is executed at Cycle 13 because the priority of the latter is higher. At Cycle 15, the DMA cycle for channel 0 is complete. At Cycle 16, a DMA cycle for channel 1 starts again. At Cycle 18, the last DMA cycle of the block transfer of channel 1 is complete.

7.2.2.2 DTS Channel Arbitration

If there are DMA transfer requests from multiple DTS channels, the DTSFSL arbitrates those DTS channels. For each DTS channel, a priority can be selected from four levels using DTS channel priority setting registers.

If there are DMA transfer request from multiple DTS channels, the arbitration is done as follows.

1. A channel with a higher priority level in the setting of DTS channel priority setting registers has a priority.
2. If two channels have the same priority level in the setting of DTS channel priority setting registers, a channel with a lower channel number has a priority.

The DTSFSL sends the DTS a DMA transfer request for the channel selected by arbitration. The DTS executes DMA transfer when it accepts the DMA transfer request.

Unlike DMA transfer with a DMAC, DMA transfer with a DTS does not allow arbitration between DTS channels in the middle of a block transfer. That means, even if a DMA transfer request with a higher priority comes during a block transfer for a channel with a lower priority, the DMA transfer with a higher priority does not start until the current block transfer for the channel with a lower priority is complete*[1].

Note 1. The timing of completion of the block transfer is when the last transfer for the block transfer 1 or the last transfer or address reload transfer for the block transfer 2 occurs.

When a DTS executes the block transfer 1 or block transfer 2, a DMA cycle of a DTS channel with a higher priority does not take over the ongoing block transfer until the last transfer.

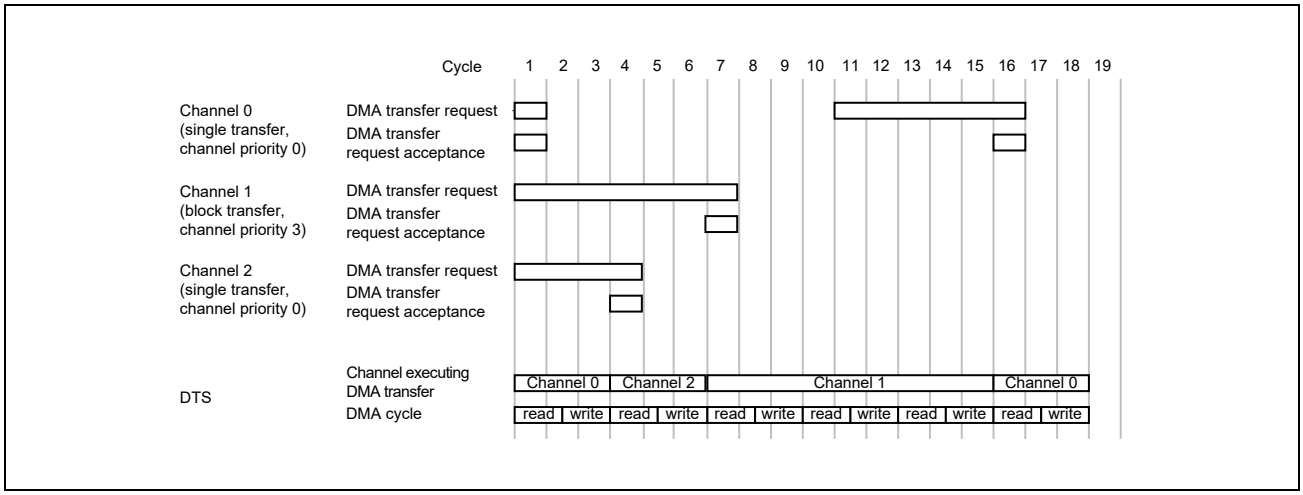


Figure 7.4 DMAC Channel Arbitration

Cycle numbers shown in **Figure 7.4** are for explanation purpose only. They do not indicate an actual number of cycles necessary for executing DMA transfer.

In **Figure 7.4**, DMA transfer requests for channels 0, 1, and 2 are generated at Cycle 1. The channel priority for channels 0 and 2 is 0 and is higher than the channel priority for channel 1, which is 3. In addition, if two channels have the same priority, the channel with the smaller channel number has a higher priority. Consequently, the priority order for arbitration is “channel 0 > channel 2 > channel 1”, and a DMA cycle for channel 0 starts because its priority is the highest. At Cycle 4, as a result of arbitration between channels 1 and 2, a DMA cycle for channel 2 starts. At Cycle 7, a DMA cycle for channel 1 starts. Channel 1 uses block transfer. Another DMA cycle continues at Cycle 10 because there are no DMA transfer requests from other channels. At Cycle 11, a DMA transfer request for channel 0 is generated. The DMA cycle for channel 1 is still ongoing and no arbitration is done until the block transfer of channel 1 is complete.

At Cycle 15, the block transfer of channel 1 is complete. At Cycle 16, a DMA cycle for channel 0 starts.

7.2.2.3 Interface Arbitration

DMAC0, DMAC1, and DTS work independently and execute DMA transfer.

If there is a conflict between requests from DMAC0, DMAC1, and DTS, round-robin arbitration is performed.

7.2.3 Reload Function

7.2.3.1 Overview of the Reload Function

The reload function updates a portion of transfer information, more specifically, the source address, destination address, transfer count, and address reload count, to the predefined values during DMA transfer.

The reload function has two types of functions: reload function 1 and reload function 2.

7.2.3.2 Operation of Reload Function 1

When the reload function 1 is enabled, actions described in **Table 7.3** are executed at the timing of the last transfer according to the reload function 1 setting.

Table 7.3 Operation of Reload Function 1

Reload Function 1 Setting (DTCTn.RLD1M[1:0])	Register	Action at the Last Transfer
00 (Reload function 1 disabled)	Source address	Not reloaded.
	Destination address	Not reloaded.
	Transfer count	Not reloaded.
	Address reload count	Not reloaded.
01 (Reload function 1 enabled. Reloading source address and transfer count.)	Source address	The reload source address is copied to this.
	Destination address	Not reloaded.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded. • If the reload function 2 is enabled: The reload address reload count is copied to this.
10 (Reload function 1 enabled. Reloading destination address and transfer count.)	Source address	Not reloaded.
	Destination address	The reload destination address is copied to this.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded. • If the reload function 2 is enabled: The reload address reload count is copied to this.
11 (Reload function 1 enabled. Reloading source address, destination address, and transfer count.)	Source address	The reload source address is copied to this.
	Destination address	The reload destination address is copied to this.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded • If the reload function 2 is enabled: The reload address reload count is copied to this.

Figure 7.5 shows an operation of the reload function 1.

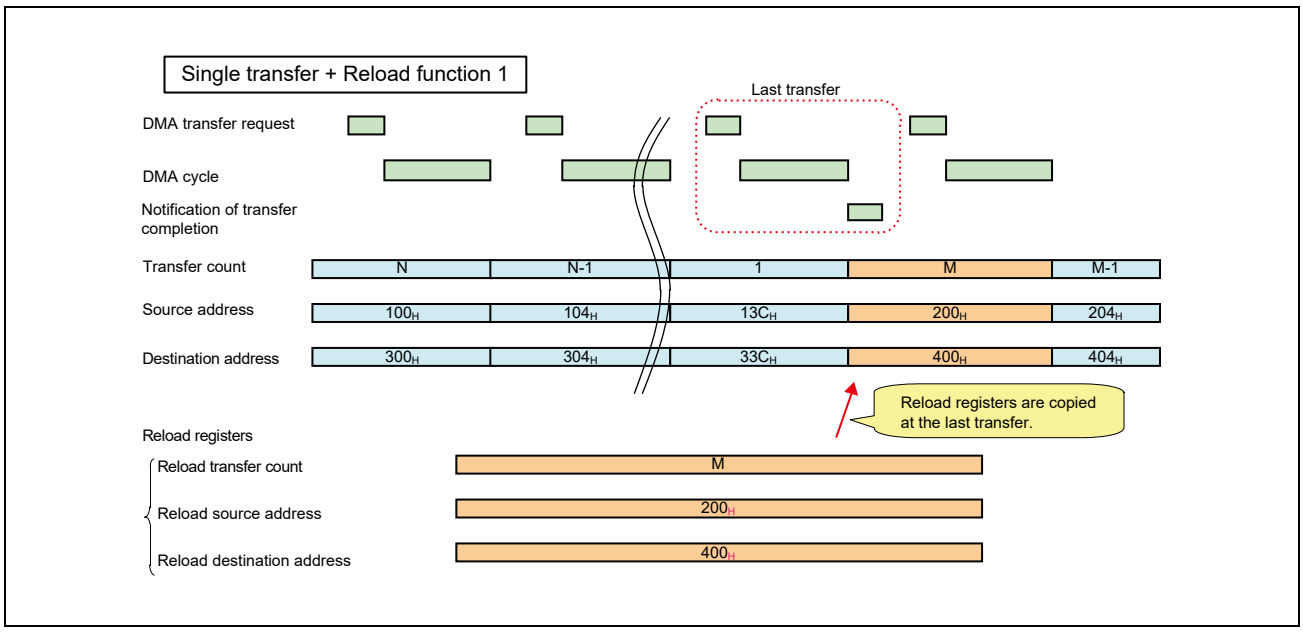


Figure 7.5 Operation of Reload Function 1

7.2.3.3 Reload Function 2

When the reload function 2 is enabled, actions described in **Table 7.4** are executed at the timing of the address reload transfer according to the reload function 2 setting.

Table 7.4 Operation of Reload Function 2

Reload Function 2 Setting (DTCTn.RLD2M[1:0])	Register	Action at the Address Reload Transfer
00 (Reload function 2 disabled)	Source address	Not reloaded.
	Destination address	Not reloaded.
	Address reload count	Not reloaded.
01 (Reload function 2 enabled. Reloading source address.)	Source address	The reload source address is copied to this.
	Destination address	Not reloaded.
	Address reload count	The reload address reload count is copied to this.
10 (Reload function 2 enabled. Reloading destination address.)	Source address	Not reloaded.
	Destination address	The reload destination address is copied to this.
	Address reload count	The reload address reload count is copied to this.
11 (Reload function 2 enabled. Reloading source address and destination address.)	Source address	The reload source address is copied to this.
	Destination address	The reload destination address is copied to this.
	Address reload count	The reload address reload count is copied to this.

Figure 7.6 shows an operation of the reload function 2.

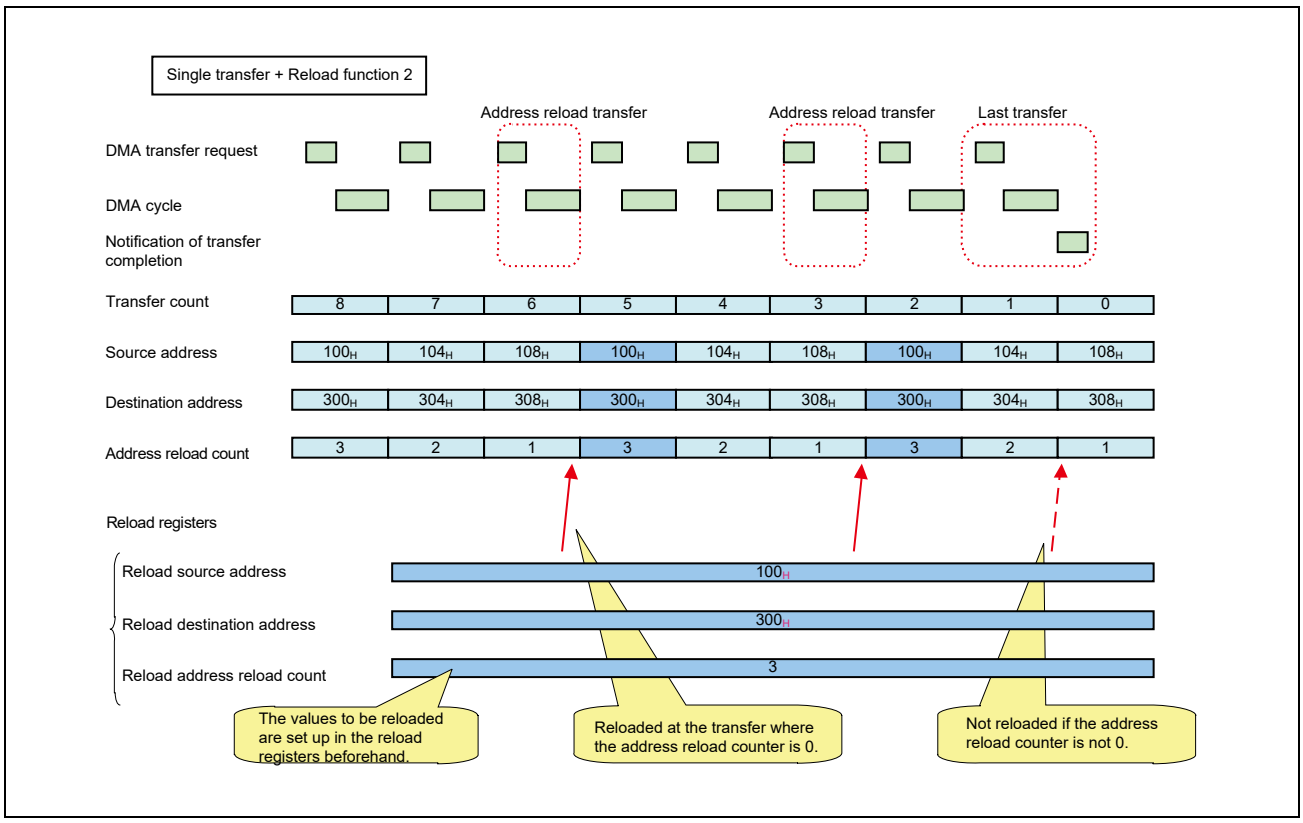


Figure 7.6 Operation of Reload Function 2

Figure 7.7 shows an operation when both the reload function 1 and the reload function 2 are used simultaneously.

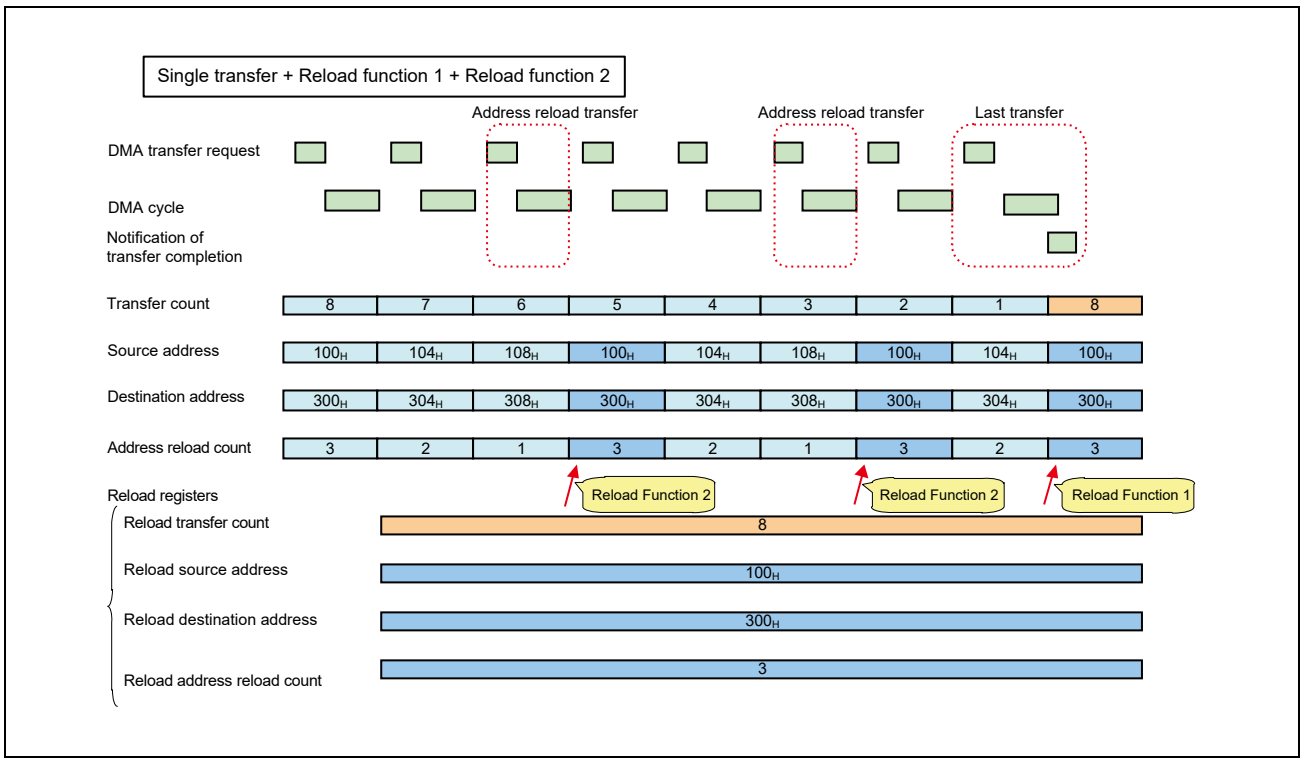


Figure 7.7 Operation when Combining Reload Function 1 and Reload Function 2

7.2.3.4 Timing of Setting DMAC Reload Registers

You can set up the reload source address register, reload destination address register, and reload transfer count register any time (even during DMA transfer). As an exception, if you update the reload source address register, reload destination address register, and reload transfer count register during DMA transfer, there may be a conflict between reloading at the last transfer or address reload transfer and updating the reload register. In order to avoid this conflict, setting up reload registers must be completed before the last transfer or address reload transfer starts.

If you need to update the reload source address register, reload destination address register, and reload transfer count register during DMA transfer, one way to know the right timing of update is to use a DMA transfer count match interrupt. In this case, you must set up the DMA transfer count compare register (DTCCn) so that you can have enough margin for the time necessary to update the reload registers.

7.2.3.5 Timing of Setting DTS Reload Registers

It should be noted that the right timing of setting up the reload source address information, reload destination address information, and reload transfer count information differs depending on the transfer mode.

In single-transfer mode, the TI fetched at the beginning of the last transfer or address reload transfer is used for reload at the completion of the DMA cycle. Therefore, if you use the reload function for single transfer, the reload source address information, reload destination address information, and reload transfer count information in the TI must be set up before the beginning of the last transfer or address reload transfer.

During block transfer, TI is fetched only at the beginning of DMA transfer. The TI fetched at the beginning of the DMA transfer is used for reload at the last transfer or address reload transfer. Therefore, if you use the reload function for block transfer, the reload source address information, reload destination address information, and reload transfer count

information in the TI must be set up before the beginning of the DMA transfer. If you update the reload source address information, reload destination address information, and reload transfer count information in the TI in the middle of a block transfer, those new settings will not be used for reload at the completion of the block transfer.

7.2.4 Chain Function

7.2.4.1 Overview

DMA offers a function called chain function. If you use the chain function, the completion of the DMA cycle or last transfer for one channel can trigger a DMA transfer request for another channel. A DMA transfer request for another channel initiated by the chain function is called a chain request.

You can select the condition for generating a chain request from the following two options.

- Always chain: A chain request is generated at the completion of every DMA cycle.
- Chain at the last transfer: A chain request is generated at the completion of the last transfer.

Figure 7.8 shows an operation of the case “always chain”.

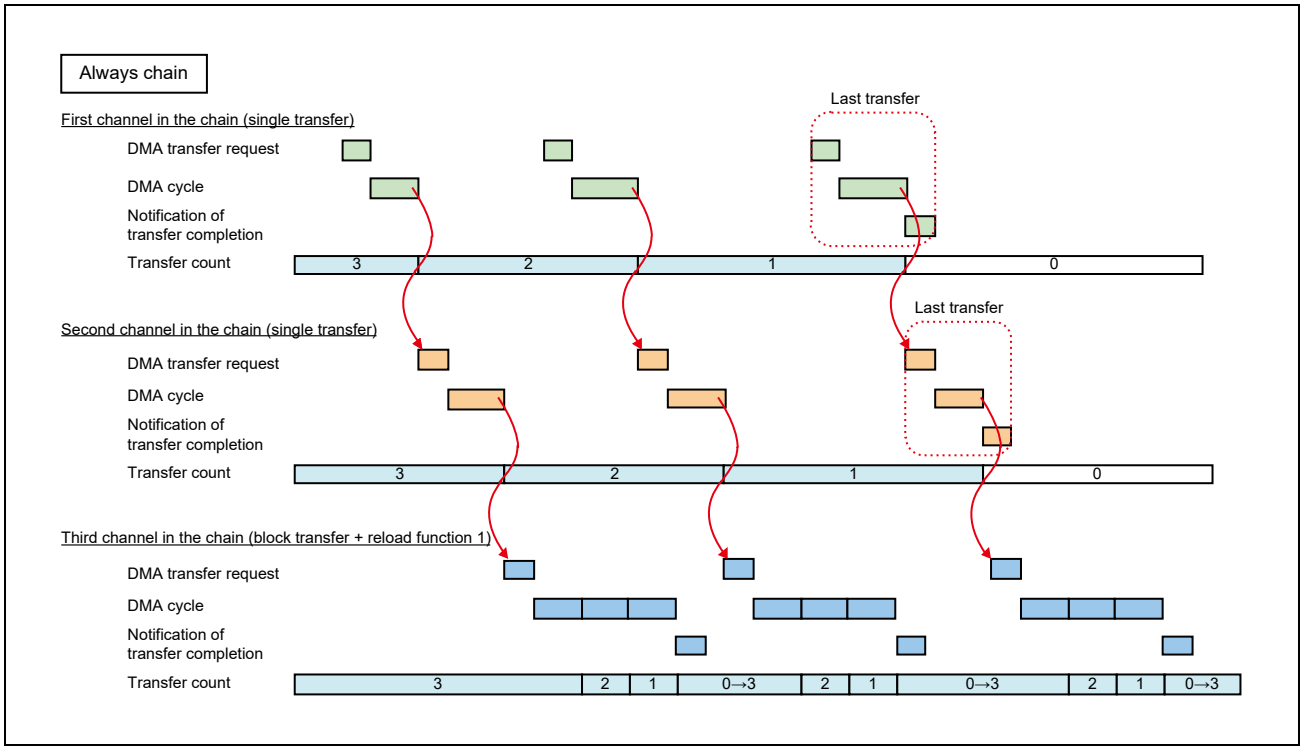


Figure 7.8 Operation of the Case “Always Chain”

Figure 7.9 shows an operation of the case “chain at the last transfer”.

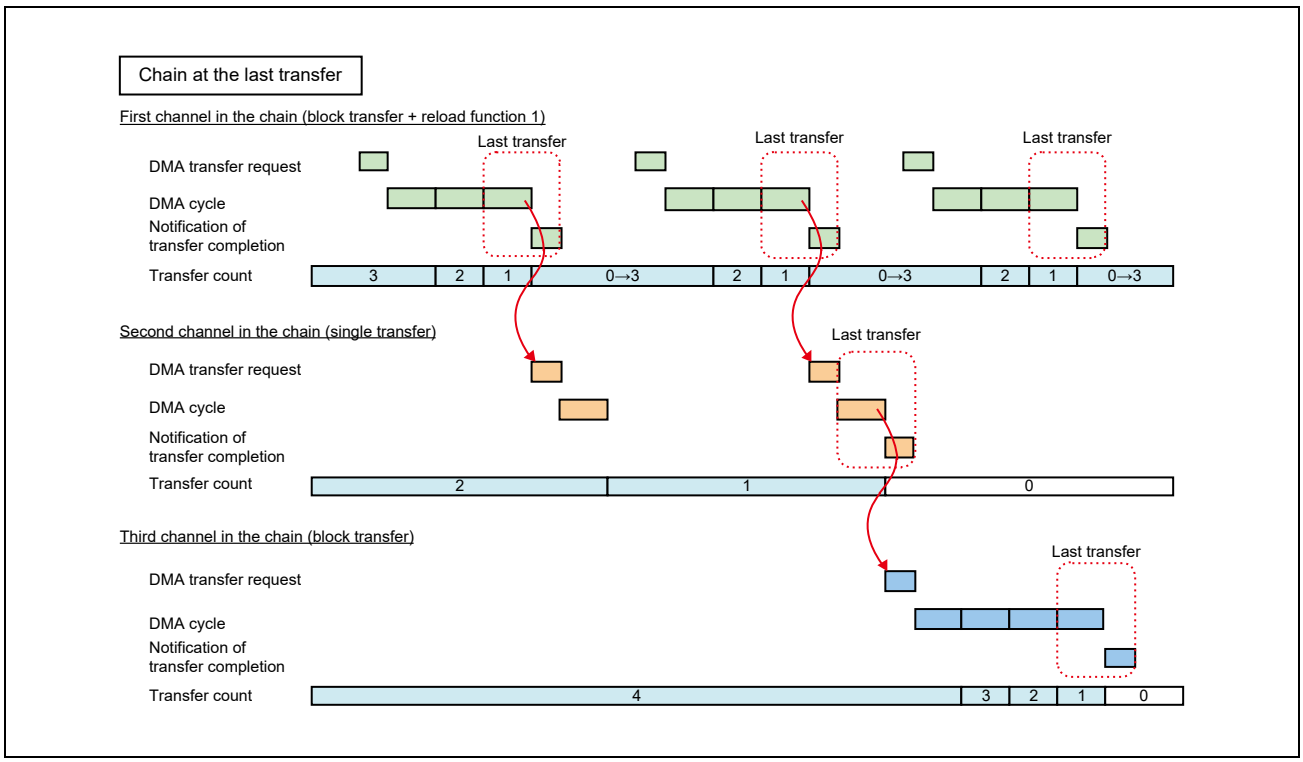


Figure 7.9 Operation of the Case “Chain at the Last Transfer”

7.2.4.2 Setting Up the Chain Function

For a DMAC, you need to write to the chain enable (DTCTn.CHNE) and the next channel in the chain selection (DTCTn.CHNSEL) in the DMAC transfer control register in order to set up the type of chain function and the next channel number in the chain.

For a DTS, you need to write to the chain enable (DTTCTnnn.CHNE) and the next channel in the chain selection (DTTCTnnn.CHNSEL) in the DTS transfer control register in order to set up the type of chain function and the next channel number in the chain.

7.2.4.3 Caution for Using the Chain Function

The chain function sets the software DMA transfer request flag of the next channel in the chain as a part of its function. Therefore, you need to set up the channel settings of the next channel in the chain in the same way as when the software DMA transfer request is used. If you specify a channel using the hardware DMA transfer request for the next channel in the chain, the chain function does not work.

A channel and its next channel in the chain must belong to the same module (DMAC0, DMAC1, and DTS). You cannot specify a channel in another module for its next channel in the chain.

7.2.5 DMAC Operation

7.2.5.1 Types of DMA Transfer Requests and Assigning DMA Transfer Requests

A DMAC starts DMA transfer by accepting a hardware DMA transfer request or software DMA transfer request. The DMA transfer request selection assignment (DRS) bit in the DMAC transfer control register (DTCTn) determines whether a hardware DMA transfer request or a software DMA transfer request is used.

In the case of a hardware DMA transfer request for a DMAC, one out of 128 hardware DMA transfer sources is selected and assigned for each channel of the DMAC in the DTFR. This assignment is configured in the DTFR setting registers.

7.2.5.2 Generating and Accepting a Software DMA Transfer Request

By setting the software DMA transfer request flag (SR) in the DMAC transfer status register (DCSTn) using the DMAC transfer status set register (DCSTSn), a software DMA transfer request can be generated.

The software DMA transfer request flag is automatically cleared when the DMAC processes the DMA transfer request. The timing when the software DMA transfer request flag is automatically cleared differs depending on the transfer mode of the DMA transfer to be executed.

- In single-transfer mode, the software DMA transfer request flag is cleared whenever the software DMA transfer request is accepted.
- In block transfer 1 mode, the software DMA transfer request flag is cleared when the last transfer starts.
- In block transfer 2 mode, the software DMA transfer request flag is cleared when the last transfer or address reload transfer starts.

The software DMA transfer request flag can also be cleared by software using the DMAC transfer status clear register (DTSTCn). When you abort a DMA transfer of a DMAC channel, you must clear the software DMA transfer request flag.

7.2.6 DTS Operation

7.2.6.1 Types of DMA Transfer Requests and Assigning DMA Transfer Requests

A DTS starts DMA transfer by accepting a hardware DMA transfer request or software DMA transfer request.

A transfer request for a DTS is retained in the transfer request pending bit of the DTSFSL for each channel. As for the DTSFSL, both a hardware DMA transfer request and a software DMA transfer request are retained in the same transfer request pending bit. When executing DMA transfer, a DTS does not care whether a DMA transfer request is a hardware DMA transfer request or software DMA transfer request.

In the case of a hardware DMA transfer request for a DTS, each of 128 hardware DMA transfer sources is assigned to one of 128 channels in the DTSFSL in the fixed manner. You cannot change this assignment by, for example, register settings.

7.2.6.2 Generating and Accepting a DMA Transfer Request

When the DTSFSL detects a hardware DMA transfer source input, the DTSFSL sets the transfer request pending bit and retains the hardware DMA transfer source as a DMA transfer request. If the transfer request pending bit is set and the transfer request enable bit (DTFSLnnn.REQEN) in the DTSFSL operation setting register is set, the DTSFSL notifies the DTS of the DMA transfer request.

Software can also generate a DMA transfer request by setting the transfer request pending bit (DTFSTnnn.DRQ) using the DTSFSL transfer request set register (DTFSSnnn).

The DTSFSL can retain only one DMA transfer request per channel. If, while the transfer request pending bit for a channel is set, a new hardware DMA transfer source input for the same channel comes, the new hardware DMA transfer source input is ignored.

When the DTS accepts a DMA transfer request, it notifies of the acceptance of the DMA transfer request.

The transfer request pending bit is automatically cleared when the DTS accepts the DMA transfer request. The DTSFSL clears the transfer request pending bit automatically when the DTS accepts the DMA transfer request regardless of the type of the DMA transfer to be executed by the DTS.

The transfer request pending bit can also be cleared using the DTSFSL transfer request clear register (DTFSCnnn). If the transfer request pending bit of a channel is cleared before the DTS accepts the DMA transfer request, DMA transfer of the channel is not executed.

7.2.6.3 Executing DMA Transfer

When the DTS accepts a DMA transfer request for a channel, the DTS executes DMA transfer of the channel. If there are DMA transfer requests from multiple channels, the DTSFSL arbitrates the DTS channels and picks up one channel for a DMA transfer request.

While the DTS is executing DMA transfer, the DTS transfer status (DTSSTS.DTSACT) bit in the DTS status register is set. In addition, the channel number of the currently ongoing DMA transfer is set in the DTS transfer channel (DTSSTS.DTSACH).

When the DMA transfer is complete or aborted because of DMA transfer error or writing to registers and no channel is currently executing DMA transfer, the DTS transfer status (DTSSTS.DTSACT) bit is cleared.

7.2.6.4 DTSRAM Access

A DTS accesses the DTSRAM when DMA transfer starts and finishes.

A DTS's action of reading transfer information from the DTSRAM when DMA transfer starts is called TI fetch.

A DTS's action of updating the transfer information on the DTSRAM when DMA transfer finishes is called TI write back.

A single transfer performs a TI fetch at the beginning of a DMA cycle and a TI write back at the end of a DMA cycle.

A block transfer performs a TI fetch at the beginning of the first DMA cycle and a TI write back at the end of the DMA cycle that satisfies the block transfer completion condition (the last transfer or address reload transfer).

Therefore, in the case of single transfer, the transfer information on the DTSRAM is updated for each DMA cycle. In the case of block transfer, the transfer information on the DTSRAM is updated after the completion of the block transfer. If software reads the transfer information on the DTSRAM during execution of a block transfer, the transfer information at the beginning of the block transfer is read.

7.3 Suspension, Resume, Transfer Abort, and Clearing a DMA Transfer Request

7.3.1 DMA Suspension and Resume by Software Control

The DMA control register (DMACTL) is used to suspend DMA transfer for all channels.

When the DMA suspension bit (DMACTL.DMASPD) in the DMA control register is set, DMA puts all channels into the suspended state. If all channels are in the suspended state and the DMA suspension bit (DMASPD) in the DMA control register is cleared, DMA restores all channels from the suspended state to the normal state and resumes the DMA transfer of the suspended channel.

When all channels are put into the suspended state, DMA transfer is suspended for all channels without changing the value of the DCENn.DTE bit of each DMAC channel and the DTSCTL1.DTSUST bit of the DTS.

CAUTION

In order to suspend or resume the DMA transfer being executed by a DTS, it is necessary to perform enabling or disabling operation by using the DMA transfer request enable bit (DTFSLnnn.REQEN). For details, refer to notes in **Section 7.3.3, Suspension, Resume, and Transfer Abort of a DTS**.

7.3.2 Suspension, Resume, and Transfer Abort of a DMAC Channel

You can suspend the DMA transfer of a DMAC channel by clearing the channel operation enable bit (DCENn.DTE) in the DMAC channel operation enable setting register or by setting the DMA suspension bit (DMACTL.DMASPD) in the DMA control register. If a DMA cycle is ongoing, the DMA transfer is suspended after the currently ongoing DMA cycle is finished. If you set the DCENn.DTE bit again or clear the DMACTL.DMASPD bit during suspension, the DMA transfer of the suspended DMA channel is resumed.

If you want to abort the currently ongoing DMA transfer of a DMAC channel, clear the channel operation enable bit (DCENn.DTE) in the DMAC channel operation enable setting register, and then clear the hardware DMA transfer request in the DTFR in the case of a hardware DMA transfer request, and clear the software DMA transfer request flag (DCSTn.SR) using the software DMA transfer request flag clear bit (DCSTCn.SRC) in the DMAC transfer status clear register in the case of a software DMA transfer request.

Figure 7.10 shows an example of suspension, resume, and transfer abort of a DMAC channel.

In **Figure 7.10**, both channels 0 and 1 are executing block transfer. At time tick 1, DMA transfer of channel 1 starts. At time tick 2, a DMA transfer request for channel 0 is accepted. As a result of DMAC channel arbitration, DMA transfer of channel 0 starts because channel 0 has a higher priority than channel 1. At time tick 3, the last transfer of channel 0 is complete, and the remaining DMA transfer in the block transfer of channel 1 starts. At time tick 4, the last transfer of channel 1 is complete. After time tick 5, DMA transfer of channel 0 and DMA transfer of channel 1 are executed similarly. At time tick 7, the DMA transfer of channel 0 is suspended and, as a result of DMAC channel arbitration, the DMA transfer of channel 1 starts. At time tick 8, the last transfer of channel 1 is complete, and then, at time tick 9, the DMA transfer of channel 0 resumes. At time tick 10, the last transfer of channel 0 is suspended again, and then, at time tick 11, the DMA transfer of channel 0 is aborted. At time tick 12, the suspended state for channel 0 is cleared, but DMA transfer is not executed because the DMA transfer is aborted at time tick 11.

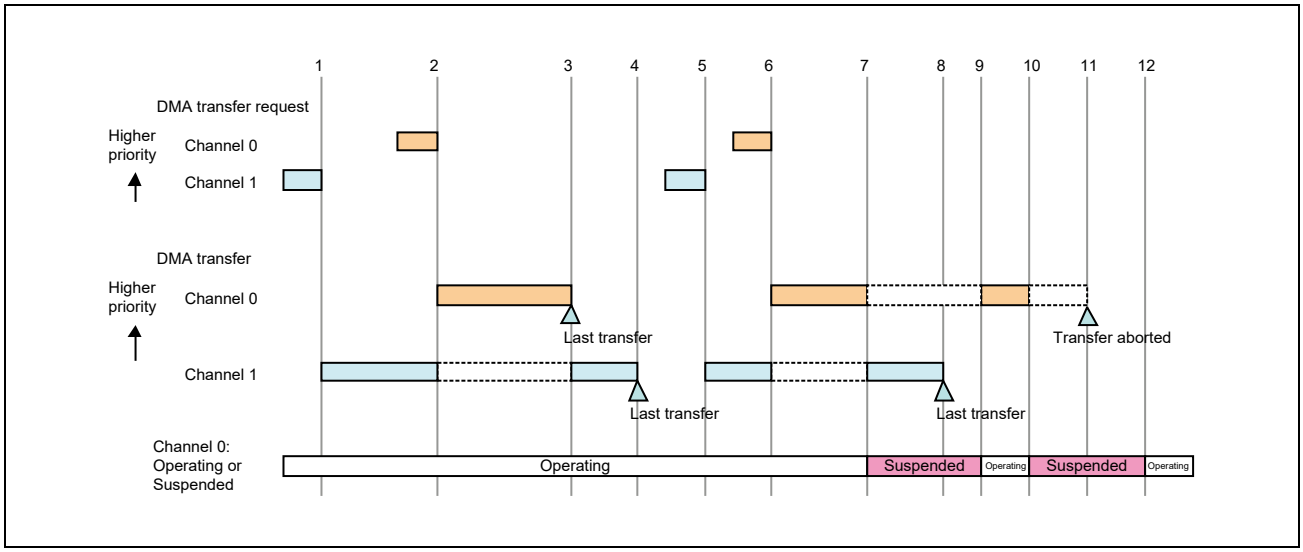


Figure 7.10 Example of Suspension, Resume, and Transfer Abort of a DMAC Channel

7.3.3 Suspension, Resume, and Transfer Abort of a DTS

You can suspend the DMA transfer executed by a DTS by setting the DTS suspend bit (DTSCTL1.DTSUST) in the DTS control register 1 or by setting the DMA suspension bit (DMACTL.DMASPD) in the DMA control register*[1]. If a DMA cycle is ongoing, the DMA transfer is suspended at the timing when the DMA cycle is finished. If the ongoing DMA cycle is a single transfer or a transfer that completes a block transfer (the last transfer or address reload transfer), the DMA transfer is suspended after a TI write back after the completion of the DMA cycle. If the ongoing DMA cycle is a type other than the above, the DMA transfer is suspended after the completion of the DMA cycle without a TI write back. If you resume the DMA transfer while the DMA transfer is suspended, clear the DTS suspend bit in the DTS control register 1, or clear the DMA suspension bit in the DMA control register*[1].

Note 1. Before suspending or resuming the DMA transfer being executed by a DTS, it is necessary to perform enabling or disabling operation by using the DMA transfer request enable bit (DTFSLnnn.REQEN). To suspend or resume it, perform the following operations.

(a) Procedure for suspending DTSs

1. Clear all the DMA transfer request enable bits (DTFSLnnn.REQEN) that are set for their respective DTS channels.
2. Set the DTSCTL1.DTSUST bit or the DMACTL.DMASPD bit.

(b) Procedure for resuming DTSs

1. Set the DMA transfer request enable bits (DTFSLnnn.REQEN) for DTS channels that are cleared in 1 of (a).
2. Clear the DTSCTL1.DTSUST bit or the DMACTL.DMASPD bit, which is set in 2 of (a).

If you want to abort the DMA transfer being executed by a DTS, suspend the DTS using the DTS suspend bit (DTSCTL1.DTSUST) in the DTS control register 1 as described above, and then set the DTS transfer abort request bit (DTSCTL2.DTSTIT) in the DTS control register 2 to abort the currently suspended DMA transfer. If transfer is aborted, no TI write back is executed. In addition, aborting the DMA transfer does not change the value of the DTS suspend bit (DTSCTL1.DTSUST). If you want the DTS to accept another DMA transfer request after the abort, clear the DTS suspend bit after setting the DMA transfer request enable bit (DTFSLnnn.REQEN) of the DTS channel.

Figure 7.11 shows an example of suspension, resume, and transfer abort of a DTS.

In **Figure 7.11**, channels 0, 1, and 2 are executing block transfer. At time tick 1, a DMA transfer request for channel 1 is accepted and DMA transfer starts. At time tick 2, DMA transfer requests for channels 0 and 2 are generated. At time tick 3, the last transfer of channel 1 is complete, and as a result of DTS channel arbitration, a DMA transfer request for channel 0 is accepted, and DMA transfer of channel 0 starts because channel 0 has a higher priority than channel 2. At time tick 4, the last transfer of channel 0 is complete, and DMA transfer of channel 2 starts. At time tick 5, the DTS is put into the suspended state, and the DMA transfer of channel 2 is suspended. At time tick 6, DMA transfer requests for channels 0 and 1 are generated. At time tick 7, the suspended state for the DTS is cleared, and the DMA transfer of channel 2, which has been suspended in the middle of a block transfer, is resumed. If DMA transfer is suspended in the middle of a block transfer, no DTS channel arbitration is done when it is resumed. At time tick 8, the last transfer of channel 2 is complete, and as a result of DTS channel arbitration, a DMA transfer request for channel 0 is accepted and the DMA transfer starts because channel 0 has a higher priority than channel 1. At time tick 9, the DTS is put into the suspended state, and at time tick 10, the suspended DMA transfer of channel is aborted. When the suspended state of the DTS 0 is cleared at time tick 11, DMA transfer of channel 1 starts because there is no currently ongoing DMA transfer and channel 1 is the only channel with a DMA transfer request.

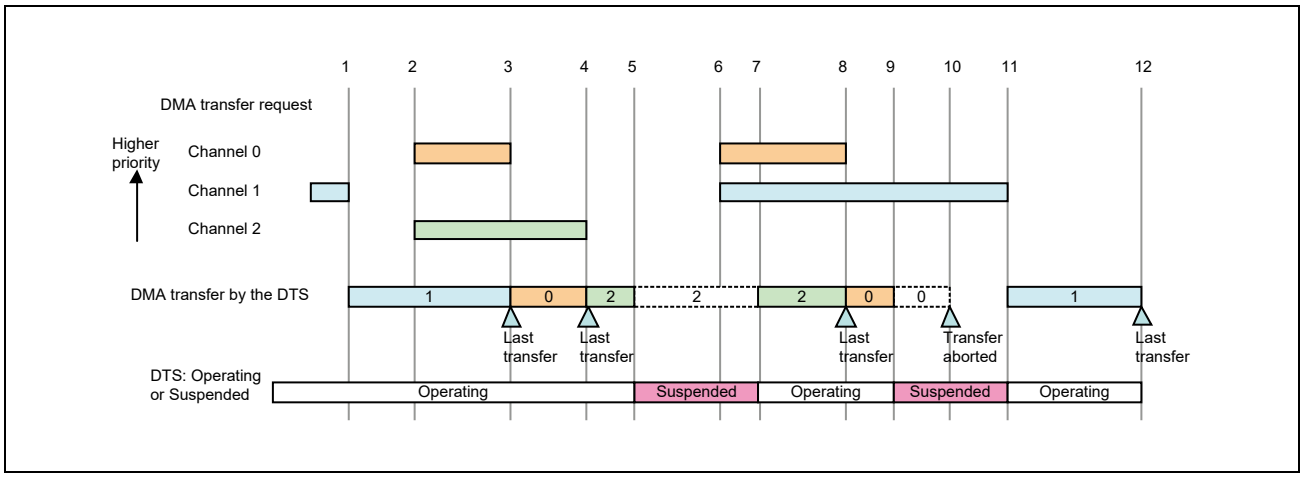


Figure 7.11 Example of Suspension, Resume, and Transfer Abort of a DTS

7.3.4 Masking and Clearing a Hardware DMA Transfer Request by the DTFR

If a DMAC uses a hardware DMA transfer request, you can temporarily disable (mask) the hardware DMA transfer request output from the DTFR to the DMAC by clearing the hardware DMA transfer source selection enable bit (DTFRn.REQEN) in the DTFR setting register.

Also, if a hardware DMA transfer source is used, you can clear a hardware DMA transfer request retained in the DTFR by using the hardware DMA transfer request clear (DTFRRQCn.DRQC) bit in the DTFR transfer request clear register.

Even if you suspend or abort DMA transfer of a DMAC channel, the hardware DMA transfer request selection/hold circuit of the DTFR is still running, and consequently, the DTFR may retain a hardware DMA transfer request that came to the DTFR during the suspension or transfer abort period of the DMAC channel. When you resume or start DMA transfer of a DMAC channel, clear the hardware DMA transfer request retained in the DTFR as required.

When DMAC uses a hardware transfer request and the block transfer 1 or block transfer 2, if software disables the hardware transfer source selection bit of DTFR (DTFRn.REQEN = 0) while DMAC is executing block transfer, the block transfer in progress is suspended.

7.3.5 Masking and Clearing a Hardware DMA Transfer Request by the DTSFSL

As for a DTS, you can temporarily disable (mask) a DMA transfer request from a channel to the DTS by clearing the transfer request enable bit (DTFSLnnn.REQEN) in the DTSFSL operation setting register. (The masking is actually done by excluding the channel from the candidates in DTS channel arbitration in the DTSFSL.)

Also, you can clear a DMA transfer request retained in the DTSFSL by using the transfer request clear (DTFSCnnn.DRQC) bit in the DTSFSL transfer request clear register.

Regardless of the state of the DTS and the transfer request enable bit (DTFSLnnn.REQEN) of the DTSFSL, the DTSFSL always monitors the hardware transfer source input, and a DMA transfer request for a channel is set when a hardware transfer source for the channel is input to the DTSFSL. When you resume or start DTS transfer, clear the hardware DMA transfer request retained in the DTSFSL as required.

7.3.6 List of Suspend, Resume, and Transfer Abort Functions

Table 7.5 List of Suspend, Resume, and Transfer Abort Functions

Function	How to execute the function	Operation	Possibility of DMA transfer abort	Master that can execute the function (See **Section 7.5, Reliability Function**)
DMA suspension and resume by software control	Setting and clearing the DMACTL.DMASPD*1	All channels are in the suspended state.	Not possible*2	Special master
Suspension and resume of a DMAC channel	Clearing and setting the DCENn.DTE in each channel register.	DMA transfer of a channel is suspended.	Possible (by clearing the DMA transfer request flag during suspension)	Special master, and general master assigned to the channel.
Suspension and resume of a DTS	Setting and clearing the DTSCTL1.DTSUST*1	DMA transfer of a DTS is suspended.	Possible (by setting the DTSCTL2.DTSTIT during suspension)	Special master

Note 1. In order to suspend or resume the DMA transfer being executed by a DTS, it is necessary to perform enabling or disabling operation by using the DMA transfer request enable bit (DTFSLnnn.REQEN). For details, refer to **Section 7.3.3, Suspension, Resume, and Transfer Abort of a DTS**.

Note 2. In order to abort DMA transfer, you need to either abort transfer for the DMAC channel or abort transfer for the DTS.

7.4 Error Control

7.4.1 Type of Error

DMA can generate the following two types of errors.

- DMA Transfer Error
  This error is generated when error is detected in the read cycle or write cycle in a DMA cycle. This error can be generated in all DMAC and DTS channels during execution of DMA transfer.
- DTSRAM Error
  This error is generated when ECC error is detected in the DTSRAM read access by a DTS. This error can be generated in the TI fetch during execution of DMA transfer for a DTS or while software is accessing the DTS channel registers.

7.4.2 DMA Transfer Error

7.4.2.1 Operation of a DMAC When DMA Transfer Error Occurs

When DMA transfer error occurs in a DMAC, the transfer error flag (DCSTn.ER) in the DMAC transfer status register of the channel with the DMA transfer error. The DMAC error register (DMACER) shows the transfer error flags of all 16 DMAC channels.

While the transfer error flag of a channel is set, a new DMA cycle is not executed if the transfer error case DMA transfer disable setting (DTCTn.ESE) bit is set. On the other hand, a DMA cycle is executed regardless of the value of the transfer error flag if the transfer error case DMA transfer disable setting (DTCTn.ESE) bit is cleared.

If you want to abort the DMA transfer of a channel with DMA transfer error, follow the procedure to abort DMA transfer of the DMAC channel.

If DMA transfer error occurs during the read cycle of a DMA cycle, the write cycle is not executed. If DMA transfer error occurs during the write cycle of a DMA cycle, the validity of the write is not guaranteed.

Regardless of whether DMA transfer error occurs in the read cycle or write cycle of a DMA cycle, the source address, destination address, transfer count, and address reload count registers are updated.

7.4.2.2 Operation of a DTS When DMA Transfer Error Occurs

When DMA transfer error occurs in a DTS, the DTS error flag (DTSER1.DTSER) in the DTS error register is set, and the DTS channel number with the DMA transfer error is stored in the DTS error channel (DTSER1.DTSERCH) in the same register.

If DMA transfer error occurs in a single transfer, a TI write back is executed to finish the DMA cycle.

If DMA transfer error occurs in the middle of a block transfer and the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is set, the remaining DMA cycles in the block transfer are not executed, but a TI write back is executed to finish the DMA cycle. At the same time, the DTS transfer status (DTSSTS.DTSACT) bit in the DTS status register is cleared. If DMA transfer error occurs in the middle of a block transfer and the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is cleared, the block transfer continues regardless of the DMA transfer error.

If DMA transfer error occurs during the read cycle of a DMA cycle, the write cycle is not executed. If DMA transfer error occurs during the write cycle of a DMA cycle, the validity of the write is not guaranteed.

Regardless of whether DMA transfer error occurs in the read cycle or write cycle of a DMA cycle, the source address, destination address, transfer count, and address reload count registers are updated, and the TI is updated by a TI write back.

If the DTS error flag in the DTS error register is set, a TI fetch is executed when the DTS accepts a DMA transfer request for the channel with the same channel number as the one stored in the DTS error channel. If, as a result of the TI fetch, the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is found to be set, a DMA cycle and a TI write back are not executed. If the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is cleared, DMA transfer is executed.

If the DTS error flag in the DTS error register is set, DMA transfer is executed when the DTS accepts a DMA transfer request for a channel with a channel number other than the one stored in the DTS error channel.

7.4.3 DTSRAM Error

There are two types of DTSRAM errors detected in the DTSRAM read access: 1-bit and 2-bit ECC error.

If a 1-bit ECC error is detected during a TI fetch, error corrected data is used, and DMA transfer continues. If a 1-bit ECC error is detected during DTS channel register access from software, error corrected data is returned as read data. In either case, the DTSRAM1 bit error flag (DTSER2.RAMSED) in the DTS error register 2 is set, and the address of the error location in the DTSRAM is stored to the DTSRAM1 bit error address (DTSER2.RAMSEDAD).

In addition, the error is notified to the ECM.

If a 2-bit ECC error is detected during a TI fetch, handling of the DMA transfer request is terminated without executing a DMA cycle and TI write back. If a 2-bit ECC error is detected during DTS channel register access from software, peripheral bus error is notified. In either case, the DTSRAM2 bit error flag (DTSER2.RAMDED) in the DTS error register 2 is set, and the address of the error location is stored to the DTSRAM2 bit error address (DTSER2.RAMDEDAD). In addition, the error is notified to the ECM.

7.5 Reliability Function

7.5.1 Overview

In this product, DMA is a resource used by multiple masters (CPU1, CPU2, or the internal EMU3 SubCPU). In order for DMA to support multi-core configuration, the following reliability functions are offered.

- Register access protection function
- Master information inherit function

7.5.2 Register Access Protection Function

This product is designed to assign each DMA channel to CPU1, CPU2, or the internal EMU3 SubCPU.

The register access protection function allows access to the transfer information of each DMA channel from the master (CPU1, CPU2, or the internal EMU3 SubCPU) assigned to the channel but prohibits access from other masters.

The register access protection function enables you, for example, to prevent the settings of the channel from being read or updated by masters other than the one assigned to the channel.

7.5.2.1 Identifying the Accessing Master

DMA identifies a master based on the ID of the accessing CPU (PEID), and whether the CPU is in supervisor mode (PSW.UM = 0) or user mode (PSW.UM = 1).

7.5.2.2 Master Access

There are the following two types of master accesses:

- Special master access (CPU1, CPU2, or the internal EMU3 SubCPU is in supervisor mode (UM = 0))
- General master access (access other than special master access)

In special master access, access to all registers is allowed.

In general master access, read access is allowed to all registers, but write access is allowed only to channel registers of channels assigned by means of channel assignment (See **Section 7.5.2.3, Channel Assignment**).

7.5.2.3 Channel Assignment

To each channel, DMA can assign a master (CPU1, CPU2, or the internal EMU3 SubCPU) so that the master is allowed to use the channel. Channel assignment is configured in the channel master setting register (DMnnCM in the case of a DMAC and DTSnnnCM in the case of a DTS) in supervisor mode by CPU1, CPU2, or the internal EMU3 SubCPU (UM = 0).

In general master access, the master assigned to a channel by channel assignment is allowed write access to the channel registers of the channel. If the channel registers of a channel is accessed for write by a master other than the master assigned to the channel, the access is called illegal access. For information about illegal access, see **Section 7.5.2.4, Illegal Access**.

7.5.2.4 Illegal Access

DMA handles the following access as illegal access.

(1) Write access from a general master to the global registers

(2) Write access to the channel registers of a channel from a general master other than the master assigned to the channel by means of channel assignment

In DMA, read access from any master is not treated as illegal access.

DMA's actions against illegal access are as follows.

For both (1) and (2),

- Write access is ignored.

Only for (2),

- The information about the illegal access is stored in a register access protection violation register.
- The DMAC0, DMAC1, and DTS have their own register access protection violation registers (DM0CMV, DM1CMV, and DTSCMV respectively).

Only the special master can access, for write, the register access protection violation registers. The special master can check whether illegal access has occurred by checking the register access protection violation registers periodically.

In addition, it is recommended that, when a master tries to use DMA and configures transfer information in the channel registers, the master should check whether the configuration has been successfully completed without illegal access by, for example, reading back the settings.

7.5.3 Master Information Inherit Function

In this product, DMA access inherits master information that is equivalent to the master information of the CPU1, CPU2, or the internal EMU3 SubCPU assigned to the DMA channel.

The master information that is output from DMA is as in **Table 7.6.**

Table 7.6 Master Information Output from DMA

Meaning	Value that is output from DMA
UM	Same as the UM bit value in the channel master setting register
SPID	Same as the SPID bit value in the channel master setting register
PEID	7
DMA access	1

7.5.4 Other Reliability Functions

7.5.4.1 Restriction on the Next Channel in the Chain

The reliability function limits the channels you can select as the next channel in the chain.

When you use the chain function, the channel master settings of a channel and its next channel in the chain must be the same.

The chain function is designed so that a channel and its next channel in the chain are managed by the same master.

When DMA detects that different masters are assigned to a channel and its next channel in the chain, it is deemed illegal and the chain function is suppressed. More specifically, when DMA tries to execute the chain function, DMA compares the chain master settings of the channel and its next channel in the chain, and if the settings are all the same for PEID and UM, the chain function is allowed and a chain request is sent to the next channel. If the settings are not the same for PEID or UM, a chain request is not sent.

7.6 Setting Up DMA Transfer

7.6.1 Overview of Setting Up DMA

Table 7.7 Channel Assignment (1/2)

No.	Master that configures the setting	Description	Register		Necessity of the setting
1	Special master (supervisor mode of CPU1, CPU2, or the internal EMU3 SubCPU (UM = 0))	Overall DMA operation setting	DTSPR0 to DTSPR7	DTS channel priority setting register	Mandatory (if a DTS is used)
2			DM00CM to DM17CM	DMAC channel master setting register	Mandatory (if a DMAC is used)
3			DTS0CM to DTS127CM	DTS channel master setting register	Mandatory (if a DTS is used)
4		Status clear	DTSERC	DTS error clear register	Recommended
5			CMVC	Channel protection violation clear register	Recommended
6	General master	Channel setting	DMATRGSELn	DMA trigger source select register	Mandatory
7	Master assigned to the DMAC channel		DSAn	DMAC source address register	Mandatory
8			DDAn	DMAC destination address register	Mandatory
9			DTCn	DMAC transfer count register	Mandatory
10			DTCTn	DMAC transfer control register	Mandatory
11			DRSAn	DMAC reload source address register	Mandatory if the reload function is used
12			DRDAn	DMAC reload destination address register	Mandatory if the reload function is used
13			DRTCn	DMAC reload transfer count register	Mandatory if the reload function is used
14			DTCCn	DMAC transfer count compare register	Mandatory if the transfer count match interrupt is used
15			DTFRn	DTFR setting register	Mandatory
16		Status clear	DCSTCn	DMAC transfer status clear register	Mandatory
17			DTFRRQCn	DTFR transfer request clear register	Recommended
18		Channel operation enable	DCENn	DMAC channel operation enable setting register	Mandatory

Table 7.7 Channel Assignment (2/2)

No.	Master that configures the setting	Description	Register		Necessity of the setting
19	General master	Channel setting	DTSTRGSELn	DTS trigger source select register	Mandatory
20	Master assigned to the DTS channel		DTSAnnn	DTS source address register	Mandatory
21			DTDAnnn	DTS destination address register	Mandatory
22			DTTCnnn	DTS transfer count register	Mandatory
23			DTTCTnnn	DTS transfer control register	Mandatory
24			DTRSAnnn	DTS reload source address register	Mandatory if the reload function is used
25			DTRDAnnn	DTS reload destination address register	Mandatory if the reload function is used
26			DTRTCnnn	DTS reload transfer count register	Mandatory if the reload function is used
27			DTTCCnnn	DTS transfer count compare register	Mandatory if the transfer count match interrupt is used
28		Status clear	DTFSCnnn	DTSFSL transfer request clear register	Recommended
29		Transfer request enable	DTFSLnnn	DTSFSL operation setting register	Mandatory

7.6.2 Setting Up the Overall DMA Operation

You need to set up the overall DMA operation before you start using DMA.

To configure the overall DMA operation, the special master (supervisor mode of CPU1, CPU2, or the internal EMU3 SubCPU (UM = 0)) needs to set up global registers. Global registers can be set up only by special master access. For details, see **Section 7.5, Reliability Function**.

The following registers must be set up to configure the overall DMA operation.

- DTS channel priority setting registers (DTSPRn, n = 0 to 7)
  Those registers configure the priority level of each DTS channel used for DTS channel arbitration.
- DMAC channel master setting registers (DMnnCM)
- DTS channel master setting registers (DTSnnnCM)
  Those registers configure channel assignment. (For details, see **Section 7.5, Reliability Function**.)

If the DMAC channel master setting registers and the DTS channel master setting registers are not properly set, DMA channel setting and DMA transfer cannot be executed properly.

Also, if errors are detected in the following registers while the overall DMA operation is set up, clearing the errors is recommended.

- DTS error register 1 (DTSER1)
- DTS error register 2 (DTSER2)
- DMAC0 register access protection violation register (DM0CMV)
- DMAC1 register access protection violation register (DM1CMV)
- DTS register access protection violation register (DTSCMV)

7.6.3 Setting Up the DMA Channel Setting

The DMA channel setting defines the transfer information and transfer source for each DMAC and DTS channel.

To configure the DMA channel setting, the master assigned to each channel by the channel assignment needs to set up channel registers.

7.6.3.1 Setting Up the DMAC Channel Setting

Follow the procedure below to set up the DMAC channel setting and use the DMAC.

(1) Disabling the DMAC Channel Operation

If the channel operation enable (DTE) in the DMAC channel operation enable setting register (DCENn) is set, clear the DTE bit to disable the channel operation.

(2) Setting Up the Transfer Information

When you set up the transfer information of the DMAC, the following registers need to be set up.

- DMAC source address register (DSAn)
- DMAC destination address register (DDAn)
- DMAC transfer count register (DTCn)
- DMAC transfer control register (DTCTn)
- DMAC reload source address register (DRSAn)
- DMAC reload destination address register (DRDAn)
- DMAC reload transfer count register (DRTCn)
- DMAC transfer count compare register (DTCCn)

(3) Setting Up the DMA Transfer Request

While setting the transfer information, you need to set up the DMA transfer request selection assignment (DTCTn.DRS) bit in the DMAC transfer control register (DTCTn) to define whether the hardware or software DMA transfer request is used.

You cannot use both the hardware and software DMA transfer requests for the same channel at the same time.

If you use the hardware DMA transfer request, you need to select from Primary channel/Secondary channel using the DMA trigger source selection register, and then to select one source used as the hardware DMA transfer request out of 128 hardware DMA transfer sources using the hardware DMA transfer source selection (DTFRn.REQSEL) in the DTFR setting register. Also, you need to enable the hardware DMA transfer source selection (DTFRn.REQEN) in the same register.

The DTFR may retain a hardware DMA transfer request that came before the hardware DMA transfer source is selected. Clear the hardware DMA transfer request (DTFRRQn.DRQ) retained in the DTFR using the DTFR transfer request clear register (DTFRRQCn) if necessary.

If you use the software DMA transfer request, disable the hardware DMA transfer source selection (DTFRn.REQEN) in the DTFR setting register.

(4) Clearing the Transfer Status

The DMAC transfer status register (DCSTn) may retain the result of the previous DMA transfer. You need to clear the flags in the DMAC transfer status register using the DMAC transfer status clear register (DCSTCn).

(5) Enabling the DMAC Channel Operation

Set the channel operation enable (DCENn.DTE) bit in the DMAC channel operation enable setting register to enable the channel operation.

After the channel operation enable bit is set, the DMAC can accept a DMA transfer request and start DMA transfer.

7.6.3.2 Setting Up the DTS Channel Setting

Follow the procedure below to set up the DTS channel setting and use the DTS.

(1) Disabling the Transfer Request by the DTSFSL

Clear the transfer request enable (DTFSLnnn.REQEN) bit in the DTSFSL operation setting register of the DTS channel you want to set up the channel setting for. This procedure is not mandatory but recommended in order to prevent a DMA transfer request from being sent mistakenly to the DTS channel currently being configured.

It is also recommended to check the DTS status register (DTSSTS) and confirm that DMA transfer is not ongoing for the DTS channel currently being configured.

(2) Setting Up the Transfer Information

When you set up the transfer information of the DTS, the following registers need to be set up to configure the transfer information.

- DTS source address register (DTSAnnn)
- DTS destination address register (DTDAnnn)
- DTS transfer count register (DTTCnnn)
- DTS transfer control register (DTTCTnnn)
- DTS reload source address register (DTRSAnnn)
- DTS reload destination address register (DTRDAnnn)
- DTS reload transfer count register (DTRTCnnn)
- DTS transfer count compare register (DTTCCnnn)

(3) Setting Up the DMA Transfer Request

Unlike a DMAC, a DTS does not care whether a DMA transfer request is a hardware DMA transfer request or software DMA transfer request. A DTS has a transfer request pending bit for each channel in the DTSFSL, and both a hardware and software DMA transfer requests are retained in the same transfer request pending bit (DTFSTnnn.DRQ). Therefore, a DTS has no setting for selecting whether the hardware or software transfer request is used.

The DTSFSL may retain a DMA transfer request that came before the transfer information is set up. Clear the DMA transfer request (DTFSTnnn.DRQ) retained in the DTSFSL if necessary, using the DTSFSL transfer request clear register (DTFSCnnn).

If you use the hardware DMA transfer request, you need to select from Primary channel/Secondary channel using the DTS trigger source register.

(4) Enabling the Transfer Request by the DTSFSL

Set the transfer request enable (DTFSLnnn.REQEN) bit in the DTSFSL operation setting register to enable the DMA transfer request for the DTS channel.

After the transfer request enable bit for the DTSFSL is set, the DTS can accept a DMA transfer request and start DMA transfer.

7.7 DMA Trigger Source

7.7.1 List of DMA Trigger Sources

The DMA trigger source assignment for DMA channel n is set in the DTFR setting register (DTFRn).

Table 7.8 List of DMA Trigger Sources (1/4)

CH	Primary Channel DMA Source Name	Function Module	C1M-A1	C1M-A2	Secondary Channel DMA Source Name	Function Module	C1M-A1	C1M-A2
DMACTRG0	ADI00 ADC0 scan group 0 end interrupt	ADCC	√	√	Reserved		—	—
DMACTRG1	ADI01 ADC0 scan group 1 end interrupt		√	√	Reserved		—	—
DMACTRG2	ADI02 ADC0 scan group 2 end interrupt		√	√	Reserved		—	—
DMACTRG3	ADI03 ADC0 scan group 3 end interrupt		√	√	Reserved		—	—
DMACTRG4	ADI04 ADC0 scan group 4 end interrupt		√	√	Reserved		—	—
DMACTRG5	ADI10 ADC1 scan group 0 end interrupt		√	√	Reserved		—	—
DMACTRG6	ADI11 ADC1 scan group 1 end interrupt		√	√	Reserved		—	—
DMACTRG7	ADI12 ADC1 scan group 2 end interrupt		√	√	Reserved		—	—
DMACTRG8	ADI13 ADC1 scan group 3 end interrupt		√	√	Reserved		—	—
DMACTRG9	ADI14 ADC1 scan group 4 end interrupt		√	√	Reserved		—	—
DMACTRG10	ADI20 ADC2 scan group 0 end interrupt		√	√	Reserved		—	—
DMACTRG11	ADI21 ADC2 scan group 1 end interrupt		√	√	Reserved		—	—
DMACTRG12	ADI22 ADC2 scan group 2 end interrupt		√	√	A signal for the DMA program command	Data Flash	√	√
DMACTRG13	ADI23 ADC2 scan group 3 end interrupt		√	√	Reserved		—	—
DMACTRG14	ADI24 ADC2 scan group 4 end interrupt		√	√	Reserved		—	—
DMACTRG15	CH0 INTTAUJ0I0 interrupt	TAUJ_0	√	√	CH0 INTTAUD2I0 interrupt	TAUD_2	×	√
DMACTRG16	CH1 INTTAUJ0I1 interrupt		√	√	CH1 INTTAUD2I1 interrupt	(CH0-3)	×	√
DMACTRG17	CH2 INTTAUJ0I2 interrupt		√	√	CH2 INTTAUD2I2 interrupt		×	√
DMACTRG18	CH3 INTTAUJ0I3 interrupt		√	√	CH3 INTTAUD2I3 interrupt		×	√
DMACTRG19	CH0 INTTAUD0I0 interrupt	TAUD_0	√	√	ADPA DMA request 0	ADPA	√	√
DMACTRG20	CH1 INTTAUD0I1 interrupt		√	√	ADPA DMA request 1		√	√
DMACTRG21	CH2 INTTAUD0I2 interrupt		√	√	ADPA DMA request 2		√	√
DMACTRG22	CH3 INTTAUD0I3 interrupt		√	√	ADPA DMA request 3		√	√
DMACTRG23	CH4 INTTAUD0I4 interrupt		√	√	ADPA DMA request 4		√	√
DMACTRG24	CH5 INTTAUD0I5 interrupt		√	√	ADPA DMA request 5		√	√
DMACTRG25	CH6 INTTAUD0I6 interrupt		√	√	CH6 INTTAUD2I6 interrupt	TAUD_2	×	√
DMACTRG26	CH7 INTTAUD0I7 interrupt		√	√	CH7 INTTAUD2I7 interrupt	(CH6-15)	×	√
DMACTRG27	CH8 INTTAUD0I8 interrupt		√	√	CH8 INTTAUD2I8 interrupt		×	√
DMACTRG28	CH9 INTTAUD0I9 interrupt		√	√	CH9 INTTAUD2I9 interrupt		×	√
DMACTRG29	CH10 INTTAUD0I10 interrupt		√	√	CH10 INTTAUD2I10 interrupt		×	√
DMACTRG30	CH11 INTTAUD0I11 interrupt		√	√	CH11 INTTAUD2I11 interrupt		×	√
DMACTRG31	CH12 INTTAUD0I12 interrupt		√	√	CH12 INTTAUD2I12 interrupt		×	√
DMACTRG32	CH13 INTTAUD0I13 interrupt		√	√	CH13 INTTAUD2I13 interrupt		×	√
DMACTRG33	CH14 INTTAUD0I14 interrupt		√	√	CH14 INTTAUD2I14 interrupt		×	√
DMACTRG34	CH15 INTTAUD0I15 interrupt		√	√	CH15 INTTAUD2I15 interrupt		×	√

Table 7.8 List of DMA Trigger Sources (2/4)

CH	Primary Channel DMA Source Name	Function Module	C1M-A1	C1M-A2	Secondary Channel DMA Source Name	Function Module	C1M-A1	C1M-A2
DMACTRG35	CH0 INTTAUD1I0 interrupt	TAUD_1	√	√	EMU30 interrupt 5	EMU3	√	√
DMACTRG36	CH1 INTTAUD1I1 interrupt		√	√	EMU30 interrupt 6		√	√
DMACTRG37	CH2 INTTAUD1I2 interrupt		√	√	EMU30 interrupt 7		√	√
DMACTRG38	CH3 INTTAUD1I3 interrupt		√	√	Reserved		—	—
DMACTRG39	CH4 INTTAUD1I4 interrupt		√	√	Reserved		—	—
DMACTRG40	CH5 INTTAUD1I5 interrupt		√	√	EMU31 interrupt 5	EMU3	√	√
DMACTRG41	CH6 INTTAUD1I6 interrupt		√	√	EMU31 interrupt 6		√	√
DMACTRG42	CH7 INTTAUD1I7 interrupt		√	√	EMU31 interrupt 7		√	√
DMACTRG43	CH8 INTTAUD1I8 interrupt		√	√	Reserved		—	—
DMACTRG44	CH9 INTTAUD1I9 interrupt		√	√	Reserved		—	—
DMACTRG45	CH10 INTTAUD1I10 interrupt		√	√	RSENT0 reception interrupt (INT_SENT_RX0)	RSENT_0	√	√
DMACTRG46	CH11 INTTAUD1I11 interrupt		√	√	RSENT1 reception interrupt (INT_SENT_RX1)	RSENT_1	√	√
DMACTRG47	CH12 INTTAUD1I12 interrupt		√	√	RSENT2 reception interrupt (INT_SENT_RX2)	RSENT_2	√	√
DMACTRG48	CH13 INTTAUD1I13 interrupt		√	√	RSENT3 reception interrupt (INT_SENT_RX3)	RSENT_3	√	√
DMACTRG49	CH14 INTTAUD1I14 interrupt		√	√	ADPA DMA request 6	ADPA	√	√
DMACTRG50	CH15 INTTAUD1I15 interrupt		√	√	ADPA DMA request 7		√	√
DMACTRG51	ENCA0 compare 0 match or Capture0 interrupt	PIC1B	√	√	CH4 INTTAUD2I4 interrupt	TAUD_2	×	√
DMACTRG52	ENCA1 compare 0 match or Capture 0 interrupt		√	√	CH5 INTTAUD2I5 interrupt	(ch4-5)	×	√
DMACTRG53	Overflow interrupt signal	ENCA_0	√	√	ADPA DMA request 8	ADPA	√	√
DMACTRG54	Compare 1 match or capture 1 interrupt		√	√	ADPA DMA request 9		√	√
DMACTRG55	Underflow interrupt		√	√	ADPA DMA request 10		√	√
DMACTRG56	Interrupt to indicate clearing due to clearing input from the encoder		√	√	ADPA DMA request 11		√	√
DMACTRG57	Overflow interrupt signal	ENCA_1	√	√	ADPA DMA request 12	ADPA	√	√
DMACTRG58	Compare 1 match or capture 1 interrupt		√	√	ADPA DMA request 13		√	√
DMACTRG59	Underflow interrupt		√	√	ADPA DMA request 14		√	√
DMACTRG60	Interrupt to indicate clearing due to clearing input from the encoder		√	√	ADPA DMA request 15		√	√
DMACTRG61	CH0 INTTAUD2I0 interrupt	TAUD_2	×	√	Cycle match detection interrupt	TPBA_0	√	√
DMACTRG62	CH1 INTTAUD2I1 interrupt	(CH0-5)	×	√	Duty match detection interrupt		√	√
DMACTRG63	CH2 INTTAUD2I2 interrupt		×	√	Pattern count match detection interrupt		√	√
DMACTRG64	CH3 INTTAUD2I3 interrupt		×	√	Cycle match detection interrupt	TPBA_1	×	√
DMACTRG65	CH4 INTTAUD2I4 interrupt		×	√	Duty match detection interrupt		×	√
DMACTRG66	CH5 INTTAUD2I5 interrupt		×	√	Pattern count match detection interrupt		×	√
DMACTRG67	RDC3A0 Z-phase interrupt	RDC_0	√	√	Reserved		—	—
DMACTRG68	RDC3A0 compare 0 match interrupt		√	√	ADPA DMA request 16	ADPA	√	√
DMACTRG69	RDC3A0 compare 1 match interrupt		√	√	ADPA DMA request 17		√	√
DMACTRG70	RDC3A0 compare 2 match interrupt		√	√	ADPA DMA request 18		√	√
DMACTRG71	RDC3A0 excitation timer (ET) DMA request		√	√	ADPA DMA request 19		√	√

Table 7.8 List of DMA Trigger Sources (3/4)

CH	Primary Channel DMA Source Name	Function Module	C1M-A1	C1M-A2	Secondary Channel DMA Source Name	Function Module	C1M-A1	C1M-A2
DMACTRG72	RDC3A1 Z-phase interrupt	RDC_1	×	√	Reserved		—	—
DMACTRG73	RDC3A1 compare 0 match interrupt		×	√	CH6 INTTAUD2I6 interrupt	TAUD_2	×	√
DMACTRG74	RDC3A1 compare 1 match interrupt		×	√	Reserved		—	—
DMACTRG75	RDC3A1 compare 2 match interrupt		×	√	Reserved		—	—
DMACTRG76	RDC3A1 excitation timer (ET) DMA request		×	√	Reserved		—	—
DMACTRG77	EMU30 interrupt 0	EMU3	√	√	Reserved		—	—
DMACTRG78	EMU30 interrupt 1		√	√	Reserved		—	—
DMACTRG79	EMU30 interrupt 2		√	√	Reserved		—	—
DMACTRG80	EMU30 interrupt 3		√	√	Reserved		—	—
DMACTRG81	EMU30 interrupt 4		√	√	Reserved		—	—
DMACTRG82	EMU31 interrupt 0		√	√	Reserved		—	—
DMACTRG83	EMU31 interrupt 1		√	√	Reserved		—	—
DMACTRG84	EMU31 interrupt 2		√	√	Reserved		—	—
DMACTRG85	EMU31 interrupt 3		√	√	Reserved		—	—
DMACTRG86	EMU31 interrupt 4		√	√	Reserved		—	—
DMACTRG87	TSG30CMP11E compare match interrupt	TSG3_0	√	√	Reserved		—	—
DMACTRG88	TSG30CMP12E compare match interrupt		√	√	Reserved		—	—
DMACTRG89	TSG30 peak interrupt (TSTIPEK)		√	√	TAPA0 peak interrupt 0	TAPA_0	√	√
DMACTRG90	TSG30 trough interrupt (TSTIVLY)		√	√	TAPA0 trough interrupt 0		√	√
DMACTRG91	TSG31CMP11E compare match interrupt	TSG3_1	√	√	Reserved		—	—
DMACTRG92	TSG31CMP12E compare match interrupt		√	√	Reserved		—	—
DMACTRG93	TSG31 peak interrupt (TSTIPEK)		√	√	TAPA1 peak interrupt 0	TAPA_1	√	√
DMACTRG94	TSG31 trough interrupt (TSTIVLY)		√	√	TAPA1 trough interrupt 0		√	√
DMACTRG95	TSG32CMP11E compare match interrupt	TSG3_2	×	√	Reserved		—	-
DMACTRG96	TSG32CMP12E compare match interrupt		×	√	Reserved		—	-
DMACTRG97	TSG32 peak interrupt (TSTIPEK)		×	√	TAPA2 peak interrupt 0	TAPA_2	×	√
DMACTRG98	TSG32 trough interrupt (TSTIVLY)		×	√	TAPA2 trough interrupt 0		×	√
DMACTRG99	CAN0 COM FIFO DMA request	RS-CANFD	√	√	Reserved		—	-
DMACTRG100	CAN1 COM FIFO DMA request		√	√	Reserved		—	-
DMACTRG101	CAN2 COM FIFO DMA request		√	√	Reserved		—	-
DMACTRG102	CAN3 COM FIFO DMA request		√	√	Reserved		—	-
DMACTRG103	CAN RX FIFO DMA REQUEST #0		√	√	ADPA DMA request 20	ADPA	√	√
DMACTRG104	CAN RX FIFO DMA REQUEST #1		√	√	ADPA DMA request 21		√	√
DMACTRG105	CAN RX FIFO DMA REQUEST #2		√	√	ADPA DMA request 22		√	√
DMACTRG106	CAN RX FIFO DMA REQUEST #3		√	√	ADPA DMA request 23		√	√
DMACTRG107	CAN RX FIFO DMA REQUEST #4		√	√	Reserved		—	-
DMACTRG108	CAN RX FIFO DMA REQUEST #5		√	√	Reserved		—	-
DMACTRG109	CAN RX FIFO DMA REQUEST #6		√	√	Reserved		—	-
DMACTRG110	CAN RX FIFO DMA REQUEST #7		√	√	Reserved		—	-

Table 7.8 List of DMA Trigger Sources (4/4)

CH	Primary Channel DMA Source Name	Function Module	C1M-A1	C1M-A2	Secondary Channel DMA Source Name	Function Module	C1M-A1	C1M-A2
DMACTRG111	Communication status interrupt	CSIH_0	√	√	CH7 INTTAUD2I7 interrupt	TAUD_2	×	√
DMACTRG112	Reception status interrupt		√	√	CH8 INTTAUD2I8 interrupt	(ch7—15)	×	√
DMACTRG113	Job completion interrupt		√	√	CH9 INTTAUD2I9 interrupt		×	√
DMACTRG114	Communication status interrupt	CSIH_1	√	√	CH10 INTTAUD2I10 interrupt		×	√
DMACTRG115	Reception status interrupt		√	√	CH11 INTTAUD2I11 interrupt		×	√
DMACTRG116	Job completion interrupt		√	√	CH12 INTTAUD2I12 interrupt		×	√
DMACTRG117	Communication status interrupt	CSIH_2	√	√	CH13 INTTAUD2I13 interrupt		×	√
DMACTRG118	Reception status interrupt		√	√	CH14 INTTAUD2I14 interrupt		×	√
DMACTRG119	Job completion interrupt		√	√	CH15 INTTAUD2I15 interrupt		×	√
DMACTRG120	RXI (reception data full)	SCI_0	√	√	INTRLIN30UR1 (reception interrupt)	RLIN3_0	√	√
DMACTRG121	TXI (transmission data empty)		√	√	INTRLIN30UR0 (transmission interrupt)		√	√
DMACTRG122	RXI (reception data full)	SCI_1	√	√	INTRLIN31UR1 (reception interrupt)	RLIN3_1	√	√
DMACTRG123	TXI (transmission data empty)		√	√	INTRLIN31UR0 (transmission interrupt)		√	√
DMACTRG124	RXI (reception data full)	SCI_2	√	√	INTRLIN32UR1 (reception interrupt)	RLIN3_2	√	√
DMACTRG125	TXI (transmission data empty)		√	√	INTRLIN32UR0 (transmission interrupt)		√	√
DMACTRG126	Word data write trigger					ICUSE	√	√
DMACTRG127	Word data read trigger/Command completion notification trigger						√	√

Note: The name of DMA trigger source may be different from that of output signal in each module.

7.8 DTS Trigger Source

7.8.1 List of DTS Trigger Sources

Table 7.9 shows the DTS trigger source assignment for DTS channel n.

Table 7.9 List of DTS Trigger Sources (1/4)

CH	Primary Channel DTS Source Name	Function Module	C1M-A1	C1M-A2	Secondary Channel DTS Source Name	Function Module	C1M-A1	C1M-A2
DTSTRG0	ADI00 ADC0 scan group 0 end interrupt	ADCC	√	√	Reserved		—	—
DTSTRG1	ADI01 ADC0 scan group 1 end interrupt		√	√	Reserved		—	—
DTSTRG2	ADI02 ADC0 scan group 2 end interrupt		√	√	Reserved		—	—
DTSTRG3	ADI03 ADC0 scan group 3 end interrupt		√	√	Reserved		—	—
DTSTRG4	ADI04 ADC0 scan group 4 end interrupt		√	√	Reserved		—	—
DTSTRG5	ADI10 ADC1 scan group 0 end interrupt		√	√	Reserved		—	—
DTSTRG6	ADI11 ADC1 scan group 1 end interrupt		√	√	Reserved		—	—
DTSTRG7	ADI12 ADC1 scan group 2 end interrupt		√	√	Reserved		—	—
DTSTRG8	ADI13 ADC1 scan group 3 end interrupt		√	√	Reserved		—	—
DTSTRG9	ADI14 ADC1 scan group 4 end interrupt		√	√	Reserved		—	—
DTSTRG10	ADI20 ADC2 scan group 0 end interrupt		√	√	Reserved		—	—
DTSTRG11	ADI21 ADC2 scan group 1 end interrupt		√	√	Reserved		—	—
DTSTRG12	ADI22 ADC2 scan group 2 end interrupt		√	√	Reserved		—	—
DTSTRG13	ADI23 ADC2 scan group 3 end interrupt		√	√	Reserved		—	—
DTSTRG14	ADI24 ADC2 scan group 4 end interrupt		√	√	Reserved		—	—
DTSTRG15	CH0 INTTAUJ0I0 interrupt	TAUJ_0	√	√	CH0 INTTAUD2I0 interrupt	TAUD_2	×	√
DTSTRG16	CH1 INTTAUJ0I1 interrupt		√	√	CH1 INTTAUD2I1 interrupt	(CH0-3)	×	√
DTSTRG17	CH2 INTTAUJ0I2 interrupt		√	√	CH2 INTTAUD2I2 interrupt		×	√
DTSTRG18	CH3 INTTAUJ0I3 interrupt		√	√	CH3 INTTAUD2I3 interrupt		×	√
DTSTRG19	CH0 INTTAUD0I0 interrupt	TAUD_0	√	√	ADPA DMA request 0	ADPA	√	√
DTSTRG20	CH1 INTTAUD0I1 interrupt		√	√	ADPA DMA request 1		√	√
DTSTRG21	CH2 INTTAUD0I2 interrupt		√	√	ADPA DMA request 2		√	√
DTSTRG22	CH3 INTTAUD0I3 interrupt		√	√	ADPA DMA request 3		√	√
DTSTRG23	CH4 INTTAUD0I4 interrupt		√	√	ADPA DMA request 4		√	√
DTSTRG24	CH5 INTTAUD0I5 interrupt		√	√	ADPA DMA request 5		√	√
DTSTRG25	CH6 INTTAUD0I6 interrupt		√	√	CH6 INTTAUD2I6 interrupt	TAUD_2	×	√
DTSTRG26	CH7 INTTAUD0I7 interrupt		√	√	CH7 INTTAUD2I7 interrupt	(CH6-15)	×	√
DTSTRG27	CH8 INTTAUD0I8 interrupt		√	√	CH8 INTTAUD2I8 interrupt		×	√
DTSTRG28	CH9 INTTAUD0I9 interrupt		√	√	CH9 INTTAUD2I9 interrupt		×	√
DTSTRG29	CH10 INTTAUD0I10 interrupt		√	√	CH10 INTTAUD2I10 interrupt		×	√
DTSTRG30	CH11 INTTAUD0I11 interrupt		√	√	CH11 INTTAUD2I11 interrupt		×	√
DTSTRG31	CH12 INTTAUD0I12 interrupt		√	√	CH12 INTTAUD2I12 interrupt		×	√
DTSTRG32	CH13 INTTAUD0I13 interrupt		√	√	CH13 INTTAUD2I13 interrupt		×	√
DTSTRG33	CH14 INTTAUD0I14 interrupt		√	√	CH14 INTTAUD2I14 interrupt		×	√
DTSTRG34	CH15 INTTAUD0I15 interrupt		√	√	CH15 INTTAUD2I15 interrupt		×	√

Table 7.9 List of DTS Trigger Sources (2/4)

CH	Primary Channel DTS Source Name	Function Module	C1M-A1	C1M-A2	Secondary Channel DTS Source Name	Function Module	C1M-A1	C1M-A2
DTSTRG35	CH0 INTTAUD1I0 interrupt	TAUD_1	√	√	EMU30 interrupt 5	EMU3	√	√
DTSTRG36	CH1 INTTAUD1I1 interrupt		√	√	EMU30 interrupt 6		√	√
DTSTRG37	CH2 INTTAUD1I2 interrupt		√	√	EMU30 interrupt 7		√	√
DTSTRG38	CH3 INTTAUD1I3 interrupt		√	√	Reserved		—	—
DTSTRG39	CH4 INTTAUD1I4 interrupt		√	√	Reserved		—	—
DTSTRG40	CH5 INTTAUD1I5 interrupt		√	√	EMU31 interrupt 5	EMU3	√	√
DTSTRG41	CH6 INTTAUD1I6 interrupt		√	√	EMU31 interrupt 6		√	√
DTSTRG42	CH7 INTTAUD1I7 interrupt		√	√	EMU31 interrupt 7		√	√
DTSTRG43	CH8 INTTAUD1I8 interrupt		√	√	Reserved		—	—
DTSTRG44	CH9 INTTAUD1I9 interrupt		√	√	Reserved		—	—
DTSTRG45	CH10 INTTAUD1I10 interrupt		√	√	RSENT0 reception interrupt (INT_SENT_RX0)	RSENT_0	√	√
DTSTRG46	CH11 INTTAUD1I11 interrupt		√	√	RSENT1 reception interrupt (INT_SENT_RX1)	RSENT_1	√	√
DTSTRG47	CH12 INTTAUD1I12 interrupt		√	√	RSENT2 reception interrupt (INT_SENT_RX2)	RSENT_2	√	√
DTSTRG48	CH13 INTTAUD1I13 interrupt		√	√	RSENT3 reception interrupt (INT_SENT_RX3)	RSENT_3	√	√
DTSTRG49	CH14 INTTAUD1I14 interrupt		√	√	ADPA DMA request 6	ADPA	√	√
DTSTRG50	CH15 INTTAUD1I15 interrupt		√	√	ADPA DMA request 7		√	√
DTSTRG51	ENCA0 compare 0 match or Capture0 interrupt	PIC1B	√	√	CH4 INTTAUD2I4 interrupt	TAUD_2	×	√
DTSTRG52	ENCA1 compare 0 match or Capture0 interrupt		√	√	CH5 INTTAUD2I5 interrupt	(ch4-5)	×	√
DTSTRG53	Overflow interrupt signal	ENCA_0	√	√	ADPA DMA request 8	ADPA	√	√
DTSTRG54	Compare 1 match or capture 1 interrupt		√	√	ADPA DMA request 9		√	√
DTSTRG55	Underflow interrupt		√	√	ADPA DMA request 10		√	√
DTSTRG56	Interrupt to indicate clearing due to clearing input from the encoder		√	√	ADPA DMA request 11		√	√
DTSTRG57	Overflow interrupt signal	ENCA_1	√	√	ADPA DMA request 12		√	√
DTSTRG58	Compare 1 match or capture 1 interrupt		√	√	ADPA DMA request 13		√	√
DTSTRG59	Underflow interrupt		√	√	ADPA DMA request 14		√	√
DTSTRG60	Interrupt to indicate clearing due to clearing input from the encoder		√	√	ADPA DMA request 15		√	√
DTSTRG61	CH0 INTTAUD2I0 interrupt	TAUD_2	×	√	Cycle match detection interrupt	TPBA_0	√	√
DTSTRG62	CH1 INTTAUD2I1 interrupt	(CH0-5)	×	√	Duty match detection interrupt		√	√
DTSTRG63	CH2 INTTAUD2I2 interrupt		×	√	Pattern count match detection interrupt		√	√
DTSTRG64	CH3 INTTAUD2I3 interrupt		×	√	Cycle match detection interrupt	TPBA_1	×	√
DTSTRG65	CH4 INTTAUD2I4 interrupt		×	√	Duty match detection interrupt		×	√
DTSTRG66	CH5 INTTAUD2I5 interrupt		×	√	Pattern count match detection interrupt		×	√
DTSTRG67	RDC3A0 Z-phase interrupt	RDC_0	√	√	Reserved		—	—
DTSTRG68	RDC3A0 compare 0 match interrupt		√	√	ADPA DMA request 16	ADPA	√	√
DTSTRG69	RDC3A0 compare 1 match interrupt		√	√	ADPA DMA request 17		√	√
DTSTRG70	RDC3A0 compare 2 match interrupt		√	√	ADPA DMA request 18		√	√
DTSTRG71	RDC3A0 excitation timer (ET) DMA request		√	√	ADPA DMA request 19		√	√

Table 7.9 List of DTS Trigger Sources (3/4)

CH	Primary Channel DTS Source Name	Function Module	C1M-A1	C1M-A2	Secondary Channel DTS Source Name	Function Module	C1M-A1	C1M-A2
DTSTRG72	RDC3A1 Z-phase interrupt	RDC_1	×	√	Reserved		—	—
DTSTRG73	RDC3A1 compare 0 match interrupt		×	√	CH6 INTTAUD2I6 interrupt	TAUD_2	×	√
DTSTRG74	RDC3A1 compare 1 match interrupt		×	√	Reserved		—	—
DTSTRG75	RDC3A1 compare 2 match interrupt		×	√	Reserved		—	—
DTSTRG76	RDC3A1 excitation timer (ET) DMA request		×	√	Reserved		—	—
DTSTRG77	EMU30 interrupt 0	EMU3	√	√	Reserved		—	—
DTSTRG78	EMU30 interrupt 1		√	√	Reserved		—	—
DTSTRG79	EMU30 interrupt 2		√	√	Reserved		—	—
DTSTRG80	EMU30 interrupt 3		√	√	Reserved		—	—
DTSTRG81	EMU30 interrupt 4		√	√	Reserved		—	—
DTSTRG82	EMU31 interrupt 0		√	√	Reserved		—	—
DTSTRG83	EMU31 interrupt 1		√	√	Reserved		—	—
DTSTRG84	EMU31 interrupt 2		√	√	Reserved		—	—
DTSTRG85	EMU31 interrupt 3		√	√	Reserved		—	—
DTSTRG86	EMU31 interrupt 4		√	√	Reserved		—	—
DTSTRG87	TSG30CMP11E compare match interrupt	TSG3_0	√	√	Reserved		—	—
DTSTRG88	TSG30CMP12E compare match interrupt		√	√	Reserved		—	—
DTSTRG89	TSG30 peak interrupt (TSTIPEK)		√	√	TAPA0 peak interrupt 0	TAPA_0	√	√
DTSTRG90	TSG30 trough interrupt (TSTIVLY)		√	√	TAPA0 trough interrupt 0		√	√
DTSTRG91	TSG31CMP11E compare match interrupt	TSG3_1	√	√	Reserved		—	—
DTSTRG92	TSG31CMP12E compare match interrupt		√	√	Reserved		—	—
DTSTRG93	TSG31 peak interrupt (TSTIPEK)		√	√	TAPA1 peak interrupt 0	TAPA_1	√	√
DTSTRG94	TSG31 trough interrupt (TSTIVLY)		√	√	TAPA1 trough interrupt 0		√	√
DTSTRG95	TSG32CMP11E compare match interrupt	TSG3_2	×	√	Reserved		—	—
DTSTRG96	TSG32CMP12E compare match interrupt		×	√	Reserved		—	—
DTSTRG97	TSG32 peak interrupt (TSTIPEK)		×	√	TAPA2 peak interrupt 0	TAPA_2	×	√
DTSTRG98	TSG32 trough interrupt (TSTIVLY)		×	√	TAPA2 trough interrupt 0		×	√
DTSTRG99	CAN0 COM FIFO DMA request	RS-CANFD	√	√	Reserved		—	—
DTSTRG100	CAN1 COM FIFO DMA request		√	√	Reserved		—	—
DTSTRG101	CAN2 COM FIFO DMA request		√	√	Reserved		—	—
DTSTRG102	CAN3 COM FIFO DMA request		√	√	Reserved		—	—
DTSTRG103	CAN RX FIFO DMA REQUEST #0		√	√	ADPA DMA request 20	ADPA	√	√
DTSTRG104	CAN RX FIFO DMA REQUEST #1		√	√	ADPA DMA request 21		√	√
DTSTRG105	CAN RX FIFO DMA REQUEST #2		√	√	ADPA DMA request 22		√	√
DTSTRG106	CAN RX FIFO DMA REQUEST #3		√	√	ADPA DMA request 23		√	√
DTSTRG107	CAN RX FIFO DMA REQUEST #4		√	√	Reserved		—	—
DTSTRG108	CAN RX FIFO DMA REQUEST #5		√	√	Reserved		—	—
DTSTRG109	CAN RX FIFO DMA REQUEST #6		√	√	Reserved		—	—
DTSTRG110	CAN RX FIFO DMA REQUEST #7		√	√	Reserved		—	—

Table 7.9 List of DTS Trigger Sources (4/4)

CH	Primary Channel DTS Source Name	Function Module	C1M -A1	C1M -A2	Secondary Channel DTS Source Name	Function Module	C1M -A1	C1M-A2
DTSTRG111	Communication status interrupt	CSIH_0	√	√	CH7 INTTAUD2I7 interrupt	TAUD_2	×	√
DTSTRG112	Reception status interrupt		√	√	CH8 INTTAUD2I8 interrupt	(ch7-15)	×	√
DTSTRG113	Job completion interrupt		√	√	CH9 INTTAUD2I9 interrupt		×	√
DTSTRG114	Communication status interrupt	CSIH_1	√	√	CH10 INTTAUD2I10 interrupt		×	√
DTSTRG115	Reception status interrupt		√	√	CH11 INTTAUD2I11 interrupt		×	√
DTSTRG116	Job completion interrupt		√	√	CH12 INTTAUD2I12 interrupt		×	√
DTSTRG117	Communication status interrupt	CSIH_2	√	√	CH13 INTTAUD2I13 interrupt		×	√
DTSTRG118	Reception status interrupt		√	√	CH14 INTTAUD2I14 interrupt		×	√
DTSTRG119	Job completion interrupt		√	√	CH15 INTTAUD2I15 interrupt		×	√
DTSTRG120	RXI(reception data full)	SCI_0	√	√	INTRLIN30UR1 (reception interrupt)	RLIN3_0	√	√
DTSTRG121	TXI(transmission data empty)		√	√	INTRLIN30UR0 (transmission interrupt)		√	√
DTSTRG122	RXI(reception data full)	SCI_1	√	√	INTRLIN31UR1 (reception interrupt)	RLIN3_1	√	√
DTSTRG123	TXI(transmission data empty)		√	√	INTRLIN31UR0 (transmission interrupt)		√	√
DTSTRG124	RXI(reception data full)	SCI_2	√	√	INTRLIN32UR1 (reception interrupt)	RLIN3_2	√	√
DTSTRG125	TXI(transmission data empty)		√	√	INTRLIN32UR0 (transmission interrupt)		√	√
DTSTRG126	Word data write trigger					ICUSE	√	√
DTSTRG127	Word data read trigger/Command completion notification trigger						√	√

7.9 Global Register

7.9.1 List of Global Register Addresses

Address = Base address "FFFF 8000_H" + Offset address

Table 7.10 List of Global Register Addresses (1/2)

Module Name	Offset Address	Register Symbol	Meaning	Accessed Permission		
				Special Master	General Master	At secure boot
DMASS	0000_H	DMACTL	DMA control register	√	*2	*3
DMASS	0010_H	DTSCTL1	DTS control register 1	√	*2	
DMASS	0014_H	DTSCTL2	DTS control register 2	√	*2	
DMASS	0018_H	DTSSTS	DTS status register	√	*2	
DMASS	0020_H	DMACER	DMAC error register	√	*2	
DMASS	0024_H	DTSER1	DTS error register 1	√	*2	
DMASS	0028_H	DTSER2	DTS error register 2	√	*2	
DMASS	$002C_H$	DTSERC	DTS error clear register	√	*2	
DMASS	0030_H	DM0CMV	DMAC0 register access protection violation register	√	*2	
DMASS	0034_H	DM1CMV	DMAC1 register access protection violation register	√	*2	
DMASS	0038_H	DTSCMV	DTS register access protection violation register	√	*2	
DMASS	$003C_H$	CMVC	Register access protection violation clear register	√	*2	*3
DMASS	0060_H	DTSPR0	DTS channel priority setting register 0	√	*2	
DMASS	0064_H	DTSPR1	DTS channel priority setting register 1	√	*2	
DMASS	0068_H	DTSPR2	DTS channel priority setting register 2	√	*2	
DMASS	$006C_H$	DTSPR3	DTS channel priority setting register 3	√	*2	
DMASS	0070_H	DTSPR4	DTS channel priority setting register 4	√	*2	
DMASS	0074_H	DTSPR5	DTS channel priority setting register 5	√	*2	
DMASS	0078_H	DTSPR6	DTS channel priority setting register 6	√	*2	
DMASS	$007C_H$	DTSPR7	DTS channel priority setting register 7	√	*2	
DMASS	0080_H	DTRECCTL	DTSRAM ECC control register	√	*2	
DMASS	0084_H	DTRERINT	DTSRAM Error notification control register	√	*2	
DMASS	0094_H	DTRTSCTL	DTSRAM test control register	√	*2	
DMASS	0098_H	DTRTWDAT	DTSRAM test write data register	√	*2	

Table 7.10 List of Global Register Addresses (2/2)

Module Name	Offset Address	Register Symbol	Meaning	Accessed Permission		
				Special Master	General Master	At secure boot
DMASS	$009C_H$	DTRTRDAT	DTSRAM test read data register	√	*2	
DMASS	0100_H	DM00CM	DMAC0 channel 0 channel master setting register	√	*2	*3
DMASS	0104_H	DM01CM	DMAC0 channel 1 channel master setting register	√	*2	*3
DMASS	0108_H	DM02CM	DMAC0 channel 2 channel master setting register	√	*2	*3
DMASS	$010C_H$	DM03CM	DMAC0 channel 3 channel master setting register	√	*2	*3
DMASS	0110_H	DM04CM	DMAC0 channel 4 channel master setting register	√	*2	*3
DMASS	0114_H	DM05CM	DMAC0 channel 5 channel master setting register	√	*2	*3
DMASS	0118_H	DM06CM	DMAC0 channel6 channel master setting register	√	*2	*3
DMASS	$011C_H$	DM07CM	DMAC0 channel 7 channel master setting register	√	*2	*3
DMASS	0120_H	DM10CM	DMAC1 channel 0 channel master setting register	√	*2	*3
DMASS	0124_H	DM11CM	DMAC1 channel 1 channel master setting register	√	*2	*3
DMASS	0128_H	DM12CM	DMAC1 channel 2 channel master setting register	√	*2	*3
DMASS	$012C_H$	DM13CM	DMAC1 channel 3 channel master setting register	√	*2	*3
DMASS	0130_H	DM14CM	DMAC1 channel 4 channel master setting register	√	*2	*3
DMASS	0134_H	DM15CM	DMAC1 channel 5 channel master setting register	√	*2	*3
DMASS	0138_H	DM16CM	DMAC1 channel 6 channel master setting register	√	*2	*3
DMASS	$013C_H$	DM17CM	DMAC1 channel 7 channel master setting register	√	*2	*3
DMASS	0200_H + 4 × [DTS channel number]*1 (0200_H to $03FC_H$)	DTSnnnCM*1	DTS channel nnn channel master setting register*1	√	*2	

Note 1. [DTS channel number] and "nnn" in the register symbols and meanings are numbers in the range from 000 to 127

Note 2. Only read access is allowed.

Note 3. At secure boot, write access is masked. For details on secure boot, refer to **Section 32, Intelligent Cryptographic Unit E (ICUSE)**.

7.9.2 Details of Global Registers

7.9.2.1 DMACTL — DMA Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8000$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMASPD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.11 DMACTL Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DMASPD	DMA suspension This bit shows whether DMA transfer for all channels is suspended. If a user writes 1 to this bit, DMA transfer for all channels can be suspended. If a user writes 0 to this bit, suspension of DMA transfer for all channels can be cleared. The suspension controlled by this bit is independent from the suspension controlled by the transfer enable bit (DTE) of each DMAC channel and the suspension setting bit (DTSUST) for a DTS. That means, if this bit is 1, all DMA transfers are suspended regardless of the values of the DTE bit of each channel and the DTSUST bit of the DTS. Writing to this bit does not affect the DTE bit of each channel and the DTSUST bit of the DTS. 0: DMA suspension cleared 1: DMA suspension request/DMA suspension ongoing

CAUTION

In order to suspend or resume the DMA transfer being executed by a DTS, it is necessary to perform enabling or disabling operation by using the DMA transfer request enable bit (DTFSLnnn.REQEN). For details, refer to notes in **Section 7.3.3, Suspension, Resume, and Transfer Abort of a DTS**.

7.9.2.2 DTSCTL1 — DTS Control Register 1

Access: This register can be read/written in 32-bit units.

Address: FFFF 8010$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSUST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.12 DTSCTL1 Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DTSUST	DTS suspension This bit shows whether DMA transfer of a DTS is suspended. If a user writes 1 to this bit, DMA transfer of a DTS can be suspended. 0: DTS suspension cleared 1: DTS suspension request/DTS suspension ongoing

CAUTION

In order to suspend or resume the DMA transfer being executed by a DTS, it is necessary to perform enabling or disabling operation by using the DMA transfer request enable bit (DTFSLnnn.REQEN). For details, refer to notes in **Section 7.3.3, Suspension, Resume, and Transfer Abort of a DTS**.

7.9.2.3 DTSCTL2 — DTS Control Register 2

Access: This register can be read/written in 32-bit units.

Address: FFFF 8014$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSTIT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.13 DTSCTL2 Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DTSTIT	DTS transfer abort request While the DTS is suspended, a user can write 1 to this bit to abort the suspended DMA transfer. When the suspended DMA transfer of a DTS is aborted, the DTSSTS.DTSACT bit is cleared to 0. This bit is always read as 0.

7.9.2.4 DTSSTS — DTS Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 8018$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DTSCYC	DTSACH[6:0]							DTSACT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.14 DTSSTS Register Contents

Bit Position	Bit Name	Function
31 to 9	Reserved	When read, the value after reset is returned.
8	DTSCYC	DMA cycle execution state This bit shows whether a DMA cycle is ongoing in the DTS. 0: DMA cycle is not ongoing. 1: DMA cycle is ongoing.
7 to 1	DTSACH[6:0]	DTS transfer channel If there is a channel in the DTS executing DMA transfer, the channel number is shown. If there is no channel in the DTS executing DMA transfer, the channel number of the last DMA transfer is shown.
0	DTSACT	DTS transfer status This bit shows whether there is a channel in the DTS executing DMA transfer. 0: There is a channel in the DTS executing DMA transfer. 1: There is no channel in the DTS executing DMA transfer. If the DTS is put into the suspended state while there is a channel executing DMA transfer, this bit remains 1. If a DTS transfer abort request is made using the DTSCTL2.DTSTIT bit, the suspended DTS transfer is aborted, and this bit is cleared to 0. When DMA transfer error occurs and the DMA transfer is aborted, this bit is cleared.

7.9.2.5 DMACER — DMAC Error Register

Access: This register can be read in 32-bit units.

Address: FFFF 8020$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DM1ER[7:0]								DM0ER[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.15 DMACER Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned.
15 to 8	DM1ER[7:0]	DMAC1 DMA transfer error status These bits show the DMA transfer error status of channels 0 through 7 of the DMAC1. Each bit is mapped from the DCSTn.ER bit of each channel of the DMAC1 and is read-only. 0: DMA transfer error is not generated 1: DMA transfer error is generated
7 to 0	DM0ER[7:0]	DMAC0 DMA transfer error status These bits show the DMA transfer error status of channels 0 through 7 of the DMAC0. Each bit is mapped from the DCSTn.ER bit of each channel of the DMAC0 and is read-only. 0: DMA transfer error is not generated 1: DMA transfer error is generated

7.9.2.6 DTSER1 — DTS Error Register 1

Access: This register can be read in 32-bit units.

Address: FFFF 8024$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DTSERCH[6:0]							—	—	—	—	—	—	DTSERWR	DTSER
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.16 DTSER1 Register Contents

Bit Position	Bit Name	Function
31 to 15	Reserved	When read, the value after reset is returned.
14 to 8	DTSERCH[6:0]	DTS error channel These bits show the DTS channel number of the first DMA transfer error after the DTSER bit is cleared to 0. These bits are read-only and cannot be cleared.
7 to 2	Reserved	When read, the value after reset is returned.
1	DTSERWR	DTS DMA transfer error occurring cycle This bit is updated at the same time as setting of the DTS DMA transfer error flag (DTSER), indicating which cycle of read or write the DMA transfer error occurs in. This bit is not updated when a new DMA transfer error occurs after the DTSER bit has been set. If the DTSER bit is cleared, this bit is also cleared to 0. 0: DMA transfer error occurs in read cycle. 1: DMA transfer error occurs in write cycle.
0	DTSER	DTS DMA transfer error flag This bit shows whether DMA transfer error is generated in the DTS. 0: DMA transfer error is not generated 1: DMA transfer error is generated If DMA transfer error is generated in the DTS while this bit is 0, this bit is set, and DTSERCH6 to DTSERCH0 retain the DTS channel number of the DMA transfer error. If DMA transfer error is generated in the DTS while this bit is 1, this bit remains 1, and TSERCH6 to DTSERCH0 do not change. This bit can be cleared by using the DTSERC register.

7.9.2.7 DTSER2 — DTS Error Register 2

Access: This register can be read in 32-bit units.

Address: FFFF 8028$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RAMDED	RAMDEDOV	—	—	RAMDEDAD[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RAMSED	RAMSEDOV	—	—	RAMSEDAD[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.17 DTSER2 Register Contents (1/2)

Bit Position	Bit Name	Function
31	RAMDED	DTSRAM 2-bit error flag This bit shows whether the 2-bit error is generated in the read access to the DTSRAM. 0: 2-bit error is not generated in the DTSRAM 1: 2-bit error is generated in the DTSRAM If the 2-bit error occurs in DTSRAM read access while this bit is 0, this bit is set, and RAMDEDAD11 to RAMDEDAD0 retain the DTSRAM address of the error. If the 2-bit error occurs in DTSRAM read access while this bit is 1, this bit remains 1, and RAMDEDAD11 to RAMDEDAD0 do not change. This bit can be cleared by using the DTSERC register
30	RAMDEDOV	DTSRAM 2-bit error overflow flag This bit is set when the RAMDED bit is 1 and the 2-bit error occurs in DTSRAM read access whose address is different from that specified by the RAMDEDAD11 to RAMDEDAD0 bits. This bit can be cleared by operation of the DTSERC register.
29, 28	Reserved	When read, the value after reset is returned.
27 to 16	RAMDEDAD[11:0]	DTSRAM 2-bit error address These bits show the DTSRAM address at which the first 2-bit error occurs in DTSRAM read access after the RAMDED bit is cleared to 0. These bits are read-only and cannot be cleared.
15	RAMSED	DTSRAM 1-bit error flag This bit shows whether the 1-bit error is generated in the read access to the DTSRAM. 0: 1-bit error is not generated in the DTSRAM 1: 1-bit error is generated in the DTSRAM If the 1-bit error occurs in the DTSRAM while this bit is 0, this bit is set, and RAMSEDAD11 to RAMSEDAD0 retain the DTSRAM address of the error. If the 1-bit error occurs in DTSRAM read access while this bit is 1, this bit remains 1, and RAMSEDAD11 to RAMSEDAD0 do not change. This bit can be cleared by using the DTSERC register.

Table 7.17 DTSER2 Register Contents (2/2)

Bit Position	Bit Name	Function
14	RAMSEDOV	DTSRAM 1-bit error overflow flag This bit is set when the RAMSED bit is 1 and the 1-bit error occurs in DTSRAM read access whose address is different from that specified by the RAMSEDAD11 to RAMSEDAD0 bits. This bit can be cleared by operation of the DTSERC register.
13, 12	Reserved	When read, the value after reset is returned.
11 to 0	RAMSEDAD[11:0]	DTSRAM 1-bit error address These bits show the DTSRAM address of the first DTSRAM 1-bit error after the RAMSED bit is cleared to 0. These bits are read-only and cannot be cleared.

7.9.2.8 DTSERC — DTS Error Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 802C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RAMDEDC	RAMDEDOVC	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RAMSEDC	RAMSEDOVC	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSERC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.18 DTSERC Register Contents

Bit Position	Bit Name	Function
31	RAMDEDC	DTSRAM 2-bit error flag clear If a user writes 1 to this bit, the DTSRAM 2-bit error flag (DTSER2.RAMDED) is cleared. 0 is always read from this bit.
30	RAMDEDOVC	DTSRAM 2-bit error overflow flag clear When the user writes 1 to this bit, the DTSRAM 2-bit error overflow flag (DTSER2.RAMDEDOV) is cleared. The read value of this bit is always 0.
29 to 16	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
15	RAMSEDC	DTSRAM 1-bit error flag clear If a user writes 1 to this bit, the DTSRAM 1-bit error flag (DTSER2.RAMSED) is cleared. 0 is always read from this bit.
14	RAMSEDOVC	DTSRAM 1-bit error overflow flag clear When the user writes 1 to this bit, the DTSRAM 1-bit error overflow flag (DTSER2.RAMSEDOV) is cleared. The read value of this bit is always 0.
13 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DTSERC	DTS error flag clear If a user writes 1 to this bit, the DTS DMA error flag (DTSER1.DTSER) is cleared. 0 is always read from this bit.

7.9.2.9 DM0CMV — DMAC0 Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8030$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	VCH[2:0]			—	—	—	VF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.19 DM0CMV Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
22 to 17	MINF[6:1]	Illegal access master information These bits show the master information of the originator of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 7	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
6 to 4	VCH[2:0]	Illegal access channel These bits show the channel number (0 to 7) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DMAC0. 0: No illegal access has occurred in the DMAC0 1: Illegal access has occurred in the DMAC0 If an illegal access occurs in the DMAC0 while this bit is 0, this bit is set, and MINF6 to MINF1 and VCH2 to VCH0 store their respective information. If an illegal access occurs in the DMAC0 while this bit is 1, this bit remains 1, and MINF6 to MINF1 and VCH2 to VCH0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.10 DM1CMV — DMAC1 Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8034$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	VCH[2:0]			—	—	—	VF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.20 DM1CMV Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is returned.
22 to 17	MINF[6:1]	Illegal access master information These bits show the master information of the originator of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 7	Reserved	When read, the value after reset is returned.
6 to 4	VCH[2:0]	Illegal access channel These bits show the channel number (0 to 7) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	Reserved	When read, the value after reset is returned.
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DMAC1. 0: No illegal access has occurred in the DMAC1 1: Illegal access has occurred in the DMAC1 If an illegal access occurs in the DMAC1 while this bit is 0, this bit is set, and MINF6 to MINF1 and VCH2 to VCH0 store their respective information. If an illegal access occurs in the DMAC1 while this bit is 1, this bit remains 1, and MINF6 to MINF1 and VCH2 to VCH0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.11 DTSCMV — DTS Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8038$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VCH[6:0]							—	—	—	VF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.21 DTSCMV Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is returned.
22 to 17	MINF[6:1]	Illegal access master information These bits show the master information of the originator of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 11	Reserved	When read, the value after reset is returned.
10 to 4	VCH[6:0]	Illegal access channel These bits show the channel number (0 to 127) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	Reserved	When read, the value after reset is returned.
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DTS. 0: No illegal access has occurred in the DTS 1: Illegal access has occurred in the DTS If an illegal access occurs in the DTS while this bit is 0, this bit is set, and MINF6 to MINF1 and VCH6 to VCH0 store their respective information. If an illegal access occurs in the DTS while this bit is 1, this bit remains 1, and MINF6 to MINF1 and VCH6 to VCH0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.12 CMVC — Register Access Protection Violation Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 803C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSVC	DM1VC	DM0VC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 7.22 CMVC Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	DTSVC	DTS illegal access flag clear The DTS illegal access flag (DTSCMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.
1	DM1VC	DMAC1 illegal access flag clear The DMAC1 illegal access flag (DM1CMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.
0	DM0VC	DMAC0 illegal access flag clear The DMAC0 illegal access flag (DM0CMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.

7.9.2.13 DTSPRn — DTS Channel Priority Setting Register (n = 0 to 7)

- DTSPR0

Access: This register can be read/written in 32-bit units.

Address: FFFF 8060$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS15PR[1:0]		DTS14PR[1:0]		DTS13PR[1:0]		DTS12PR[1:0]		DTS11PR[1:0]		DTS10PR[1:0]		DTS9PR[1:0]		DTS8PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS7PR[1:0]		DTS6PR[1:0]		DTS5PR[1:0]		DTS4PR[1:0]		DTS3PR[1:0]		DTS2PR[1:0]		DTS1PR[1:0]		DTS0PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.23 DTSPR0 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[15:0] PR[1:0]	DTS channel [15:0] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR1

Access: This register can be read/written in 32-bit units.

Address: FFFF 8064$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS31PR[1:0]		DTS30PR[1:0]		DTS29PR[1:0]		DTS28PR[1:0]		DTS27PR[1:0]		DTS26PR[1:0]		DTS25PR[1:0]		DTS24PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS23PR[1:0]		DTS22PR[1:0]		DTS21PR[1:0]		DTS20PR[1:0]		DTS19PR[1:0]		DTS18PR[1:0]		DTS17PR[1:0]		DTS16PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.24 DTSPR1 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[31:16] PR[1:0]	DTS channel [31:16] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR2

Access: This register can be read/written in 32-bit units.

Address: FFFF 8068$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS47PR[1:0]		DTS46PR[1:0]		DTS45PR[1:0]		DTS44PR[1:0]		DTS43PR[1:0]		DTS42PR[1:0]		DTS41PR[1:0]		DTS40PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS39PR[1:0]		DTS38PR[1:0]		DTS37PR[1:0]		DTS36PR[1:0]		DTS35PR[1:0]		DTS34PR[1:0]		DTS33PR[1:0]		DTS32PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.25 DTSPR2 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[47:32] PR[1:0]	DTS channel [47:32] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR3

Access: This register can be read/written in 32-bit units.

Address: FFFF 806C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS63PR[1:0]		DTS62PR[1:0]		DTS61PR[1:0]		DTS60PR[1:0]		DTS59PR[1:0]		DTS58PR[1:0]		DTS57PR[1:0]		DTS56PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS55PR[1:0]		DTS54PR[1:0]		DTS53PR[1:0]		DTS52PR[1:0]		DTS51PR[1:0]		DTS50PR[1:0]		DTS49PR[1:0]		DTS48PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.26 DTSPR3 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[63:48] PR[1:0]	DTS channel [63:48] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR4

Access: This register can be read/written in 32-bit units.

Address: FFFF 8070$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS79PR[1:0]		DTS78PR[1:0]		DTS77PR[1:0]		DTS76PR[1:0]		DTS75PR[1:0]		DTS74PR[1:0]		DTS73PR[1:0]		DTS72PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS71PR[1:0]		DTS70PR[1:0]		DTS69PR[1:0]		DTS68PR[1:0]		DTS67PR[1:0]		DTS66PR[1:0]		DTS65PR[1:0]		DTS64PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.27 DTSPR4 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[79:64] PR[1:0]	DTS channel [79:64] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR5

Access: This register can be read/written in 32-bit units.

Address: FFFF 8074$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS95PR[1:0]		DTS94PR[1:0]		DTS93PR[1:0]		DTS92PR[1:0]		DTS91PR[1:0]		DTS90PR[1:0]		DTS89PR[1:0]		DTS88PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS87PR[1:0]		DTS86PR[1:0]		DTS85PR[1:0]		DTS84PR[1:0]		DTS83PR[1:0]		DTS82PR[1:0]		DTS81PR[1:0]		DTS80PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.28 DTSPR5 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[95:80] PR[1:0]	DTS channel [95:80] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR6

Access: This register can be read/written in 32-bit units.

Address: FFFF 8078$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS111PR[1:0]		DTS110PR[1:0]		DTS109PR[1:0]		DTS108PR[1:0]		DTS107PR[1:0]		DTS106PR[1:0]		DTS105PR[1:0]		DTS104PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS103PR[1:0]		DTS102PR[1:0]		DTS101PR[1:0]		DTS100PR[1:0]		DTS99PR[1:0]		DTS98PR[1:0]		DTS97PR[1:0]		DTS96PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.29 DTSPR6 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[111:96] PR[1:0]	DTS channel [111:96] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR7

Access: This register can be read/written in 32-bit units.

Address: FFFF 807C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS127PR[1:0]		DTS126PR[1:0]		DTS125PR[1:0]		DTS124PR[1:0]		DTS123PR[1:0]		DTS122PR[1:0]		DTS121PR[1:0]		DTS120PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS119PR[1:0]		DTS118PR[1:0]		DTS117PR[1:0]		DTS116PR[1:0]		DTS115PR[1:0]		DTS114PR[1:0]		DTS113PR[1:0]		DTS112PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.30 DTSPR7 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[127:112] PR[1:0]	DTS channel [127:112] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

7.9.2.14 DTRECCTL — DTSRAM ECC Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8080H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 7.31 DTRECCTL Register Contents

Bit Position	Bit Name	Function
31, 30	PROT[1:0]	These bits enable or disable writing to the ECCDIS and SECDIS bits. The written data is not retained. The read value is always 0. These bits should be written when (PROT1, PROT0) = (0, 1)
29 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	SECDIS	DTSRAM SEC error correction disable This bit enables or disables SEC error correction when the ECCDIS bit is 0. 1-bit ECC error detection operation is always executed when ECCDIS bit is 0 regardless of the state of this bit. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: Error correction is enabled when the SEC error is detected. 1: Error correction is disabled when the SEC error is detected.
0	ECCDIS	DTSRAM ECC disable This bit enables or disables DTSRAM ECC error detection and correction. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: Error detection and correction are enabled. 1: Error detection and correction are disabled. **Note**: The encoding function is effective when error detection and correction are disabled.

7.9.2.15 DTRERINT — DTSRAM Error Notification Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8084$_H$

Value after reset: 0000 0002$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 7.32 DTRERINT Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	DEDIE	DTSRAM 2-bit error notification enable This bit enables or disables notification of 2-bit error detection to the ECM when the ECCDIS bit in DTRECCTL is 0. 0: Notification of 2-bit error to ECM is disabled. 1: Notification of 2-bit error to ECM is enabled.
0	SEDIE	DTSRAM 1-bit error notification enable This bit enables or disables notification of 1-bit error detection to the ECM when the ECCDIS bit in DTRECCTL is 0. 0: Notification of 1-bit error to ECM is disabled. 1: Notification of 1-bit error to ECM is enabled.

7.9.2.16 DTRTSCTL — DTSRAM Test Control Register

This register is used for ECC test (self-diagnosis). It enables setting of ECC test mode and selection of ECC data to be written to the DTSRAM.

Access: This register can be read/written in 32-bit units.

Address: FFFF 8094$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECCTST	DATSEL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 7.33 DTRTSCTL Register Contents

Bit Position	Bit Name	Function
31, 30	PROT[1:0]	These bits enable or disable writing to the ECCTST and DATSEL bits. The written data is not retained. The read value is always 0. These bits should be written when (PROT1, PROT0) = (0, 1)
29 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	ECCTST	DTSRAM ECC Test Mode This bit enables or disables DTSRAM ECC test mode. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: ECC test mode is enabled. 1: ECC test mode is disabled.
0	DATSEL	ECC Test Data Selection This bit is valid when ECCTST is 1 and selects ECC data to be written to the DTSRAM. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: ECC encoded from the written data is used. 1: The value specified by the DTSRAM test write data register (DTRTWDAT) is used.

7.9.2.17 DTRTWDAT — DTSRAM Test Write Data Register

This register is used for ECC test (self-diagnosis). It specifies ECC data to be written to the DTSRAM after ECC test mode is enabled (ECCTST = 1).

Access: This register can be read/written in 32-bit units.

Address: FFFF 8098$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TWDAT[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.34 DTRTWDAT Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
6 to 0	TWDAT[6:0]	ECC Test Write Data These bits specify ECC data to be written to the DTSRAM when DTRTSCTL.ECCTST = 1 and DTRTSCTL.DATSEL = 1. Writing to these bits is enabled when DTRTSCTL.ECCTST = 1 When DTRTSCTL.ECCTST = 0, these bits cannot be written and their read value is 0.

7.9.2.18 DTRTRDAT — DTSRAM Test Read Data Register

This register is used for ECC test (self-diagnosis). It reads out ECC data of the DTSRAM after ECC test mode is enabled (ECCTST = 1).

Access: This register can be read in 32-bit units.

Address: FFFF 809C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TRDAT[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.35 DTRTRDAT Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, the value after reset is returned.
6 to 0	TRDAT[6:0]	ECC Test Read Data These bits retain the last ECC data read out from the DTSRAM when DTRTSCTL.ECCTST = 1. When DTRTSCTL.ECCTST = 0, the read value of these bits is 0.

7.9.2.19 DMnnCM — DMAC Channel Master Setting Register (nn = 00 to 07, 10 to17)

Access: This register can be read/written in 32-bit units.

Address: FFFF 8100_H + 4 × Ch. No. n (n = 0 to 7)

FFFF 8120_H + 4 × Ch. No. n - 10 (n = 10 to 17)

Value after reset: $0000\ 0010_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PEID[2:0]			SPID[1:0]		UM	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 7.36 DMnnCM Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
6 to 4	PEID[2:0]	Channel master PEID setting Specify the PEID information of the master assigned to the channel.
3, 2	SPID[1:0]	Channel master SPID setting Specify the SPID information used by the master assigned to the channel.
1	UM	Channel master UM setting Specifies the UM information of the master assigned to the channel.
0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

CAUTION

DM00CM to DM07CM configure the channel master information of the DMAC0 channels 0 to 7 respectively.

DM10CM to DM17CM configure the channel master information of the DMAC1 channels 0 to 7 respectively.

For information about the functions this register offers, see **Section 7.5, Reliability Function**.

7.9.2.20 DTSnnnCM — DTS Channel Master Setting Register (nnn = 000 to 127)

Access: This register can be read/written in 32-bit units.

Address: FFFF 8200$_H$ + 4$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PEID[2:0]			SPID[1:0]		UM	—
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMC[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.37 DTSnnnCM Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
22 to 20	PEID[2:0]	Channel master PEID setting Specify the PEID information of the master assigned to the channel.
19, 18	SPID[1:0]	Channel master SPID setting Specify the SPID information used by the master assigned to the channel.
17	UM	Channel master UM setting Specifies the UM information of the master assigned to the channel.
16	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
15 to 0	CMC[15:0]	Transfer count compare In terms of contents, this field is the same as bits [15:0] of the Section **7.11.3.8, DTTCCnnn — DTS Transfer Count Compare Register**.

CAUTION

DTS000CM to DTS127CM configure the channel master information of the DTS channels 0 to 127 respectively.

For information about the functions this register offers, see **Section 7.5, Reliability Function**.

CAUTION

The 16 low-order bits of this register are shared with the DTS transfer count compare register, one of the DTS channel registers.

If you write to this register, the DTS transfer count compare register is updated as well.

Recommended setup procedure of the DTS channel master setting register

When the special master configures the overall DMA operation, the channel master setting must be configured in bits 22 to 17 in this register, and 0 must be specified in bits 15 to 0 and reserved bits 31 to 23 and 16.

When the master assigned to the channel updates the value of the transfer count compare, use the DTS transfer count compare register instead.

Reserved bits of this register can be read or written. Our recommendation is as follows. When you write to those bits, write 0. When you read from those bits, discard the read value by software.

7.10 DMAC Channel Register

7.10.1 DMAC Channel Register Address

Address = Base address "FFFF 8000_H" + Offset address

Table 7.38 DMAC Channel Register Address

Module Name	Offset Address	Register Symbol	Meaning	Accessed Permission		
				Special Master	General Master	At secure boot
DMASS	$0400_H + 40_H \times$ [channel number]*1	DSAn	DMAC source address	√	√	*2
DMASS	$0404_H + 40_H \times$ [channel number]*1	DDAn	DMAC destination address	√	√	*2
DMASS	$0408_H + 40_H \times$ [channel number]*1	DTCn	DMAC transfer count	√	√	*2
DMASS	$040C_H + 40_H \times$ [channel number]*1	DTCTn	DMAC transfer control	√	√	*2
DMASS	$0410_H + 40_H \times$ [channel number]*1	DRSAn	DMAC reload source address	√	√	*2
DMASS	$0414_H + 40_H \times$ [channel number]*1	DRDAn	DMAC reload destination address	√	√	*2
DMASS	$0418_H + 40_H \times$ [channel number]*1	DRTCn	DMAC reload transfer count	√	√	*2
DMASS	$041C_H + 40_H \times$ [channel number]*1	DTCCn	DMAC transfer count compare	√	√	*2
DMASS	$0420_H + 40_H \times$ [channel number]*1	DCENn	DMAC channel operation enable setting	√	√	*2
DMASS	$0424_H + 40_H \times$ [channel number]*1	DCSTn	DMAC transfer status	√	√	
DMASS	$0428_H + 40_H \times$ [channel number]*1	DCSTSn	DMAC transfer status set	√	√	*2
DMASS	$042C_H + 40_H \times$ [channel number]*1	DCSTCn	DMAC transfer status clear	√	√	*2
DMASS	$0430_H + 40_H \times$ [channel number]*1	DTFRn	DTFR setting	√	√	*2
DMASS	$0434_H + 40_H \times$ [channel number]*1	DTFRRQn	DTFR transfer request status	√	√	
DMASS	$0438_H + 40_H \times$ [channel number]*1	DTFRRQCn	DTFR transfer request clear	√	√	*2

Note 1. The [channel number] in the offset addresses and "n" in the register symbols are numbers in the range from 0 to 15, and the correspondence between the channel number n and the channel is as follows.

Note 2. At secure boot, write access on DMAC0 channels 0 to 7 is masked. For details on secure boot, refer to **Section 32, Intelligent Cryptographic Unit E (ICUSE)**.

Channel number n	Channel	Channel number n	Channel
0	DMAC0 channel 0	8	DMAC1 channel 0
1	DMAC0 channel 1	9	DMAC1 channel 1
2	DMAC0 channel 2	10	DMAC1 channel 2
3	DMAC0 channel 3	11	DMAC1 channel 3
4	DMAC0 channel 4	12	DMAC1 channel 4
5	DMAC0 channel 5	13	DMAC1 channel 5
6	DMAC0 channel 6	14	DMAC1 channel 6
7	DMAC0 channel 7	15	DMAC1 channel 7

7.10.2 Details of DMAC Channel Registers

The “n” in the register symbols indicates the DMA channel number (n = 0 to 15).

7.10.2.1 DSAn — DMAC Source Address Register

Access: This register can be read/written in 32-bit units.

Address: $FFFF8400_H + 40_H \times$ Ch. No. n (n = 0 to 15)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.39 DSAn Register Contents

Bit Position	Bit Name	Function
31 to 0	SA[31:0]	Source address Specify the DMA transfer source address. Those bits are updated whenever a DMA cycle is executed. If you read from those bits, the transfer source address for the next DMA cycle is read.

CAUTIONS

1. Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.
2. The address must be set up while the DTE bit is 0.
3. DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit).
   Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	SA3	SA2	SA1	SA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.2 DDAn — DMAC Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8404_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.40 DDAn Register Contents

Bit Position	Bit Name	Function
31 to 0	DA[31:0]	Destination address Specify the DMA transfer destination address. Those bits are updated whenever a DMA cycle is executed. If you read from those bits, the transfer destination address for the next DMA cycle is read.

CAUTIONS

1. Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.
2. The address must be set up while the DTE bit is 0.
3. If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the destination address is updated.
4. DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit).
   Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	DA3	DA2	DA1	DA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.3 DTCn — DMAC Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8408_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.41 DTCn Register Contents

Bit Position	Bit Name	Function
31 to 16	ARC[15:0]	Address reload count Specify the number of transfers until the address reload when the reload function 2 is used, and also specifies the number of transfers when the block transfer 2 is used. If you read from those bits during DMA transfer, the address reload count for the next DMA cycle is read. When the reload function 2 or block transfer 2 is used, ARC[15:0] is decremented by one for every DMA cycle. When the reload function 2 or block transfer 2 is not used, ARC[15:0] bits are not updated. If the value is 0000_H, it means that the number of transfers until the address reload when the reload function 2 is used and the number of transfers when the block transfer 2 is used are 65536.
15 to 0	TRC[15:0]	Transfer count Configure the number of transfers. TRC[15:0] is decremented by one whenever a DMA cycle is executed. If you read from those bits, the remaining number of transfers for the next DMA cycle is read. If the reload function is not used, after the last transfer is complete, the value at the completion (0000_H) is retained.

TRC15-0	Operation
0000_H	The number of transfers is 65536, or the transfer is complete.
0001_H	The number of transfers or remaining transfers is 1.
:	:
$FFFF_H$	The number of transfers or remaining transfers is 65535.

CAUTION

1. Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.
2. If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the transfer count and the address reload count are updated.

7.10.2.4 DTCTn — DMAC Transfer Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 840C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	ESE	DRS	—	—	—	—	—	CHNSEL[2:0]			CHNE[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCE	TCE	MLE	RLD2M[1:0]		RLD1M[1:0]		DACM[1:0]		SACM[1:0]		DS[2:0]			TRM[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.42 DTCTn Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
27	ESE	Transfer error case DMA transfer disable setting Configures whether a DMA cycle is executed when the DCSTn.ER bit is set due to DMA transfer error. If this bit is cleared to 0, even when the DCSTn.ER bit is set due to DMA transfer error, the following DMA cycles can be executed. If this bit is set to 1, the following DMA cycles are not executed when the DCSTn.ER bit is set due to DMA transfer error. 0: DMA cycles are executed while the DCSTn.ER bit is set. 1: DMA cycles are not executed while the DCSTn.ER bit is set.
26	DRS	DMA transfer request selection assignment Selects the type of DMA transfer requests to be accepted. 0: Software DMA transfer request 1: Hardware DMA transfer request
25 to 21	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
20 to 18	CHNSEL[2:0]	Next channel in the chain Specify the next channel in the chain. The next channel must be another channel in the same DMAC. You cannot specify a channel in the different DMAC or in the DTS. You cannot specify the same channel for the next channel. (If you do, correct operation is not guaranteed.)
17, 16	CHNE[1:0]	Chain enable Select the chain function. 00: Disabled 01: Chain at the last transfer A chain request is generated at the completion of the DMA cycle in which the remaining transfer count is one. 10: Setting prohibited (No guarantee of operation) 11: Always chain A chain request is generated at the completion of every DMA cycle.

Table 7.42 DTCTn Register Contents (2/3)

Bit Position	Bit Name	Function
15	CCE	Transfer count match interrupt enable If this bit is set, a transfer count match interrupt is generated at the completion of the DMA cycle in which the transfer count compare register and the remaining transfer count have the same value.
14	TCE	Transfer completion interrupt enable If this bit is set, a transfer completion interrupt is generated at the completion of the last transfer.
13	MLE	Continuous transfer enable If this bit is set, the DTE bit is not cleared at the completion of DMA transfer. In addition, even if the TC bit is not cleared, DMA transfer starts when there is a DMA transfer request. 0: The DTE bit is cleared at the completion of DMA transfer. In addition, the next DMA transfer can start only after the TC bit is cleared. 1: The DTE bit is not cleared at the completion of DMA transfer. In addition, even if the TC bit is not cleared, DMA transfer starts when there is a DMA transfer request.
12, 11	RLD2M[1:0]	Reload function 2 setting Configure the reload function 2. 00: Reload function 2 is disabled. 01: Reload function 2 is enabled. The source address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 10: Reload function 2 is enabled. The destination address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 11: Reload function 2 is enabled. The source address, destination address, and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1.
10, 9	RLD1M[1:0]	Reload function 1 setting Configure the reload function 1. 00: Reload function 1 is disabled. 01: Reload function 1 is enabled. The source address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 10: Reload function 1 is enabled. The destination address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 11: Reload function 1 is enabled. The source address, destination address, and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.)
8, 7	DACM[1:0]	Destination address count direction Specify the direction of counting for the destination address.

DACM1	DACM0	Direction of Counting
0	0	Up
0	1	Down
1	0	Fixed
1	1	Setting prohibited (No guarantee of operation)

Table 7.42 DTCTn Register Contents (3/3)

Bit Position	Bit Name	Function
6, 5	SACM[1:0]	Source address count direction Specify the direction of counting for the source address.

SACM1	SACM0	Direction of Counting
0	0	Up
0	1	Down
1	0	Fixed
1	1	Setting prohibited (No guarantee of operation)

Bit Position	Bit Name	Function
4 to 2	DS[2:0]	Transfer data size Specify the transfer data size.

DS2	DS1	DS0	Transfer Data Size
0	0	0	8 bits
0	0	1	16 bits
0	1	0	32 bits
0	1	1	64 bits
1	0	0	128 bits
Other than the above			Setting prohibited (No guarantee of operation)

Bit Position	Bit Name	Function
1, 0	TRM[1:0]	Transfer mode Specify the DMA transfer mode. 00: Single transfer 01: Block transfer 1 (The number of transfers is specified by the transfer count.) 10: Block transfer 2 (The number of transfers is specified by the address reload count.) 11: Setting prohibited (No guarantee of operation)

CAUTIONS

1. Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.
2. If prohibited settings are made to some of the bits, correct operation is not guaranteed.

7.10.2.5 DRSAn — DMAC Reload Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8410$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.43 DRSAn Register Contents

Bit Position	Bit Name	Function
31 to 0	RSA[31:0]	Reload source address Specify the source address to be reloaded to the DMA source address register when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit).

Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	RSA3	RSA2	RSA1	RSA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.6 DRDAn — DMAC Reload Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8414$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.44 DRDAn Register Contents

Bit Position	Bit Name	Function
31 to 0	RDA[31:0]	Reload destination address Specify the destination address to be reloaded to the DMA destination address register when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit).

Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	RDA3	RDA2	RDA1	RDA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.7 DRTCn — DMAC Reload Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8418$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RARC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTRC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.45 DRTCn Register Contents

Bit Position	Bit Name	Function
31 to 16	RARC[15:0]	Reload address reload count Specify the value to be loaded to the address reload count in the transfer count register at the timing of reload when the reload function 2 is used.
15 to 0	RTRC[15:0]	Reload transfer count Specify the value to be loaded to the transfer count in the transfer count register at the timing of reload when the reload function 1 is used.

7.10.2.8 DTCCn — DMAC Transfer Count Compare Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 841C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.46 DTCCn Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
15 to 0	CMC[15:0]	Transfer count compare Configure the transfer count to be compared to the transfer count register. At the completion of the DMA cycle in which the remaining transfer count is the same as the value in this register, the transfer count match flag (DCSTn.CC) in the DMAC transfer status register is set. Furthermore, if the transfer count match interrupt enable (DTCTn.CCE) bit is 1, a transfer count match interrupt is generated. If 0000$_H$ is set, comparison with the transfer count is disabled. In this case, the transfer count match flag in the DMAC transfer status register is never set, and a transfer count match interrupt is never generated.

CAUTION

Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.

7.10.2.9 DCENn — DMAC Channel Operation Enable Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8420$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DTE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.47 DCENn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DTE	Channel operation enable Specifies whether to enable or disable the transfer operation of the channel. If the DTE bit is 1, DMA transfer starts when there is a DMA transfer request. If the MLE bit is 0, the DTE bit is cleared automatically at the completion of the DMA transfer. In addition, if 0 is written to the DTE bit during DMA transfer, the DMA transfer is suspended. If 1 is written to the DTE bit during suspension, the suspension is cleared and the DMA transfer resumes. 0: Channel operation is disabled/Channel suspended 1: Channel operation is enabled/Channel suspension cleared

7.10.2.10 DCSTn — DMAC Transfer Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 8424$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ERWR	—	—	CY	ER	—	CC	TC	—	—	DR	SR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.48 DCSTn Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 12	Reserved	When read, the value after reset is returned.
11	ERWR	DMA Transfer Error occurring cycle This bit is updated at the same time as setting of the DMA transfer error flag (ER), indicating which cycle of read or write the DMA transfer error occurs in. This bit is not updated when a new DMA transfer error occurs after the ER bit has been set. If the ER bit is cleared, this bit is also cleared to 0. 0: DMA transfer error occurs in read cycle. 1: DMA transfer error occurs in write cycle.
10, 9	Reserved	When read, the value after reset is returned.
8	CY	DMA cycle execution state This bit shows whether a DMA cycle is ongoing in this channel. 0: DMA cycle is not ongoing. 1: DMA cycle is ongoing.
7	ER	Transfer error flag This bit is set when DMA transfer error is generated. If this bit is 1 and the DTCTn.ESE bit is set, a DMA cycle is not executed when a DMA transfer request is generated. 0: No DMA transfer error is generated 1: DMA transfer error is generated
6	Reserved	When read, the value after reset is returned.
5	CC	Transfer count match flag This bit is set at the completion of the DMA cycle in which the remaining transfer count is the same as the value set in the transfer compare register. 0: No compare match has occurred with the transfer count compare register. 1: Compare match has occurred with the transfer count compare register.

Table 7.48 DCSTn Register Contents (2/2)

Bit Position	Bit Name	Function
4	TC	Transfer completion flag This bit is set at the completion of the last transfer and shows whether the DMA transfer is complete. If the MLE bit is 0 and this bit is 1, a DMA cycle is not executed when a DMA transfer request is generated. 0: DMA transfer incomplete 1: DMA transfer complete
3, 2	Reserved	When read, the value after reset is returned.
1	DR	Hardware DMA transfer request status This bit shows whether there is a hardware DMA transfer request (DMARQ) from the DTFR. This bit changes regardless of the value of the DTE bit when a hardware DMA transfer request from the DTFR is generated. If the software DMA transfer request is selected by the transfer request selection bit (DRS) in the DMAC transfer control register, this bit is not set even when a hardware DMA transfer request is input from the DTFR. 0: There is no hardware DMA transfer request 1: There is a hardware DMA transfer request
0	SR	Software DMA transfer request flag This bit shows whether there is a software DMA transfer request (DMARQ). This bit is automatically cleared when DMA transfer is executed. A user can set this bit by writing 1 to the SRS bit in the DMAC transfer status set register (DCSTSn). In addition, a user can clear this bit by writing 1 to the SRC bit in the DMAC transfer status clear register (DCSTCn), but if this is done, the ongoing DMA transfer is aborted and cannot be resumed. 0: There is no software DMA transfer request 1: There is a software DMA transfer request

7.10.2.11 DCSTSn — DMAC Transfer Status Set Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8428$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SRS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.49 DCSTSn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	SRS	Software DMA transfer request flag A user can set the software DMA transfer request flag (SR) by writing 1 to this bit. 0 is always read from this bit.

7.10.2.12 DCSTCn — DMAC Transfer Status Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 842C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ERC	—	CCC	TCC	—	—	—	SRC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R	R	R	R/W

Table 7.50 DCSTCn Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
7	ERC	Transfer error flag clear The DMA transfer error flag (ER) can be cleared by writing 1 to this bit. 0 is always read from this bit.
6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	CCC	Transfer count match flag clear The transfer count match flag (CC) can be cleared by writing 1 to this bit. 0 is always read from this bit.
4	TCC	Transfer completion flag clear The transfer completion flag (TC) can be cleared by writing 1 to this bit. 0 is always read from this bit.
3 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	SRC	Software DMA transfer request flag clear The software DMA transfer request flag (SR) can be cleared by writing 1 to this bit. 0 is always read from this bit.

7.10.2.13 DTFRn — DTFR Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8430$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	REQSEL[6:0]							REQEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.51 DTFRn Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
7 to 1	REQSEL[6:0]	Hardware DMA transfer source selection Select one out of 128 hardware DMA transfer sources as the hardware DMA transfer request. 000_0000: Selecting the DMACTRG[0] input : 111_1111: Selecting the DMACTRG[127] input
0	REQEN	Hardware DMA transfer source selection enable This bit enables or disables the hardware DMA transfer source selection. 0: Hardware DMA transfer source selection is disabled. 1: Hardware DMA transfer source selection is enabled. If this bit is 0, even when the hardware DMA transfer source selected by the REQSEL6 to REQSEL0 bits is activated, it is not recognized as a hardware DMA transfer request, and a hardware DMA transfer request is not generated.

7.10.2.14 DTFRRQn — DTFR Transfer Request Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 8434$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQ
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.52 DTFRRQn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned.
0	DRQ	Hardware DMA transfer request status If this bit is set, it means that a hardware DMA transfer request exists or is retained. • If the hardware DMA transfer request is an edge detection type*1 This bit shows whether a hardware DMA transfer request generated by edge detection is retained. When the DMA transfer request acceptance signal (DMAAKn) from the DMAC is asserted, this bit is automatically cleared. A user can clear this bit by writing 1 to the DTFRRQC.DRQC bit. • If the hardware DMA transfer request is a level input type*1 This bit shows whether there is a hardware DMA transfer request input. Even when the DMA transfer request acceptance signal (DMAAKn) from the DMAC is asserted, this bit is not automatically cleared. In addition, this bit is not cleared even when a user writes to the DTFRRQC.DRQC bit. This bit changes regardless of the value of the DTFRn.REQEN bit when a hardware DMA transfer request is generated. 0: There is no hardware DMA transfer request 1: There is a hardware DMA transfer request

Note 1. Whether the hardware DMA transfer request is an edge detection type or level input type depends on the hardware DMA transfer source selected by DTFRn.REQSEL. The hardware DMA transfer request is only an edge detection type.

7.10.2.15 DTFRRQCn — DTFR Transfer Request Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8438$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.53 DTFRRQCn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DRQC	Hardware DMA transfer request clear If the hardware DMA transfer request is an edge detection type*1, a user can clear the DTFRRQ.DRQ bit by writing 1 to this bit. If the hardware DMA transfer request is a level input type*1, the DTFRRQ.DRQ bit cannot be cleared by writing to this bit. 0 is always read from this bit.

Note 1. Whether the hardware DMA transfer request is an edge detection type or level input type depends on the hardware DMA transfer source selected by DTFRn.REQSEL. The hardware DMA transfer request is only an edge detection type.

7.11 DTS Channel Register

7.11.1 Transfer information of the DTS (TI)

7.11.1.1 Structure of the TI

The transfer information of the DTS is called TI. One set of TI consists of 32 bits. 8 sets of TI are assigned for each channel. The 8 sets of TI is called TI-A, TI-B, TI-C, TI-D, TI- E, TI-F, TI-G, and TI-H.

Figure 7.12 shows the structure of the TI.

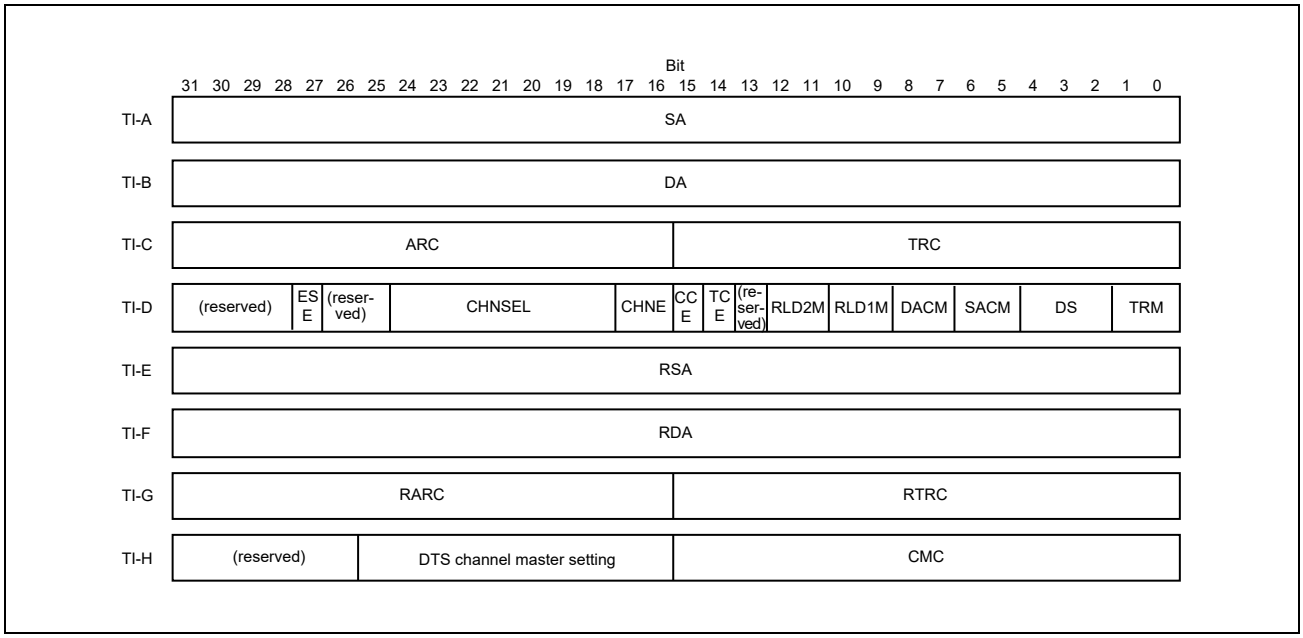


Figure 7.12 Structure of the TI

7.11.1.2 Organization of the TI in the DTSRAM

A user indirectly accesses the DTSRAM by way of the DTS channel registers for each channel and the DTS channel master setting registers.

Therefore, usually, a user does not have to think about the address organization of the TI in the DTSRAM. As an exception, when ECC error occurs while the DTSRAM is read, the address of the ECC error in the DTSRAM is stored to the DTSRAM error register 2 (DTSER2), one of the global registers. In order to know which channel and TI have generated the error, you need to understand the address organization of the TI in the DTSRAM.

Figure 7.13 shows the address organization of the TI in the DTSRAM.

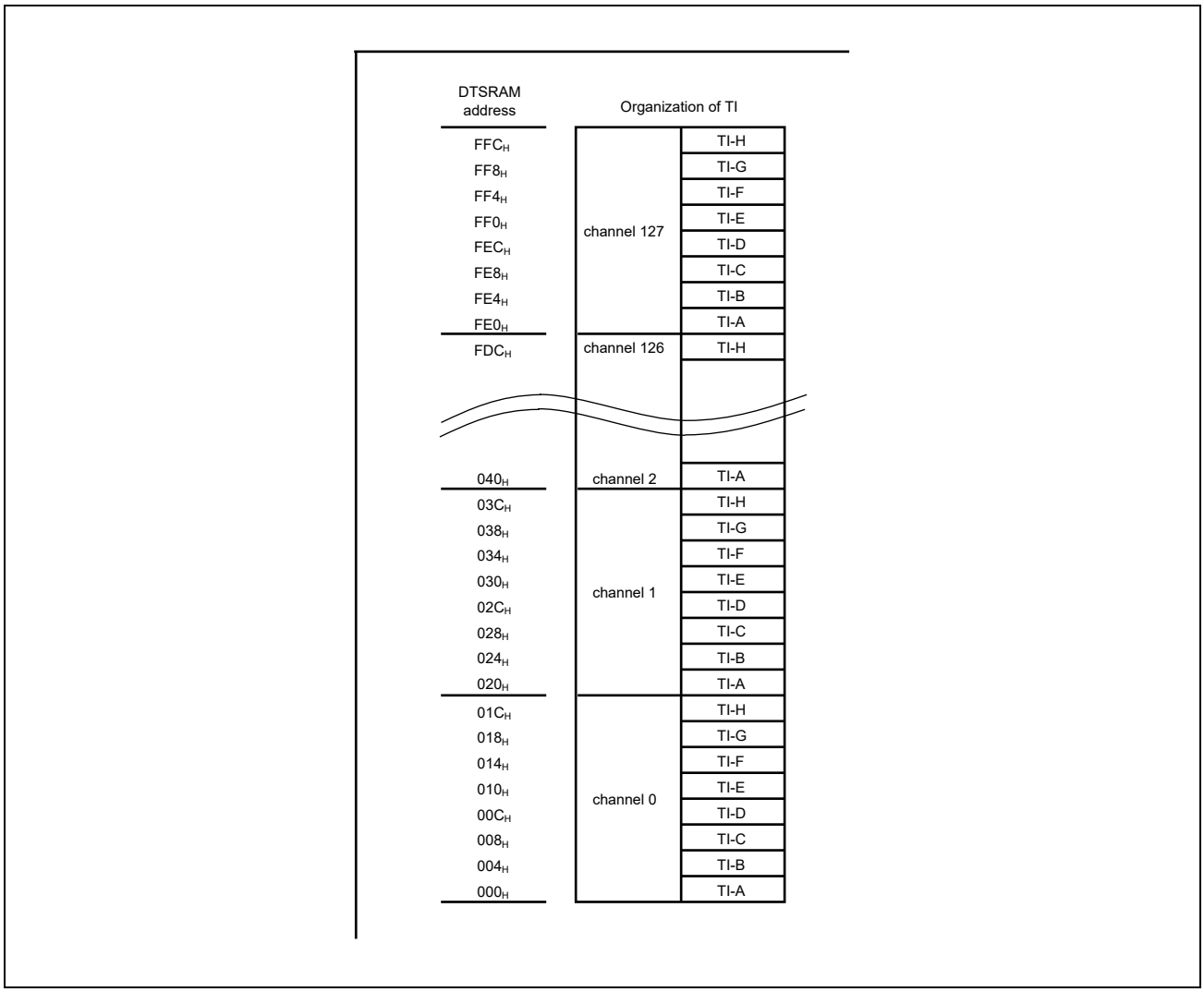


Figure 7.13 Organization of the TI in the DTSRAM

7.11.1.3 Accessing the TI

TI-A can be accessed by way of the DTS source address register (DTSAnnn) for each channel.

TI-B can be accessed by way of the DTS destination address register (DTDAnnn) for each channel.

TI-C can be accessed by way of the DTS transfer count register (DTTCnnn) for each channel.

TI-D can be accessed by way of the DTS transfer control register (DTTCTnnn) for each channel.

TI-E can be accessed by way of the DTS reload source address register (DTRSAnnn) for each channel.

TI-F can be accessed by way of the DTS reload destination address register (DTRDAnnn) for each channel.

TI-G can be accessed by way of the DTS reload transfer count register (DTRTCnnn) for each channel.

TI-H can be accessed by way of the channel master setting register (DTSnnnCM), which is a global register, and the transfer count compare register (DTTCCnnn) for each channel.

7.11.1.4 Caution about accessing the TI

The data of the DTS channel master setting register and the data of the DTS transfer count compare register are stored to the same TI-H.

Access to the DTS channel master setting register (DTSnnnCM) is actually 32-bit access to the whole TI-H. Therefore, when you write to the DTS channel master setting register, the values of the 16 lower-order bits of the DTS transfer count compare (CMC) are updated at the same time. When you read from the DTS channel master setting register, the value of the DTS transfer count compare (CMC) is read into the 16 lower-order bits.

When you read from the DTS transfer count compare register (DTTCCnnn), 32-bit data is read from the TI-H, but only data of the 16 lower-order bits is actually seen in the result of the register read. When you write to the DTS transfer count compare register (DTTCCnnn), read-modify-write access to the 16 lower-order bits for the 32-bit TI-H is made. Data in the TI immediately after the reset is undefined. It should be noted that, if you try to write to the DTS transfer count compare register (DTTCCnnn) before setting up the DTS channel master setting register, ECC error may be detected during the read of the read-modify-write access.

Bits 31 to 26 of the TI-H are not used, but you can read and write those bits by accessing the DTS channel master setting register. Our recommendation is as follows. When you write to those bits, write 0.When you read from those bits, discard the read value by software.

After the reset, the values in the DTSRAM, which stores the TI, are undefined. After the reset, if you read TI before you write to the TI, ECC error is generated. Therefore, the first access to the following registers after the reset must be write access. The first access after the reset should never be read access.

- DTS source address register (DTSAnnn)
- DTS destination address register (DTDAnnn)
- DTS transfer count register (DTTCnnn)
- DTS transfer control register (DTTCTnnn)
- DTS reload source address register (DTRSAnnn)
- DTS reload destination address register (DTRDAnnn)
- DTS reload transfer count register (DTRTCnnn)
- Channel master setting registers (DTSnnnCM)

In addition, after the reset, the first access to the DTS transfer count compare register (DTTCCnnn) must be always after write access to the channel master setting register (DTSnnnCM).

You can access the TI from a CPU while the DTS is executing DMA transfer. But if you do so, the following should be noted.

- While a channel is executing DMA transfer, the TI of the channel should not be updated by TI access from a CPU. If this situation happens, the result of the DMA transfer and the contents of the TI may mismatch.
- If TI access is requested from a CPU while a TI fetch or TI write back is executed, the TI access is executed after the completion of the TI fetch or TI write back. If a TI fetch or TI write back is requested while a TI access request from a CPU is processed, the TI fetch or TI write back is executed after the processing of the TI access is complete.

7.11.2 DTS Channel Register Address

Address = Base address "FFFF 9000_H" + Offset address

Table 7.54 DTS Channel Register Address

Module Name	Offset Address	Register Symbol	Meaning	Accessed Permission	
				Special Master	General Master
DMASS	$0000_H + 40_H \times$ [channel number]*1	DTSAnnn	DTS source address	√	√
DMASS	$0004_H + 40_H \times$ [channel number]*1	DTDAnnn	DTS destination address	√	√
DMASS	$0008_H + 40_H \times$ [channel number]*1	DTTCnnn	DTS transfer count	√	√
DMASS	$000C_H + 40_H \times$ [channel number]*1	DTTCTnnn	DTS transfer control	√	√
DMASS	$0010_H + 40_H \times$ [channel number]*1	DTRSAnnn	DTS reload source address	√	√
DMASS	$0014_H + 40_H \times$ [channel number]*1	DTRDAnnn	DTS reload destination address	√	√
DMASS	$0018_H + 40_H \times$ [channel number]*1	DTRTCnnn	DTS reload transfer count	√	√
DMASS	$001C_H + 40_H \times$ [channel number]*1	DTTCCnnn	DTS transfer count compare	√	√
DMASS	$0020_H + 40_H \times$ [channel number]*1	DTFSLnnn	DTSFSL operation setting	√	√
DMASS	$0024_H + 40_H \times$ [channel number]*1	DTFSTnnn	DTSFSL transfer request status	√	√
DMASS	$0028_H + 40_H \times$ [channel number]*1	DTFSSnnn	DTSFSL transfer request set	√	√
DMASS	$002C_H + 40_H \times$ [channel number]*1	DTFSCnnn	DTSFSL transfer request clear	√	√

Note 1. The [channel number] in the offset addresses is a number in the range from 0 to 127.
The "nnn" in the register symbols is a 3-digit number in the range from 000 to 127.

7.11.3 Details of DTS Channel Registers

The “nnn” in the register symbols indicates the DTS channel number (nnn = 000 to 127).

7.11.3.1 DTSAnnn — DTS Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9000_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA[31:16]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.55 DTSAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	SA[31:0]	Source address Specify the DMA transfer source address. SA[31:0] is updated at the timing of the TI write back and retains the DMA transfer source address for the next DMA transfer.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	SA3	SA2	SA1	SA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.2 DTDAnnn — DTS Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9004_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA[31:16]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.56 DTDAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	DA[31:0]	Destination address Specify the DMA transfer destination address. DA[31:0] is updated at the timing of the TI write back and retains the DMA transfer destination address for the next DMA transfer.

CAUTIONS

1. If DMA transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the destination address is updated.
2. DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	DA3	DA2	DA1	DA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.3 DTTCnnn — DTS Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9008_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARC[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRC[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.57 DTTCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	ARC[15:0]	Address reload count Specify the number of transfers until the address reload when the reload function 2 is used, and also specifies the number of transfers when the block transfer 2 is used. When the reload function 2 or block transfer 2 is used, ARC[15:0] is decremented by one for every DMA cycle and updated at the timing of the TI write back. When the reload function 2 or block transfer 2 is not used, ARC[15:0] is not updated. If 0000_H is set, address reload is disabled. If the value at the start of a DMA cycle is 0000_H, subtraction from the address reload count does not proceed even after the DMA cycle.
15 to 0	TRC[15:0]	Transfer count Configure the number of transfers. TRC[15:0] is decremented by one whenever a DMA cycle is executed. It is updated at the timing of the TI write back. If the reload function is not used, after the last transfer is complete, the value at the completion (0000_H) is retained. If 0000_H is set, DMA transfer is not executed even when a DMA transfer request is accepted.

TRC[15:0]	Operation
0000_H	Transfer is disabled or complete.
0001_H	The number of transfers or remaining transfers is 1.
:	:
$FFFF_H$	The number of transfers or remaining transfers is 65535.

CAUTIONS

1. If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the transfer count and the address reload count are updated.
2. Unlike a DMAC, "0000_H" in the transfer count of the DTS does not mean "65536 transfers" but means that transfer is disabled or complete.

7.11.3.4 DTTCTnnn — DTS Transfer Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 900C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	ESE	—	—	CHNSEL[6:0]							CHNE[1:0]	
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCE	TCE	—	RLD2M[1:0]		RLD1M[1:0]		DACM[1:0]		SACM[1:0]		DS[2:0]			TRM[1:0]	
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.58 DTTCTnnn Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
27	ESE	Transfer error case DMA transfer abort setting Specifies whether to abort DMA transfer when DMA transfer error is generated. If this bit is cleared to 0, DMA transfer continues when DMA transfer error is generated. If this bit is set to 1, DMA transfer is aborted when DMA transfer error is generated. 0: DMA transfer continues when DMA transfer error is generated. 1: DMA transfer is aborted when DMA transfer error is generated.
26, 25	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
24 to 18	CHNSEL[6:0]	Next channel in the chain Specify the next channel in the chain. The next channel must be another channel in the DTS. You cannot specify a channel in a DMAC. You cannot specify the same channel for the next channel. (If you do, correct operation is not guaranteed.)
17, 16	CHNE[1:0]	Chain enable Select the chain function. 00: Disabled 01: Chain at the last transfer A chain request is generated at the completion of the DMA cycle in which the remaining transfer count is one. 10: Setting prohibited (No guarantee of operation) 11: Always chain A chain request is generated at the completion of every DMA cycle.
15	CCE	Transfer count match interrupt enable If this bit is set, a transfer count match interrupt is generated at the completion of the DMA cycle in which the transfer count compare register and the remaining transfer count have the same value.
14	TCE	Transfer completion interrupt enable If this bit is set, a transfer completion interrupt is generated at the completion of the last transfer.

Table 7.58 DTTCTnnn Register Contents (2/3)

Bit Position	Bit Name	Function
13	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
12, 11	RLD2M[1:0]	Reload function 2 setting Configure the reload function 2. 00: Reload function 2 is disabled. 01: Reload function 2 is enabled. The source address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 10: Reload function 2 is enabled. The destination address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 11: Reload function 2 is enabled. The source address, destination address, and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1.
10, 9	RLD1M[1:0]	Reload function 1 setting Configure the reload function 1. 00: Reload function 1 is disabled. 01: Reload function 1 is enabled. The source address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 10: Reload function 1 is enabled. The destination address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 11: Reload function 1 is enabled. The source address, destination address, and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1.(If the reload function 2 is enabled, the address reload count is also reloaded.)
8, 7	DACM[1:0]	Destination address count direction Specify the direction of counting for the destination address.

DACM1	DACM0	Direction of Counting
0	0	Up
0	1	Down
1	0	Fixed
1	1	Setting prohibited (No guarantee of operation)

6, 5	SACM[1:0]	Source address count direction Specify the direction of counting for the source address.

SACM1	SACM0	Direction of Counting
0	0	Up
0	1	Down
1	0	Fixed
1	1	Setting prohibited (No guarantee of operation)

Table 7.58 DTTCTnnn Register Contents (3/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>4 to 2</td><td>DS[2:0]</td><td>Transfer data size
Specify the transfer data size.
<table>
<tr><th>DS2</th><th>DS1</th><th>DS0</th><th>Transfer Data Size</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>8 bits</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>16 bits</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>32 bits</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>64 bits</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>128 bits</td></tr>
<tr><td colspan="3">Other than the above</td><td>Setting prohibited (No guarantee of operation)</td></tr>
</table></td></tr>
<tr><td>1, 0</td><td>TRM[1:0]</td><td>Transfer mode
Specify the DMA transfer mode.
00: Single transfer
01: Block transfer 1 (The number of transfers is specified by the transfer count.)
10: Block transfer 2 (The number of transfers is specified by the address reload count.)
11: Setting prohibited (No guarantee of operation)</td></tr>
</table>

CAUTIONS

1. If prohibited settings are made to some of the bits, correct operation is not guaranteed.
2. Bits 31 to 28, 26, 25, and 13 are unused, but you can read and write those bits. Our recommendation is as follows. When you write to those bits, write 0. When you read from those bits, discard the read value.

7.11.3.5 DTRSAnnn — DTS Reload Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9010_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSA[31:16]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSA[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.59 DTRSAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	RSA[31:0]	Reload source address Specify the source address to be reloaded when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	RSA3	RSA2	RSA1	RSA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.6 DTRDAnnn — DTS Reload Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9014_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDA[31:16]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDA[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.60 DTRDAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	RDA[31:0]	Reload destination address Specify the destination address to be reloaded when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	RDA3	RDA2	RDA1	RDA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.7 DTRTCnnn — DTS Reload Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9018$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RARC[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTRC[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.61 DTRTCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	RARC[15:0]	Reload address reload count Specify the value to be reloaded to the address reload count when the reload function 2 is used.
15 to 0	RTRC[15:0]	Reload transfer count Specify the value to be reloaded to the transfer count when the reload function 1 is used.

RTRC[15:0]	Operation
0000$_H$	No DMA transfer
0001$_H$	1 transfer
:	:
FFFF$_H$	65535 transfers

7.11.3.8 DTTCCnnn — DTS Transfer Count Compare Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 901C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMC[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.62 DTTCCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
15 to 0	CMC[15:0]	Transfer count compare Configure the transfer count to be compared to the transfer count register. If the transfer count match interrupt enable (DTTCTnnn.CCE) bit in the DTS transfer control register is 1, a transfer count match interrupt is generated at the completion of the DMA cycle in which the remaining transfer count is the same as the value in this register. If 0000$_H$ is set, comparison with the transfer count is disabled. In this case, a transfer count match interrupt is not generated.

CAUTION

This register must be accessed after the DTS channel master setting register is set up.

If you access this register without setting up the DTS channel master setting register after the reset, ECC error may be generated during access to this register.

7.11.3.9 DTFSLnnn — DTSFSL Operation Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9020$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	REQEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.63 DTFSLnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	REQEN	DMA transfer request enable This bit selects whether a DMA transfer request from this channel retained in the DTSFSL is used as a candidate in DTS channel arbitration. 0: A DMA transfer request from this channel is not used as a candidate in DTS channel arbitration. 1: A DMA transfer request from this channel is used as a candidate in DTS channel arbitration. If this bit is 0, even if the DTSFSL retains a DMA transfer request, this channel is not a candidate in DTS channel arbitration inside the DTSFSL, and consequently a DMA transfer request from this channel is not generated.

7.11.3.10 DTFSTnnn — DTSFSL Transfer Request Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 9024$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQ
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.64 DTFSTnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned.
0	DRQ	DMA transfer request pending This bit shows whether a DMA transfer request of this channel is pending. This bit is set when input of a hardware transfer source is detected, or when software writes “1” to the DTFSSnnn.DRQ bit. This bit is automatically cleared when the DTS receives the DMA transfer request signal while the DTSFSL is requesting DMA transfer of this channel. Alternatively, software can clear this bit by writing 1 to the DTFSCnnn.DRQC bit. 0: A DMA transfer request is not pending. 1: A DMA transfer request is pending.

7.11.3.11 DTFSSnnn — DTSFSL Transfer Request Set Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9028$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.65 DTFSSnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DRQS	DMA transfer request set A user can set the DTFSTnnn.DRQ bit by writing 1 to this bit. 0 is always read from this bit.

7.11.3.12 DTFSCnnn — DTSFSL Transfer Request Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 902C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.66 DTFSCnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DRQC	DMA transfer request clear A user can clear the DTFSTnnn.DRQ bit by writing 1 to this bit. 0 is always read from this bit.

7.12 DMA/DTS Trigger Source Select Registers

7.12.1 DMA/DTS Trigger Source Select Register Addresses

DMA/DTS trigger sources are each selected from two trigger sources: Primary trigger source and Secondary trigger source. For DMA trigger sources, refer to **Table 7.8** For DTS trigger sources, refer to **Table 7.9** When setting up DMA transfer, configure these registers.

Table 7.67 DMA/DTS Trigger Source Select Register Addresses

Offset Address	Register Abbreviation	Meaning	Access Permission	
			Special Master	General Master
$FFD41000_H$	DTSTRGSEL0	DTS Primary/Secondary select register 0	√	√
$FFD41004_H$	DTSTRGSEL1	DTS Primary/Secondary select register 1	√	√
$FFD41008_H$	DTSTRGSEL2	DTS Primary/Secondary select register 2	√	√
$FFD4100C_H$	DTSTRGSEL3	DTS Primary/Secondary select register 3	√	√
$FFD41100_H$	DMATRGSEL0	DMA Primary/Secondary select register 0	√	√
$FFD41104_H$	DMATRGSEL1	DMA Primary/Secondary select register 1	√	√
$FFD41108_H$	DMATRGSEL2	DMA Primary/Secondary select register 2	√	√
$FFD4110C_H$	DMATRGSEL3	DMA Primary/Secondary select register 3	√	√

7.12.2 Details of DMA/DTS Trigger Source Select Registers

7.12.2.1 DTSTRGSELn — DTS Trigger Source Select Register n

Access This register can be read/written in 32-bit units.

Address $FFD4\ 1000_H + 4_H \times n$ (n = 0 to 3)

Value after rese $0000\ 0000_H$

- DTSTRGSEL0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTSSEL31	DTSSEL30	DTSSEL29	DTSSEL28	DTSSEL27	DTSSEL26	DTSSEL25	DTSSEL24	DTSSEL23	DTSSEL22	DTSSEL21	DTSSEL20	DTSSEL19	DTSSEL18	DTSSEL17	DTSSEL16
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTSSEL15	DTSSEL14	DTSSEL13	DTSSEL12	DTSSEL11	DTSSEL10	DTSSEL9	DTSSEL8	DTSSEL7	DTSSEL6	DTSSEL5	DTSSEL4	DTSSEL3	DTSSEL2	DTSSEL1	DTSSEL0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DTSTRGSEL1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTSSEL63	DTSSEL62	DTSSEL61	DTSSEL60	DTSSEL59	DTSSEL58	DTSSEL57	DTSSEL56	DTSSEL55	DTSSEL54	DTSSEL53	DTSSEL52	DTSSEL51	DTSSEL50	DTSSEL49	DTSSEL48
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTSSEL47	DTSSEL46	DTSSEL45	DTSSEL44	DTSSEL43	DTSSEL42	DTSSEL41	DTSSEL40	DTSSEL39	DTSSEL38	DTSSEL37	DTSSEL36	DTSSEL35	DTSSEL34	DTSSEL33	DTSSEL32
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DTSTRGSEL2

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTSSEL95	DTSSEL94	DTSSEL93	DTSSEL92	DTSSEL91	DTSSEL90	DTSSEL89	DTSSEL88	DTSSEL87	DTSSEL86	DTSSEL85	DTSSEL84	DTSSEL83	DTSSEL82	DTSSEL81	DTSSEL80
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTSSEL79	DTSSEL78	DTSSEL77	DTSSEL76	DTSSEL75	DTSSEL74	DTSSEL73	DTSSEL72	DTSSEL71	DTSSEL70	DTSSEL69	DTSSEL68	DTSSEL67	DTSSEL66	DTSSEL65	DTSSEL64
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DTSTRGSEL3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTSSEL127	DTSSEL126	DTSSEL125	DTSSEL124	DTSSEL123	DTSSEL122	DTSSEL121	DTSSEL120	DTSSEL119	DTSSEL118	DTSSEL117	DTSSEL116	DTSSEL115	DTSSEL114	DTSSEL113	DTSSEL112
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTSSEL111	DTSSEL110	DTSSEL109	DTSSEL108	DTSSEL107	DTSSEL106	DTSSEL105	DTSSEL104	DTSSEL103	DTSSEL102	DTSSEL101	DTSSEL100	DTSSEL99	DTSSEL98	DTSSEL97	DTSSEL96
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.68 DTSTRGSELn Contents

Bit Position	Bit Name	Function
31 to 0	DTSSEL[31 + 32n:0 + 32n]	DTS trigger source select DTSSEL0 to 127 select DTS trigger sources input into DTS. DTSSEL=0: Primary channel is selected DTSSEL=1: Secondary channel is selected

7.12.2.2 DMATRGSELn — DMA Trigger Source Select Register n

Access This register can be read/written in 32-bit units.

Address FFD4 1100_H + 4_H ×n (n = 0 to 3)

Value after reset 0000 0000_H

- DMATRGSEL0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMASEL31	DMASEL30	DMASEL29	DMASEL28	DMASEL27	DMASEL26	DMASEL25	DMASEL24	DMASEL23	DMASEL22	DMASEL21	DMASEL20	DMASEL19	DMASEL18	DMASEL17	DMASEL16
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMASEL15	DMASEL14	DMASEL13	DMASEL12	DMASEL11	DMASEL10	DMASEL9	DMASEL8	DMASEL7	DMASEL6	DMASEL5	DMASEL4	DMASEL3	DMASEL2	DMASEL1	DMASEL0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMATRGSEL1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMASEL63	DMASEL62	DMASEL61	DMASEL60	DMASEL59	DMASEL58	DMASEL57	DMASEL56	DMASEL55	DMASEL54	DMASEL53	DMASEL52	DMASEL51	DMASEL50	DMASEL49	DMASEL48
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMASEL47	DMASEL46	DMASEL45	DMASEL44	DMASEL43	DMASEL42	DMASEL41	DMASEL40	DMASEL39	DMASEL38	DMASEL37	DMASEL36	DMASEL35	DMASEL34	DMASEL33	DMASEL32
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMATRGSEL2

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMASEL95	DMASEL94	DMASEL93	DMASEL92	DMASEL91	DMASEL90	DMASEL89	DMASEL88	DMASEL87	DMASEL86	DMASEL85	DMASEL84	DMASEL83	DMASEL82	DMASEL81	DMASEL80
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMASEL79	DMASEL78	DMASEL77	DMASEL76	DMASEL75	DMASEL74	DMASEL73	DMASEL72	DMASEL71	DMASEL70	DMASEL69	DMASEL68	DMASEL67	DMASEL66	DMASEL65	DMASEL64
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMATRGSEL3

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMASEL127	DMASEL126	DMASEL125	DMASEL124	DMASEL123	DMASEL122	DMASEL121	DMASEL120	DMASEL119	DMASEL118	DMASEL117	DMASEL116	DMASEL115	DMASEL114	DMASEL113	DMASEL112
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DMASEL111	DMASEL110	DMASEL109	DMASEL108	DMASEL107	DMASEL106	DMASEL105	DMASEL104	DMASEL103	DMASEL102	DMASEL101	DMASEL100	DMASEL99	DMASEL98	DMASEL97	DMASEL96
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.69 DMATRGSELn Register Contents

Bit Position	Bit Name	Function
31 to 0	DMASEL[31 + 32n:0 + 32n]	DMA trigger source select DMASEL0 to 127 select DMA trigger sources input into DMA. DMASEL = 0: Primary channel is selected DMASEL = 1: Secondary channel is selected

Section 8 Resets

8.1 Features of the RH850/C1M-A Reset

- The $\overline{\text{RESET}}$ pin incorporates a noise canceller.
- The source of a reset can be determined by referring to the reset source determination register.
- The CPU can assert a reset signal by setting a register.

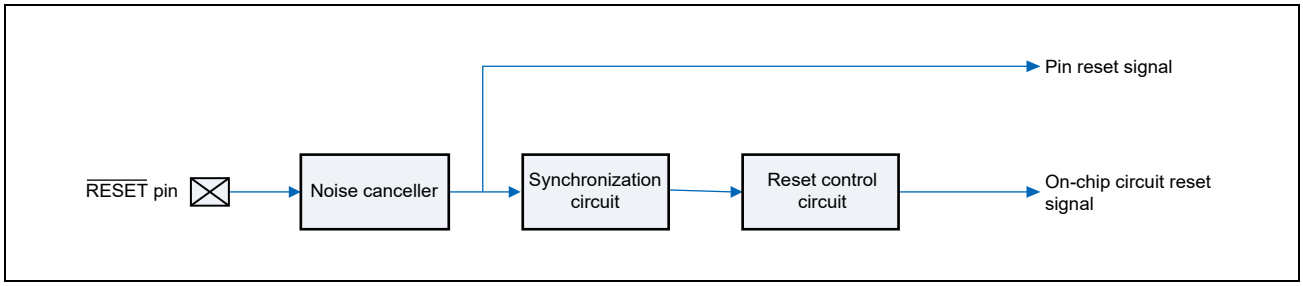


Figure 8.1 Reset Circuit

8.2 Reset State

8.2.1 External Reset State

When a low-level pulse longer than the noise cancel width (t_{RESNCW}) is input to the $\overline{\text{RESET}}$ pin, external reset is received and this LSI transits to the external reset state. When external reset is received, individual pins transit to the external reset state. Refer to **Table 2.63** in **Section 2, Pins**, for the state of individual pins during external reset.

The low-level pulse width of the input signal must be longer than t_{RESW} (t_{cyc}) because the $\overline{\text{RESET}}$ pin incorporates a noise cancelling circuit. This LSI transits to the internal reset state by driving the $\overline{\text{RESET}}$ pin to the high-level after the required low-level period.

CAUTION

Refer to **Section 39, Electrical Characteristics**, for t_{RESNCW} and t_{RESW}.

8.2.2 Internal Reset State

When a high-level signal longer than the noise cancel width (t_{RESNCW}) is input to the $\overline{\text{RESET}}$ pin in the external reset state, this product transits to the internal reset state. Refer to **Table 2.63** in **Section 2, Pins**, for the state of individual pins.

The internal reset state is released after external reset release, and the CPU starts the reset exception process.

CAUTION

Refer to **Section 39, Electrical Characteristics**, for t_{RESNCW}.

8.3 Reset Sources

The following are the reset sources of this product.

Some registers are only initialized by the external reset state. That is, they are only initialized by input of the low level on the $\overline{\text{RESET}}$ pin.

However, most registers are initialized by either an external or internal reset state. This means that they are initialized by a reset from any source.

Regarding sources for the initialization of registers, see the descriptions of the registers in the relevant sections. Where there is no explicit description of the sources for resetting and initializing a register, the register is initialized by either an external or internal reset state. That is, such registers are initialized by a reset from any source. However, the value after resets of some registers are undefined. Take care since the values of such registers following a reset are not fixed.

Source	Description
When the $\overline{\text{RESET}}$ pin is driven to the low level	Transits to the external reset state.
When a reset request is issued from the ECM	Transits to the internal reset state.
When a reset is issued from the debugger (when a forced reset is issued)	Transits to the external reset state.
When the CPU sets the software reset request register	Transits to the internal reset state

8.4 Register Specifications

8.4.1 List of Registers

Reset registers are listed in the following table.

Table 8.1 Registers

Module Name	Register Name	Symbol	Address	Access Protection
SYS	Reset source determination register	RESF	FFF8 2800_H	
SYS	Reset source clear register	RESFC	FFF8 2808_H	
SYS	Software reset request register	SWRESA	FFF8 $AC18_H$	PROT1PHCMD

8.4.2 RESF — Reset Source Determination Register

This register determines the reset source.

The flags of this register can be cleared by the RESFC register and can only be initialized by an external reset, but not an internal reset.

Access: This register is readable in 32-bit units.

Address: FFF8 2800$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RESF1	RESF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 8.2 RESF Register Contents

Bit Position	Bit Name	Function
1	RESF1	ECM Reset Indicates that a reset event occurred. 0: Event not occurred 1: Event occurred
0	RESF0	Software Reset Indicates that a reset event occurred. 0: Event not occurred 1: Event occurred

8.4.3 RESFC — Reset Source Clear Register

This register clears the reset source indicated by the RESF register.

This register is always read as 0000 0000_H, and can only be initialized by an external reset, but not an internal reset.

Access: This register is readable/writable in 32-bit units.

Address: FFF8 2808_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RESFC1	RESFC0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 8.3 RESFC Register Contents

Bit Position	Bit Name	Function
1	RESFC1	ECM reset Clears the status bit. 0: Do not clear 1: Clear
0	RESFC0	Software reset Clears the status bit. 0: Do not clear 1: Clear

8.4.4 SWRESA — Software Reset Request Register

This register generates an internal reset when accessed. A software reset is issued when 1 is written to SWRESA.

This register can be protected by the PROT1PHCMD register.

This register can be reset by either an internal or external reset.

Access: This register is readable/writable in 32-bit units.

Address: FFF8 AC18$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SWRESA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 8.4 SWRESA Register Contents

Bit Position	Bit Name	Function
0	SWRESA	0: — (default) 1: Internal reset enabled (internal reset triggered)

Example) Sequence of writing to the SWRESA register

Writing to SWRESA requires the special sequence to unlock protection described below.

A protection error is displayed in the PROTIPS register if the following steps are not observed.

For details of the PROT1PS register, see **Section 10.2.11, PROT1PS — Protect 1 Status Register**.

Step 1. Write the fixed value (0000 00A5$_H$) to the PROT1PHCMD register.

Step 2. Write the setting value (0000 0001$_H$) to the SWRESA register.

Step 3. Write the inverse (FFFF FFFE$_H$) to the SWRESA register.

Step 4. Write the new setting value (0000 0001$_H$) to the SWRESA register.

The value 0000 0001_H can be written to the SWRESA register by following the steps described above.

If the above steps are not followed correctly, the sequence to unlock protection fails. The value 0000 0001_H will not be set to the SWRESA register and 1 is set to the PROT1PS. PROTERR bit. In a case of failure, repeat the sequence from the beginning.

If writing to a different register is attempted during the unlocking sequence, or if an access to a different register is made in response to the interrupt occurred during the unlocking sequence, writing to the protected registers is handled as follows.

- When the writing is to another register in the same module*1, writing to the protected register fails and the PROT1PS.PROTERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

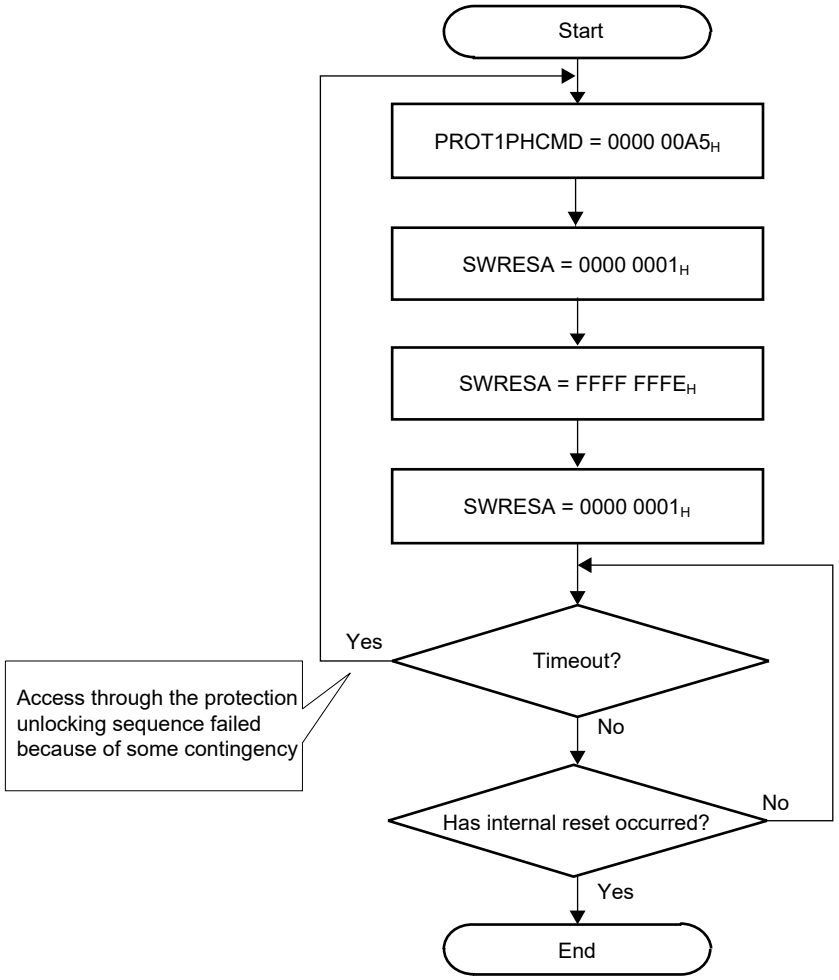
Reading of another register while the sequence is in progress does not lead to failure.

Note 1. For the target registers, see the note in **Section 10.2.10, PROT1PHCMD — Protect 1 Command Register**.

8.5 Procedure

8.5.1 Software Reset

Figure 8.2 shows the flow for setting a software reset.



Note: Timeout time should be 5 µs or longer.

Figure 8.2 Flow of Software Reset

8.6 Notes

The debug function includes a reset masking function. To prevent unexpected operations of this function, fix the $\overline{\text{DCUTRST}}$ pin to the low level when the debug function is not used.

Section 9 Power Supply Circuit

This section contains a generic description of the RH850/C1M-A power supply circuit.

9.1 Features of the RH850/C1M-A Power Supply Circuit

- Power supply

  The power supply pins and the power supply voltages to be used are listed in the following table.
  During operations, supply the specified voltages to all power lines. When stopping operation, turn off all of the power supplies.

  – Connect AnVCC and RVCC to the same electric potential.

Table 9.1 List of Power Supply Pins

Power Supply Name for Power Supply Pin	Pin Name	Power Supply Voltage in Operation	Use of Power Supply
SYSVCC		4.5 V to 5.5 V	Power for system logic, PLL
VCC		4.5 V to 5.5 V	Power for oscillator, flash programming, and ports (5 V)
VDD		1.15 V to 1.35 V	Power for the core (direct power supply) and connection of the stabilizing capacitor for core power
AnVCC		4.5 V to 5.5 V	Power for SAR AD
	AnVREFH	4.5 V to 5.5 V	Reference voltage for SAR AD
RVCC		4.5 V to 5.5 V	Power for RDC

9.2 Procedure

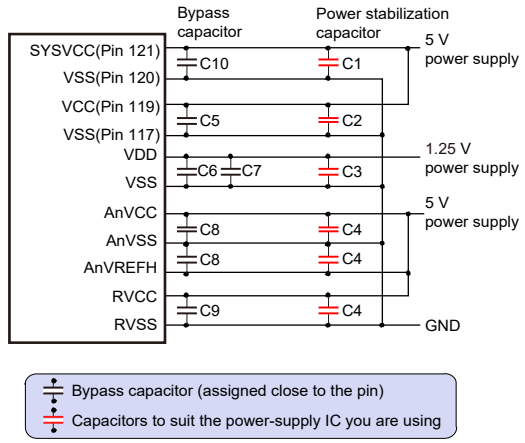
9.2.1 Power-On Sequence

For power-on and -off specifications, see **Section 39, Electrical Characteristics**.

9.3 Notes

9.3.1 Example of Connections between Power-Supply Pins and External Capacitors

The example shown below is for reference only. Take the voltage fluctuations seen under the actual conditions of your system into consideration in evaluating and assigning optimal capacitors in terms of bringing the power supply pins of the microcomputer into accord with the voltage levels in the product specifications.



The following are reference values of the power stabilization capacitors for the power-supply IC.
Determine the optimal capacitances to suit the power-supply IC you are using.

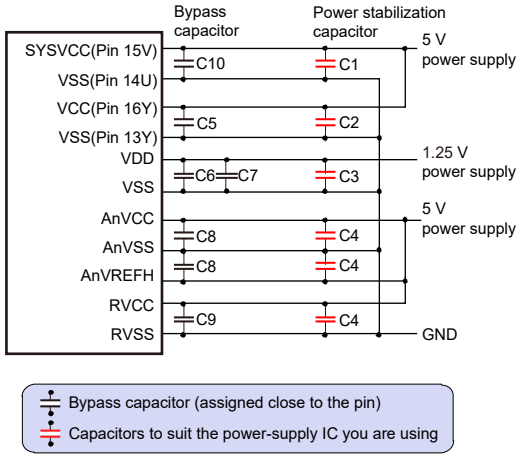
Essential Component	Reference Value
C1	≥ 10 µF
C2	≥ 10 µF
C3	≥ 10 µF
C4	≥ 10 to 100 µF

The following are reference values of the bypass capacitors for this device.

Essential Component*1	Reference Value
C5	≥ 0.22 µF or 0.1 µF × 2
C6	≥ 0.1 µF
C7	≥ 1.0 µF
C8	≥ 0.1µF
C9	≥ 0.1µF
C10	≥ 0.1µF

Note 1. We recommend adding a bypass capacitor for each power supply pin.

Figure 9.1 Example of Connections between Power-Supply Pins and External Capacitors (RH850/C1M-A1 (QFP))



The following are reference values of the power stabilization capacitors for the power-supply IC.
Determine the optimal capacitances to suit the power-supply IC you are using.

Essential Component	Reference Value
C1	≥ 10 µF
C2	≥ 10 µF
C3	≥ 10 µF
C4	≥ 10 to 100 µF

The following are reference values of the bypass capacitors for this device.

Essential Component*1	Reference Value
C5	≥ 0.22 µF or 0.1 µF × 2
C6	≥ 0.1 µF
C7	≥ 1.0 µF
C8	≥ 0.1µF
C9	≥ 0.1µF
C10	≥ 0.1µF

Note 1. We recommend adding bypass capacitors directly under the package.

Figure 9.2 Example of Connections between Power-Supply Pins and External Capacitors (RH850/C1M-A2 (BGA))

Section 10 Clock Controller

This section contains a generic description of the Clock Controller.

The first part of this section describes all RH850/C1M-A specific properties, such as the types of clock, the block diagrams, and the input/output pins. The remainder of the section describes the functions and registers of the clock controller.

10.1 Features of RH850/C1M-A Clock Controller

- An oscillation circuit is included (Main OSC).
- Oscillation at 20 MHz is possible without using external capacitors (only with limited oscillators).
- On-chip PLLs are included and can be used to multiply clock frequencies.
- PLL0 is capable of operating as a spread spectrum clock generator (SSCG) for reducing radiated noise.
  The clock signal produced by PLL1 is not frequency modulated (i.e. is a clean clock signal) and so is suitable for use with timers and communications modules.
- A low-speed internal oscillator (LS IntOSC) is included.
- Error notifications to ECM can be generated when the Main OSC goes outside the frequency range specified by the clock monitor.
- Inrush currents after release from the reset state can be suppressed by having software increase the clock frequencies in a stepwise manner. The division ratios for the CPU clock and the peripheral clocks can be selected with register settings (1/4, 1/2, and 1/1)*[1].

Note 1. Correct operations of ADCC and RDC3A are not guaranteed if 1/4 or 1/2 is selected. Other peripheral circuits operate by using the clock frequency that was input, but their electrical characteristics are not guaranteed. Therefore, only use the product after the sequence for shifting the clock gear up has reached the point where the division ratio for dividers 0A and 1A is 1/1 (no division).

10.1.1 Type of Clocks

Table 10.1 shows list of clocks, **Table 10.2** shows the operation clocks of each unit functional module, and **Figure 10.1** shows the block diagram of clocks.

Table 10.1 List of Clocks

When operating at 20 MHz input frequency (Main OSC)

Clock Name	Symbol	Clock Frequency*1			Remarks
		Division Ratio of Divider 0A/ Divider 1A			
		1/1	1/2	1/4	
CPU clock	CLK_CPU	320 MHz(*1) 240 MHz(*2)	160 MHz(*1) 120 MHz(*2)	80 MHz(*1) 60 MHz(*2)	PLL0 (SSCG can be selected)
GRAM clock	CLK_GRAM	160 MHz(*1) 120 MHz(*2)	80 MHz(*1) 60 MHz(*2)	40 MHz(*1) 30 MHz(*2)	
High speed peripheral clock	CLK_HSB	80 MHz	40 MHz	20 MHz	
Low speed peripheral clock	CLK_LSB	40 MHz	20 MHz	10 MHz	
Sub-CPU clock	CLK_EMU_H	320 MHz(*1) 240 MHz(*2)	160 MHz(*1) 120 MHz(*2)	80 MHz(*1) 60 MHz(*2)	
Motor control hardware accelerator clock	CLK_EMU_L	160 MHz(*1) 120 MHz(*2)	80 MHz(*1) 60 MHz(*2)	40 MHz(*1) 30 MHz(*2)	*1: For C1M-A2 *2: For C1M-A1
Unmodulated high speed peripheral clock	CLKC_HSB	80 MHz	40 MHz	20 MHz	PLL1 (SSCG cannot be selected)
Unmodulated low speed peripheral clock	CLKC_LSB	40 MHz	20 MHz	10 MHz	
Low speed internal clock	CLK_LIOSC	240 kHz			
WDTA counter clock	WDTCLKI	250 kHz/240 kHz			1/80 of main OSC or CLK LIOSC, selectable

Note 1. Only use the product after the sequence for shifting the clock gear up has reached the point where the division ratio for dividers 0A and 1A is 1/1 (no division).

Table 10.2 Clocks and Functional Modules

Clock Name	Functional Module Name
CPU clock	CPU1, CPU2
GRAM clock	GRAM
High speed peripheral clock	INTC, INTIF, DMAC, DTS, CSIH, RS-CANFD, EMU3, RDC3A
Low speed peripheral clock	ECM, DCRA, PORT, Data FLASH, FLASH Controller, FLSCI3, EINT, RLIN3, ADCC, ADPA
Sub-CPU clock	EMU3
Motor control hardware accelerator clock	EMU3
Unmodulated high speed peripheral clock	TAUD, TAUJ, TSG3, TAPA, TPBA, PIC1B, PIC2D, ENCA, EMU3
Unmodulated low speed peripheral clock	RS-CANFD, RLIN3, SCI3, WDTA, SWDTA, OSTM, RDC3A, ADCC, RSENT
Low speed internal clock	CLMA
WDTA counter clock	WDTA, SWDTA

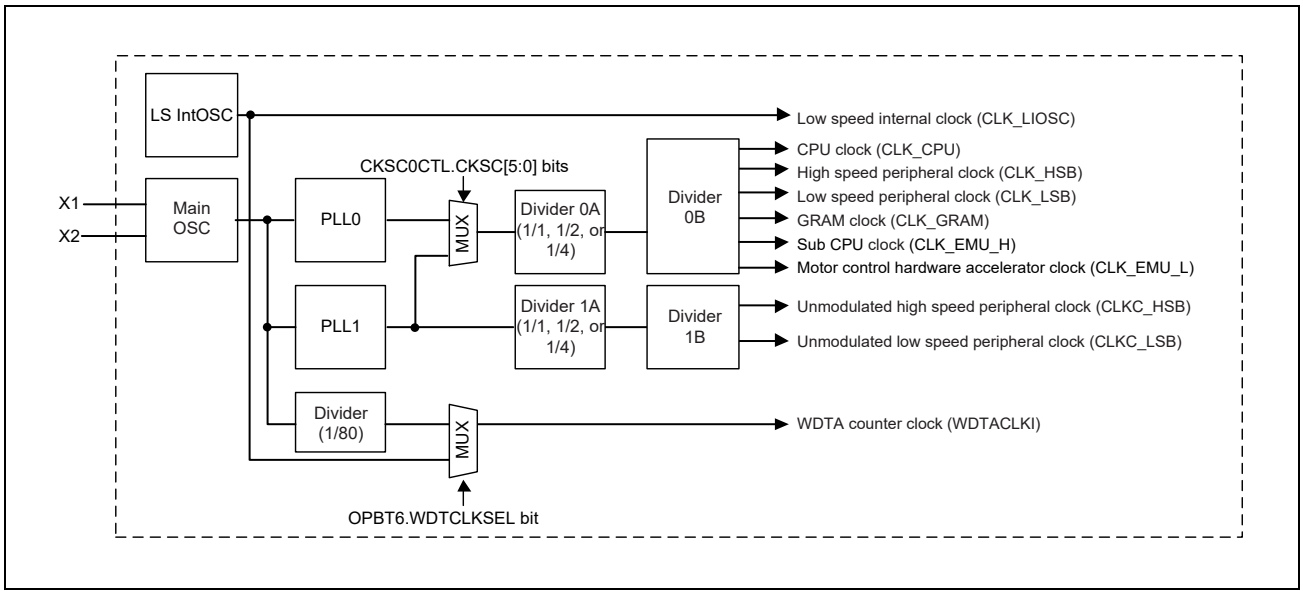


Figure 10.1 Clock Controller Block Diagram

10.1.2 External Input/Output Pins

Table 10.3 shows the pins related to the clock controller.

Table 10.3 Pins Related to the Clock Controller

Name	Pin Name	Input/Output	Function
Crystal input	X1	Input	Connected to a crystal oscillator
Crystal output	X2	Output	Connected to a crystal oscillator

10.1.3 How to Connect a Crystal Oscillator

Figure 10.2 shows how to connect a crystal oscillator. When the oscillator recommended by the company is used (the detail information is separately provided), any external component such as a load capacitor and a dumping resistor is not necessarily required for oscillation in general. However, proper operation should be verified under actual conditions prior to use.

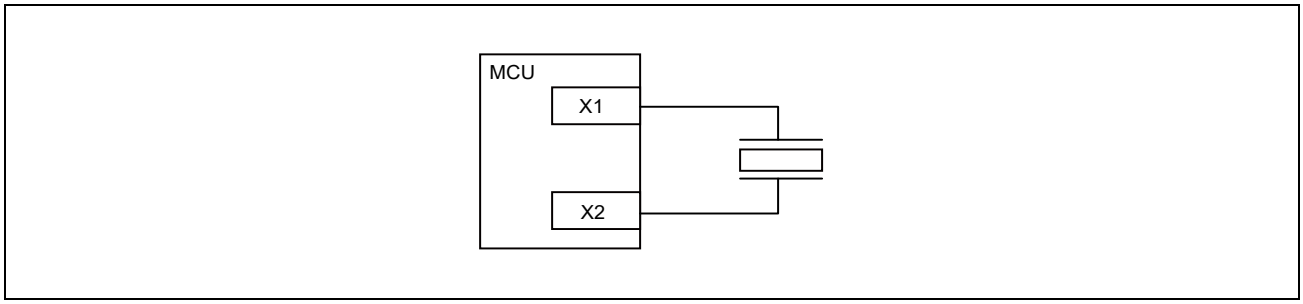


Figure 10.2 Connection Example of Crystal Oscillator

10.2 Register

10.2.1 List of Registers

Table 10.4 shows the list of registers.

Table 10.4 Registers Related to the Clock Controller

Module Name	Register Name	Symbol	Address	Access Protection
SYS	PLL0 status register	PLL0CLKS	FFF8 8004$_H$	
SYS	PLL0 control register 1	PLL0CLKC1	FFF8 8200$_H$	PROT1PHCMD
SYS	Clock 0 selection control register	CKSC0CTL	FFF8 9000$_H$	PROT1PHCMD
SYS	Clock 0 selection active register	CKSC0ACT	FFF8 9008$_H$	
SYS	Clock 0 division register	CLKD0DIV	FFF8 8800$_H$	PROT1PHCMD
SYS	Clock 0 division status register	CLKD0STAT	FFF8 8804$_H$	
SYS	Clock 1 selection control register	CKSC1CTL	FFF8 9040$_H$	PROT1PHCMD
SYS	Clock 1 selection active register	CKSC1ACT	FFF8 9048$_H$	
SYS	Protect 1 command register	PROT1PHCMD	FFF8 B000$_H$	
SYS	Protect 1 status register	PROT1PS	FFF8 B004$_H$	

10.2.2 PLL0CLKS – PLL0 Status Register

This register indicates the state of the PLL0 clock effective, active, or stable.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 8004$_H$

Value after reset: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKACT	CLKSTAB	CLKEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0 *1	0 *1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. Depends on the read timing after the CPU starts operation.

Table 10.5 PLL0CLKS Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0.
2	CLKACT	PLL0 Clock Source State 0: PLL0 clock source is inactive. 1: PLL0 clock source is active.
1	CLKSTAB	PLL0 Clock Stable State 0: PLL0 is unstable. 1: PLL0 is stable.
0	CLKEN	PLL0 Clock Operating State 0: PLL0 is stopped. 1: PLL0 is operating.

10.2.3 PLL0CLKC1 ─ PLL0 Control Register 1

This register controls the operation of the spread spectrum clock generator (SSCG) for PLL0.

This register can be set when PLL0 is running.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 8200$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SSMODE1	—	SELMFREQ[4:0]					SELMPERCENT[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 10.6 PLL0CLKC1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 10	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
9	SSMODE1	SSCG Control 0: Modulation disabled 1: Modulation enabled
8	—	Reserved This bit is always read as 0. When writing to this bit, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

Table 10.6 PLL0CLKC1 Register Contents (2/2)

Bit Position	Bit Name	Function
7 to 3	SELMFREQ [4:0]	SSCG Cycle Setting The modulation cycle of SSCG can be selected by setting these bits. Selectable settings are as follows: 10000_B : 80.65 kHz 10001_B : 75.76 kHz 10010_B : 69.44 kHz 10011_B : 65.79 kHz 10100_B : 62.50 kHz 10101_B : 59.52 kHz 10110_B : 58.14 kHz 10111_B : 50.00 kHz 11000_B : 41.67 kHz 11001_B : 39.68 kHz 11010_B : 37.31 kHz 11011_B : 33.33 kHz 11100_B : 30.12 kHz 11101_B : 25.00 kHz 11110_B : 20.00 kHz When SSMODE1 is set to 1, do not set the values other than the above.
2 to 0	SELMPERCENT[2:0]	SSCG Modulation Range Setting The modulation range of SSCG can be selected by setting these bits. Selectable settings are as follows: 100_B: – 5.0% When SSMODE1 is set to 1, do not set the values other than the above.

When changing the modulation depth of PLL0, turn off the SSCG (PLL0CLKC1.SSMODE1 = 0) and then wait for at least 1.6 ms.

10.2.4 CKSC0CTL ─ Clock 0 Selection Control Register

This register is used for selection of a clock source for the frequency divider 0A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 9000$_H$

Value after reset: 0000 0020$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CKSC[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 10.7 CKSC0CTL Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
5 to 0	CKSC[5:0]	These bits select a clock source for the frequency divider 0A. 100000_B: Selects the PLL1 clock. 100011_B: Selects the PLL0 clock. Setting other than the above is prohibited.

10.2.5 CKSC0ACT ─ Clock 0 Selection Active Register

This register indicates the state of a clock source for the frequency divider 0A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 9008_H

Value after reset: $0000\ 0020_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLKACT[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 10.8 CKSC0ACT Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0.
5 to 0	CLKACT[5:0]	These bits indicate that switching of a clock source for divider 0A is completed when their values are the same as those specified in the CKSC[5:0] bits in the CKSC0CTL register.

10.2.6 CLKD0DIV ─ Clock 0 Division Register

This register specifies a frequency division ratio of divider 0A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 8800$_H$

Value after reset: 0000 0004$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKD0DIV[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 10.9 CLKD0DIV Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	CLKD0DIV[2:0]	These bits select a frequency division ratio of divider 0A. 001_B: No division 010_B: Divided by 2 100_B: Divided by 4 Setting other than the above is prohibited.

10.2.7 CLKD0STAT ─ Clock 0 Division Status Register

This register indicates a clock state of the frequency divider 0A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 8804$_H$

Value after reset: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKD0 SYNC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 10.10 CLKD0STAT Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0.
0	CLKD0SYNC	Divider synchronization status 0: Divider 0A is switching to the division ratio specified in CLKD0DIV. 1: Divider 0A is running at the division ratio specified in CLKD0DIV.

10.2.8 CKSC1CTL ─ Clock 1 Selection Control Register

This register is used for selection of an output clock for the frequency divider 1A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 9040$_H$

Value after reset: 0000 0021$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CKSC[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 10.11 CKSC1CTL Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
5 to 0	CKSC[5:0]	These bits select an output clock for the frequency divider 1A. 100001_B: Selects 1/4 clock of PLL1. 100010_B: Selects 1/2 clock of PLL1. 100100_B: Selects 1/1 clock of PLL1. Setting other than the above is prohibited.

10.2.9 CKSC1ACT ─ Clock 1 Selection Active Register

This register indicate a clock state of the frequency divider 1A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 9048$_H$

Value after reset: 0000 0021$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLKACT[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 10.12 CKSC1ACT Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0.
5 to 0	CLKACT[5:0]	These bits indicate that switching of an output clock for the frequency divider 1A is completed when their values are the same as those specified in the CKSC[5:0] bits in the CKSC1CTL register. 100001_B: 1/4 x PLL1 100010_B: 1/2 x PLL1 100100_B: 1/1 x PLL1

10.2.10 PROT1PHCMD ─ Protect 1 Command Register

PROT1PHCMD is a protection command register to start the protection unlocking sequence necessary for access to write-protected registers.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 B000$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PCMD[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 10.13 PROT1PHCMD Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0.
7 to 0	PCMD[7:0]	Write protection command register

This register is used in protecting against inadvertent access to write-protected registers, i.e. registers to which writing raises the possibility of serious effects on application systems, such as programs crashing and the like.

Writing to the write-protected registers is only possible with the protection unlocking sequence described below.

Step 1. Write the fixed value (0000 00A5$_H$) to the PROT1PHCMD register.

Step 2. Write the new setting to the write-protected register. Write the value after a reset to the reserved bits.

Step 3. Write the bitwise inverse of the setting value to the same register as in step 2. Write the inverse of the value after a reset to the reserved bits.

Step 4. Write the new setting to the same register as in step 2. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the PROT1PS.PROTERR bit is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the PROT1PS.PROTERR bit is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*[1], writing to the protected register fails and the PROT1PS.PROTERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. PROT1PHCMD applies to the registers allocated to the addresses in the range from FFF8 8004_H to FFF8 $B004_H$. For the register names, register symbols, and module names, see the list of registers in **Table 8.1**, **Table 10.4**, **Table 29.122** to **Table 29.124**. and **Table 35.10**.

10.2.11 PROT1PS ─ Protect 1 Status Register

PROT1PS is a status register of the protection unlocking sequence. This register indicates whether an error occurs or not in access to write-protected registers.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 B004$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PROTERR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 10.14 PROT1PS Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0.
0	PROTERR	Protection Error Flag 0: Protection error does not occur. 1: Protection error occurs.

Operating conditions of the PROTERR bit

[Setting Condition]

Access to a write-protected register without executing the step of the protection unlocking sequence that involves the PROT1PHCMD register.

[Clearing Condition]

Writing of 0000 00A5$_H$ to the PROT1PHCMD register (step 1 in the protection unlocking sequence).

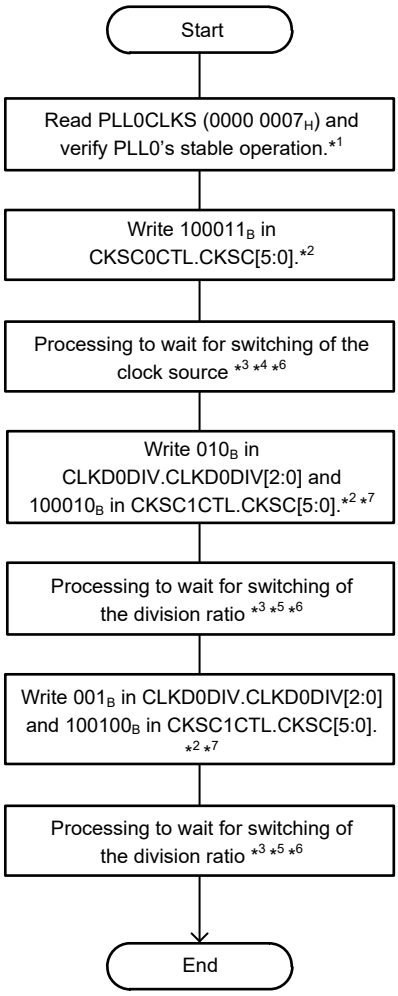
10.3 Function

10.3.1 Operation When the Divide Function is Used

Follow the procedure given below to switch the clock signals.

1. When a user program is running after release from the reset state, PLL0 and PLL1 will be oscillating and PLL1 will be in use as the internal operating clock. Also, the division ratio of divider 0A is set to 1/4.
   Read the PLL0CLKS register and verify that its value is 07_H, indicating that PLL0 operation is stable.
2. Write 23_H to the CKSC0CTL.CKSC[5:0] bits to select PLL0 as the clock source.
3. When switching the clock source, wait for at least 140 or 105 cycles of the CPU clock in the respective cases of products with CPUs having maximum operating frequencies of 320 MHz or 240 MHz. After that, read CKSC0ACT and verify that the value of the CKSC0ACT.CLKACT[5:0] bits is 23_H.
4. Write 010_B to the CLKD0DIV.CLKD0DIV[2:0] bits to select 1/2 as the division ratio for the divider.
5. For switching of the division ratio, wait for at least 140 or 105 cycles of the CPU clock in the respective cases of products with CPUs having maximum operating frequencies of 320 MHz or 240 MHz. Then read CLKD0STAT and verify that the value of the CLKD0SYNC is 1.
6. Write 001_B to the CLKD0DIV.CLKD0DIV[2:0] bits to select 1/1 as the division ratio for the divider.
7. For switching of the division ratio, wait for at least 140 or 105 cycles of the CPU clock in the respective cases of products with CPUs having maximum operating frequencies of 320 MHz or 240 MHz. Then read CLKD0STAT and verify that the value of the CLKD0SYNC is 1.

Change the division ratio of the divider 1A of the PLL1 side by following steps 4 to 7 above.



[Step 1]
The clock source for the SSCG clock domain is changed from PLL1 to PLL0.

[Step 2]
Division ratio of clock source PLL0 is changed from 1/4 to 1/2.
Division ratio of clock source PLL1 is changed from 1/4 to 1/2.

[Step 3]
Division ratio of clock source PLL0 is changed from 1/2 to 1/1.
Division ratio of clock source PLL1 is changed from 1/2 to 1/1.

Note 1. After release from the reset state, verify that PLL0 operation is stable before executing the sequence to shift the clock gear up. PLL0CLKS = 07_H indicates that PLL0 is stable.

Note 2. The CKSC0CTL and CLKD0DIV registers are protected by the PROT1PHCMD register. Execute the protection unlocking sequence (see the description of the protect 1 command register) when setting these registers.

Note 3. This processing is required because actual switching of the clock source or division ratio from the previous register settings takes time. The processing must produce periods of waiting of at least 140 or 105 cycles of the CPU clock in the respective cases of products with CPUs having maximum operating frequencies of 320 or 240 MHz.

Note 4. Verify that the value of the CKSC0ACT.CLKACT[5:0] bits is the same as that specified by the CKSC0CTL.CKSC[5:0] bits after switching the source clock for frequency divider 0A.
Similarly, verify that the value of the CKSC1ACT.CLKACT[5:0] bits is the same as that specified by the CKSC1CTL.CKSC[5:0] bits after switching the source clock for frequency divider 1A.

Note 5. Verify that the CLKD0STAT.CLKD0SYNC bit is 1 after changing the division ratio of divider 0A.

Note 6. Switching of clock sources or division ratios produces transient voltage fluctuations which last for a certain time, so measures such as processing to add more cycles of waiting may be required. Whether or not this is the case will depend on evaluation of the actual environment where the LSI chip is used.

Note 7. In the procedure described above, the division ratios of both PLL0 and PLL1 are set at the same time. When minimizing the transient voltage fluctuations during this procedure is crucial, separate the register settings and subsequent periods of waiting for PLL0 and PLL1 to achieve this (which also prolongs the procedure).

Figure 10.3 Example of Sequence for Shifting the Clock Gear Up

10.4 Notes

Avoid using commands that employ the computing unit, such as FPU operations, while the division ratio is being changed. Increased current fluctuation may make the operation unstable.

10.4.1 Board Design Notes

As shown in **Figure 10.4**, do not cross any other signal lines over the signal lines to the X1 and X2 pins. Induction may inhibit proper oscillation.

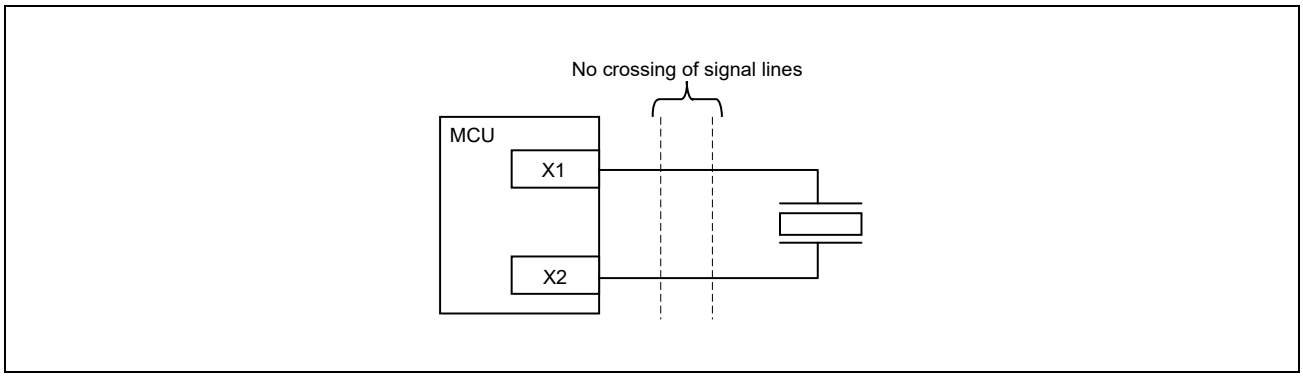


Figure 10.4 Board Design Notes

Section 11 Clocked Serial Interface H (CSIH)

This section contains a generic description of the Clocked Serial Interface H (CSIH).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of CSIH.

11.1 Features of RH850/C1M-A CSIH

11.1.1 Units

This LSI has the following number of CSIH units.

CSIH Each unit has one channel interface.

Table 11.1 Number of Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	3	3
Name	CSIHn (n = 0 to 2)	CSIHn (n = 0 to 2)

Table 11.2 Indices

Index	Meaning
n	Throughout this section, the individual CSIH units are identified by the index “n” (n = 0 to 2); for example, CSIHnCTL0 is the CSIHn control register 0.
x	CSIHn has up to 4 chip select signals. Throughout this section, the individual chip select signals are identified by the index “x”: that is, CSx denotes a non-specified chip select signal.
y	A variable used for explanation is identified by the index “y”; for example, CSIHnBRSy is a non-specified baud rate setting register of CSIHn.

The number of chip select signals for each channel of CSIH is given in the following table:

Table 11.3 Number of Chip Select Signals

	Number of Chip Select Signals	
Unit Name	RH850/C1M-A2	RH850/C1M-A1
CSIH0	CSx (x = 0 to 3)	CSx (x = 0 to 3)
CSIH1	CSx (x = 0 to 3)	CSx (x = 0 to 3)
CSIH2	CSx (x = 0 to 3)	CSx (x = 3)*1

Note 1. The number of chip select signals provided for CSIH2 in RH850/C1M-A1 is one.

11.1.2 Register Base Addresses

CSIH base addresses are listed in the following table.

CSIH register addresses are given as offsets from the base addresses in general.

Table 11.4 Register Base Address

Base Address Name	Base Address
<CSIH0_base>	FFD8 0000_H
<CSIH1_base>	FFD8 2000_H
<CSIH2_base>	FFD8 4000_H

11.1.3 Clock Supply

CSIH interrupt requests are listed in the following table.

Table 11.5 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
CSIHn	PCLK	CLK_HSB (high-speed peripheral clock)

11.1.4 Interrupt Requests

CSIH interrupt requests are listed in the following table.

Table 11.6 Interrupt Requests

Unit Interrupt Name	Outline	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
CSIH0						
INTCSIHTIC	Communication status interrupt	301	111	—	111	—
INTCSIHTIR	Receive status interrupt	302	112	—	112	—
INTCSIHTIRE	Communication error interrupt	303	—	—	—	—
INTCSIHTIJC	Job completion interrupt	304	113	—	113	—
CSIH1						
INTCSIHTIC	Communication status interrupt	305	114	—	114	—
INTCSIHTIR	Receive status interrupt	306	115	—	115	—
INTCSIHTIRE	Communication error interrupt	307	—	—	—	—
INTCSIHTIJC	Job completion interrupt	308	116	—	116	—
CSIH2						
INTCSIHTIC	Communication status interrupt	309	117	—	117	—
INTCSIHTIR	Receive status interrupt	310	118	—	118	—
INTCSIHTIRE	Communication error interrupt	311	—	—	—	—
INTCSIHTIJC	Job completion interrupt	312	119	—	119	—

—: No number is assigned.

Note 1. 1st: Primary channel, 2nd: Secondary channel

11.1.5 Reset Sources

The CSIHn and the CSIHn registers are initialized by the following reset signal.

Table 11.7 Reset Sources

Unit Name	Reset Sources
CSIHn	All reset sources

11.1.6 External Input/Output Signals

External input/output signals of CSIH are listed in the following table.

Table 11.8 External Input/Output Signals of RH850/C1M-A2 and RH850/C1M-A1

CSIHn Signal	Function	Alternative Port Pin Signal
CSIH0		
CSIHTSCK	Serial clock input signal	CSIH0SC
CSIHTSI	Serial data input signals	CSIH0SI
$\overline{\text{CSIHTSSI}}$	Slave select input signals	$\overline{\text{CSIH0SSI}}$
CSIHTRYI	Ready / Busy input signal	CSIH0RYI
CSIHTSO	Serial data output signals	CSIH0SO
CSIHTRYO	Ready / Busy output signal	CSIH0RYO
CSIHTCSS[3:0]	Chip select signals	CSIH0CSS[3:0]
CSIH1		
CSIHTSCK	Serial clock input signal	CSIH1SC
CSIHTSI	Serial data input signals	CSIH1SI
$\overline{\text{CSIHTSSI}}$	Slave select input signals	$\overline{\text{CSIH1SSI}}$
CSIHTRYI	Ready / Busy input signal	CSIH1RYI
CSIHTSO	Serial data output signals	CSIH1SO
CSIHTRYO	Ready / Busy output signal	CSIH1RYO
CSIHTCSS[3:0]	Chip select signals	CSIH1CSS[3:0]
CSIH2		
CSIHTSCK	Serial clock input signal	CSIH2SC
CSIHTSI	Serial data input signals	CSIH2SI
$\overline{\text{CSIHTSSI}}$	Slave select input signals	$\overline{\text{CSIH2SSI}}$
CSIHTRYI	Ready / Busy input signal	CSIH2RYI
CSIHTSO	Serial data output signals	CSIH2SO
CSIHTRYO	Ready / Busy output signal	CSIH2RYO
CSIHTCSS[3:0]	Chip select signals	CSIH2CSS[3:0]*1

Note 1. The number of chip select signals provided for CSIH2 in RH850/C1M-A1 is one for CSIH2CSS[3].

11.1.7 Data Consistency Check

The following table lists data consistency checking for the port on which CSIHnSO pin functions are multiplexed. See **Section 11.4.18, Error Detection** for details on data consistency checking.

Table 11.9 Data Consistency Checking and Port Pins

CSIHn Signal	Port Pin Name	Alternative Function
CSIH0		
CSIHTSO	P4_8	ALT_OUT6
CSIH1		
CSIHTSO	P4_1	ALT_OUT7
CSIH2		
CSIHTSO	P7_2	ALT_OUT7

11.2 Overview

11.2.1 Functional Overview

- Three-wire serial synchronous data transfer
- Master mode and slave mode selectable
- Multiple slaves configuration plus RCB (Recessive Configuration for Broadcasting) thanks to 4 configurable chip select output signals
- Slave select input signal ($\overline{\text{CSIHTSSI}}$) included
- Built-in baud rate generator
- Transfer clock frequency adjustable in master mode; determined by input clock in slave mode
- Maximum transfer clock frequency:
  - in master mode: PCLK/8
  - in slave mode: PCLK/16
- Phase of clock and data selectable
- Data transfer with MSB or LSB first selectable
- Transfer data length selectable from 2 to 16 bits in 1-bit units
- EDL (Extended Data Length) function for transferring data with more than 16 bits included
- Three selectable transfer modes:
  - transmit-only mode
  - receive-only mode
  - transmit/receive mode
- Handshake function included
- Error detection (data consistency check, parity, time-out, overflow, and overrun) included
- Support of job concept
- 128 words I/O buffer memory
- Selectable direct access mode and memory mode (FIFO, dual buffer, and transmit-only buffer)
- Four different interrupt request signals (INTCSIHTIC, INTCSIHTIR, INTCSIHTIRE, INTCSIHTIJC)
- LBM (Loop Back Mode) function for self-test included
- CPU-controlled high-priority communication function
- Enforced chip select idle setting
- RCB (Recessive Configuration for Broadcasting) bit included
- Job enable control bit for AUTOSAR included

11.2.2 Functional Overview Description

The Clocked Serial Interface uses three signals for communication:

- Transmission clock CSIHTSCK (output in master mode, input in slave mode)
- Data output signal CSIHTSO
- Data input signal CSIHTSI

Additional signals are available for external control and monitoring.

- $\overline{\text{CSIHTSSI}}$: Slave select input signal
- CSIHTRYO: Ready/busy output signal (handshake signal)
- CSIHTRYI: Ready/busy input signal (handshake signal)
- CSIHTCSS[3:0]: Chip select signals

Data transmission is bit-wise and serial, and synchronous to the transmission clock.

The following table shows the most important registers for setting up the CSIH.

Table 11.10 Main Registers of CSIH

Register	Function
CSIHnCTL0	Enables or disables transmission clock, and permits or prohibits data transmission and data reception. Defines end-of-job behavior and enables or disables buffering (bypass of the buffer).
CSIHnCTL1	Controls options like interrupt timing, extended data length, job feature, data consistency check, loop-back mode, handshake, etc.
CSIHnCTL2	Selects master or slave mode, and the transfer clock frequency of the on-chip baud rate generator (BRG) in master mode.
CSIHnBRSy	Registers to configure the transfer clock frequency for each chip select signal
CSIHnMCTL0	Selects memory mode and specifies the time-out value
CSIHnMCTL1	Controls the memory in FIFO mode
CSIHnMCTL2	Controls the memory in dual buffer mode or transmit-only buffer mode
CSIHnCFGx	Registers to configure the communication protocol for each chip select signal

11.2.3 Block Diagram

The block diagram shows the main components of the CSIH.

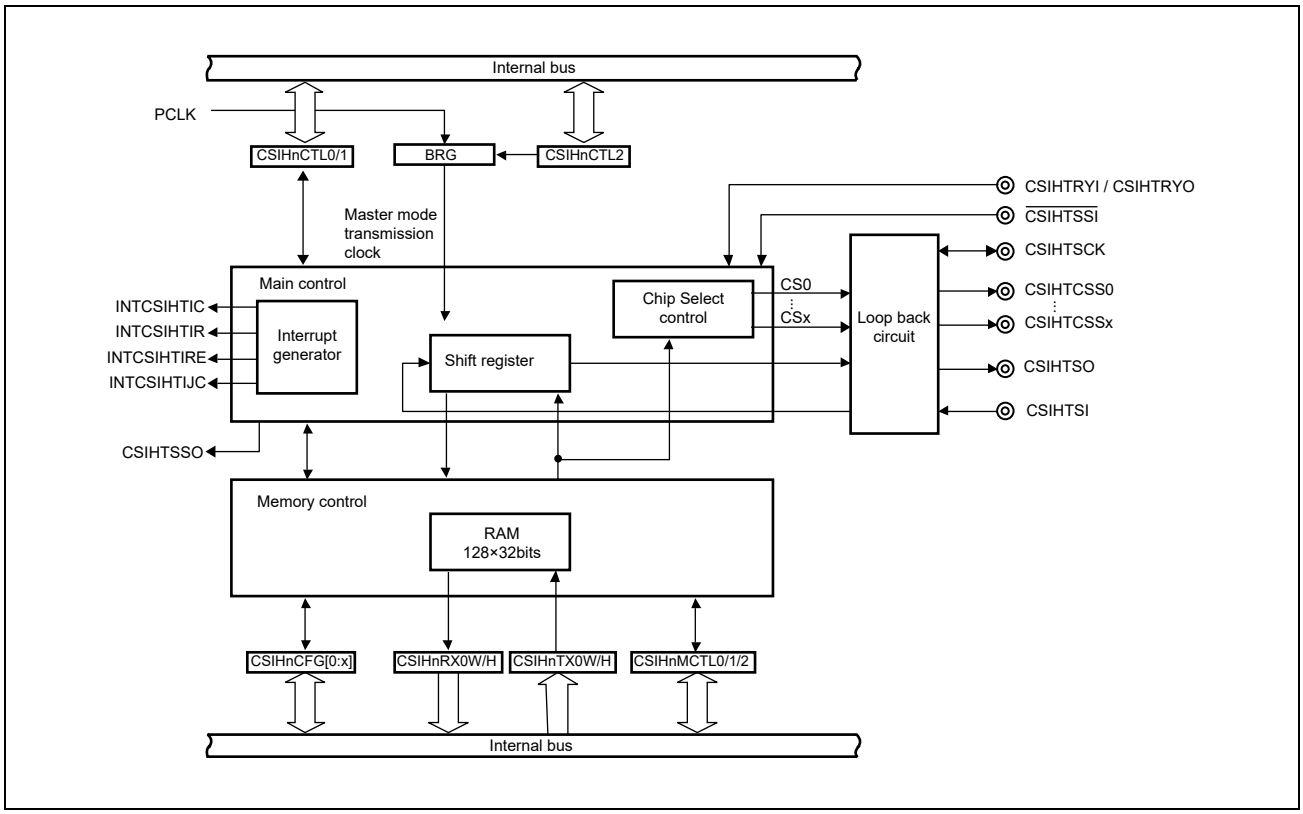


Figure 11.1 CSIH Block Diagram

In master mode, the transmission clock CSIHTSCK is generated by the built-in baud rate generator (BRG). In slave mode, the transmission clock is provided from an external source.

The built-in memory can be configured as FIFO, dual buffer (separate transmit and receive buffers), or transmit-only buffer. It can also be bypassed for data transmission and reception without buffering.

The loop back circuit disconnects the CSIH completely from the ports and handles internal self-test.

NOTE

This section describes the following modes:

- The “operating mode” separates between master and slave mode. In this context, only a master can control and communicate with several slaves (for details, see **Section 11.4.7, Operating Modes (Master/Slave)**).
- The “job mode” is related to the AUTOSAR job concept (for details see **Section 11.4.9.3, Job Concept**).
- The “memory mode” takes the various configurations of the associated buffer memory into account (for details see **Section 11.4.12, CSIH Buffer Memory**).
- The “data transfer mode” specifies the kind of the communication – transmit-only, receive only, or transmit/receive (for details see **Section 11.4.13, Data Transfer Modes**).

11.3 Registers

11.3.1 List of Registers

CSIH registers are listed in the following table.

For information on <CSIHn_base>, see **Section 11.1.2, Register Base Addresses.**

Table 11.11 List of Registers

Module Name	Register Name	Symbol	Address
CSIHn	CSIHn control register 0	CSIHnCTL0	<CSIHn_base> + 0000_H
CSIHn	CSIHn control register 1	CSIHnCTL1	<CSIHn_base> + 0010_H
CSIHn	CSIHn control register 2	CSIHnCTL2	<CSIHn_base> + 0014_H
CSIHn	CSIHn status register 0	CSIHnSTR0	<CSIHn_base> + 0004_H
CSIHn	CSIHn status clear register 0	CSIHnSTCR0	<CSIHn_base> + 0008_H
CSIHn	CSIHn memory control register 0	CSIHnMCTL0	<CSIHn_base> + 1040_H
CSIHn	CSIHn memory control register 1	CSIHnMCTL1	<CSIHn_base> + 1000_H
CSIHn	CSIHn memory control register 2	CSIHnMCTL2	<CSIHn_base> + 1004_H
CSIHn	CSIHn memory read/write pointer register 0	CSIHnMRWP0	<CSIHn_base> + 1018_H
CSIHn	CSIHn configuration register 0	CSIHnCFG0	<CSIHn_base> + 1044_H
CSIHn	CSIHn configuration register 1	CSIHnCFG1	<CSIHn_base> + 1048_H
CSIHn	CSIHn configuration register 2	CSIHnCFG2	<CSIHn_base> + $104C_H$
CSIHn	CSIHn configuration register 3	CSIHnCFG3	<CSIHn_base> + 1050_H
CSIHn	CSIHn transmit data register 0 for word access	CSIHnTX0W	<CSIHn_base> + 1008_H
CSIHn	CSIHn transmit data register 0 for half-word access	CSIHnTX0H	<CSIHn_base> + $100C_H$
CSIHn	CSIHn receive data register 0 for word access	CSIHnRX0W	<CSIHn_base> + 1010_H
CSIHn	CSIHn receive data register 0 for half-word access	CSIHnRX0H	<CSIHn_base> + 1014_H
CSIHn	CSIHn baud rate setting register 0	CSIHnBRS0	<CSIHn_base> + 1068_H
CSIHn	CSIHn baud rate setting register 1	CSIHnBRS1	<CSIHn_base> + $106C_H$
CSIHn	CSIHn baud rate setting register 2	CSIHnBRS2	<CSIHn_base> + 1070_H
CSIHn	CSIHn baud rate setting register 3	CSIHnBRS3	<CSIHn_base> + 1074_H

11.3.2 CSIHnCTL0 — CSIHn Control Register 0

This register controls the operation clock and enables/disables transmission/reception and the memory part for transmission and/or reception. It forces the stop of communication at the end of the current job.

Access: Readable/writable in 8- or 1-bit units.

Address: <CSIHn_base> + 0000_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CSIHnPWR	CSIHnTXE	CSIHnRXE	—	—	—	CSIHnJOBE	CSIHnMBS
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

Table 11.12 CSIHnCTL0 Register Contents

Bit Position	Bit Name	Function
7	CSIHnPWR	Controls the operational clock. 0: Stops the operational clock. 1: Supplies the operational clock. Clearing CSIHnPWR to 0 resets the internal circuits, stops operation, and sets CSIH to standby state. No clock is provided to internal circuits. If CSIHnPWR is cleared during communication, ongoing communication is immediately aborted. In this case, communication setting must be started over.
6	CSIHnTXE	Enables or disables transmission. 0: Disables transmission. 1: Enables transmission.
5	CSIHnRXE	Enables or disables reception. 0: Disables reception. 1: Enables reception.
4 to 2	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
1	CSIHnJOBE	Stops communication at the end of the current job. (Communication ends if data is written to the transmission buffer when CSIHnTX0W.CSIHnEOJ = 1 (job completion)). 0: Communication stop is not requested. 1: Stops communication. This bit can be used to abort an ongoing job. This bit is cleared to 0 automatically. If this bit is set to 1, the read value is always 0. In FIFO mode, the pointer must be cleared by setting CSIHnSTCR0.CSIHnPCT = 1 before the next communication is started.
0	CSIHnMBS	Bypasses the memory for transmission and/or reception data. 0: Memory mode CSIH memory is used for transmission and/or reception data. 1: Direct access mode CSIH memory is bypassed.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.3 CSIHnCTL1 — CSIHn Control Register 1

This register specifies the interrupt timing and the interrupt delay mode. It also enables or disables extended data length control, data consistency check, loop-back mode, handshake functionality, and job mode. It selects the active level of the output of each chip select signal and the behavior of the chip select signals after the transfer of the final data.

Access: Readable/writable in 32-bit units.

Address: <CSIHn_base> + 0010_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	CSIHn PHE	CSIHn CKR	CSIHn SLIT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CSIHn CSL3	CSIHn CSL2	CSIHn CSL1	CSIHn CSL0	CSIHn EDLE	CSIHn JE	CSIHn DCS	CSIHn CSRI	CSIHn LBM	CSIHn SIT	CSIHn HSE	CSIHn SSE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.13 CSIHnCTL1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 19	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
18	CSIHnPHE	Sets the CPU-controlled priority-based communication function. 0: The CPU-controlled high-priority communication function is disabled. 1: The CPU-controlled high-priority communication function is enabled. To enable the CPU-controlled high-priority communication function, set this bit to 1 and set CSIHnJE = 1. This bit can only be set in transmit-only buffer mode.
17	CSIHnCKR	CSIHTSCK clock inversion function 0: The default level of CSIHTSCK is high 1: The default level of CSIHTSCK is low For details, see **Section 11.3.11, CSIHnCFGx — CSIHn Configuration Register x**.
16	CSIHnSLIT	Selects the timing of interrupt INTCSIHTIC. 0: Normal interrupt timing (interrupt is generated after the transfer) 1: As soon as the contents of the CSIHnTX0W/H register are transferred to the shift register, an interrupt is generated (this function is activated only in direct access memory mode/transmit-only buffer mode). For details, see **Section 11.4.3, INTCSIHTIC (Communication Status Interrupt)**.
15 to 12	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.

Table 11.13 CSIHnCTL1 Register Contents (2/2)

Bit Position	Bit Name	Function
11 to 8	CSIHnCSLx	Selects the active output level of chip select signal x (CSIHTCSSx) (x = 0 to 3). 0: Chip select is active low. 1: Chip select is active high. For details, see **Section 11.4.9, Chip Selection (CS) Features**.
7	CSIHnEDLE	Enables/disables extended data length (EDL) mode. 0: Disables extended data length mode. 1: Enables extended data length mode. For details, see **Section 11.4.14.2, Data length greater than 16 bits**.
6	CSIHnJE	Enables/disables job mode. 0: Disables job mode. 1: Enables job mode. For details, see **Section 11.4.9.3, Job Concept**. The CSIHnCTL0.CSIHnJOBE, CSIHnTX0W.CSIHnEOJ, and CSIHnTX0W.CSIHnCIRE bits are enabled only when CSIHnJE = 1. Setting this bit in slave mode is prohibited. In addition, to enable the CPU-controlled high-priority communication function, set CSIHnPHE = 1 and this bit to 1.
5	CSIHnDCS	Enables/disables data consistency check. 0: Disables data consistency check. 1: Enables data consistency check. For details, see **Section 11.4.18.1, Data consistency check**.
4	CSIHnCSRI	Defines chip select signal behavior after last data transfer. 0: Chip select signal holds the active level. 1: Chip select signal returns to the inactive level. The last data is determined at the interrupt timing in direct access mode or FIFO mode. When CSIHnCTL1.CSIHnSLIT = 1, the last data is determined in direct access mode.
3	CSIHnLBM	Controls loop-back mode (LBM). 0: Deactivates loop-back mode. 1: Activates loop-back mode. For details, see **Section 11.4.19, Loop-back Mode**.
2	CSIHnSIT	Selects interrupt delay mode. 0: No delay is generated. 1: Delay of half a clock cycle is generated for all interrupts. This bit is only valid in master mode. In slave mode, no delay is generated. For details, see **Section 11.4.2, Interrupt Delay**.
1	CSIHnHSE	Enables/disables the handshake function. 0: Disables the handshake function. 1: Enables the handshake function. For details refer to **Section 11.4.17, Handshake Function**.
0	CSIHnSSE	Enables/disables the slave select function. 0: Input signal $\overline{\text{CSIHTSSI}}$ is disabled. 1: Input signal $\overline{\text{CSIHTSSI}}$ is enabled. If the slave select function is not used, this bit must be set to 0 (see also **Section 11.4.8, Master/Slave Connections**).

Details about CSIHnCTL1.CSIHnSSE are shown in the following tables.

Table 11.14 Operation of the Slave Select Function During Reception

CSIHnCTL0.CSIHnRXE	CSIHnCTL1.CSIHnSSE	$\overline{\text{CSIHTSSI}}$	Receive Operation
0	—	—	Reception is prohibited
1	0	—	Possible
1	1	0	Possible
1	1	1	Disabled

Table 11.15 Operation of the Slave Select Function During Transmission

CSIHnCTL0.CSIHnTXE	CSIHnCTL1.CSIHnSSE	$\overline{\text{CSIHTSSI}}$	Transmit Operation
0	—	—	Transmission is prohibited
1	0	—	Possible
1	1	0	Possible
1	1	1	Disabled

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.4 CSIHnCTL2 — CSIHn Control Register 2

This register selects operating mode and the basic clock value, and specifies the transfer clock frequency.

For details see **Section 11.4.11, Transmission clock selection**.

Access: Readable/writable in 16-bit units.

Address: <CSIHn_base> + 0014_H

Value after reset: $E000_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnPRS[2:0]			—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.16 CSIHnCTL2 Register Contents

Bit Position	Bit Name	Function
15 to 13	CSIHnPRS[2:0]	These bits select the operation mode and the reference clock value.
12 to 0	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.

CSIHnPRS2	CSIHnPRS1	CSIHnPRS0	Selection of Reference Clock (PRSOUT)
0	0	0	PCLK (Master mode)
0	0	1	PCLK/2 (Master mode)
0	1	0	PCLK/4 (Master mode)
0	1	1	PCLK/8 (Master mode)
1	0	0	PCLK/16 (Master mode)
1	0	1	PCLK/32 (Master mode)
1	1	0	PCLK/64 (Master mode)
1	1	1	External clock via CSIHTSCK (in) (Slave mode)

In master mode, the following bits are used to set the transfer clock frequency:

CSIHnCTL2.CSIHnPRS[2:0],

CSIHnCFGx.CSIHnBRSSx[1:0],

CSIHnBRSy.CSIHnBRS[11:0]

In addition, any of the four different transfer clock frequency settings that are specified by the CSIHnBRSy.CSIHnBRS[11:0] bits is selected according to the chip select signal. To select the transfer clock frequency setting for each chip select signal, use the CSIHnCFGx.CSIHnBRSSx[1:0] bits.

The following table shows the relationship between CSIHnCFGx.CSIHnBRSSx[1:0] and CSIHnBRSy.CSIHnBRS[11:0].

CSIHnCFGx.CSIHnBRSSx[1:0]	Baud Rate Setting Bit to be Selected
00	CSIHnBRS0.CSIHnBRS[11:0]
01	CSIHnBRS1.CSIHnBRS[11:0]
10	CSIHnBRS2.CSIHnBRS[11:0]
11	CSIHnBRS3.CSIHnBRS[11:0]

The following table shows the relationship between the transfer clock frequency and the transfer clock frequency setting (CSIHnBRS[11:0]) selected by the CSIHnBRSSx[1:0] bits when the bit value of the CSIHnPRS[2:0] bits is α.

CSIHnBRS[11:0]	Transfer Clock Frequency
0	BRG stopped
1	PCLK / (2^{α} × 1 × 2)
2	PCLK / (2^{α} × 2 × 2)
3	PCLK / (2^{α} × 3 × 2)
4	PCLK / (2^{α} × 4 × 2)
:	:
4095	PCLK / (2^{α} × 4095× 2)

When a time-out error is used in slave mode, the clock selected by this setting is used. In slave mode, the CSIHnPRS[2:0] bits are set to 111_B. In this case, the prescaler has the same setting as when the CSIHnPRS[2:0] bits are set to 000_B. If you are using a time-out error, set the CSIHnBRSy.CSIHnBRS[11:0] bits to a value other than 000_H.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.5 CSIHnSTR0 — CSIHn Status Register 0

This register indicates the status of CSIH.

Access: Readable in 32-bit units.

Address: <CSIHn_base> + 0004_H

Value after reset: $0000\ 0010_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHnSRP[7:0]								CSIHnSPF[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHn TMOE	CSIHn OFE	—	—	—	—	—	CSIHn HPST	CSIHn TSF	—	CSIHn FLF	CSIHn EMF	CSIHn DCE	—	CSIHn PE	CSIHn OVE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.17 CSIHnSTR0 Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 24	CSIHnSRP[7:0]	Indicates the number of received data packets in FIFO mode. CSIHnSRP[7:0] / Description 00_H / Number of received data packets (0 to 128) : / 80_H / Other than the above / Undefined These bits are cleared by CSIHnSTCR0.CSIHnPCT. In direct access mode, dual buffer mode, or transmit-only buffer mode, this value is fixed to 00_H. In direct access mode, this bit is fixed to 0 because there is no pointer. In buffer mode, this bit is fixed to 0 because the number of data packets is managed by CSIHnMCTL2.CSIHnND[7:0].
23 to 16	CSIHnSPF[7:0]	Indicates the number of unsent data in FIFO mode. (The number of data written by the CPU is the number of sent data.) CSIHnSPF[7:0] / Description 00_H / Number of unsent data packets (0 to 128) : / 80_H / Other than the above / Undefined These bits are cleared by CSIHnSTCR0.CSIHnPCT. In direct access mode, dual buffer mode, or transmit-only buffer mode, this value is fixed to 00_H. In direct access mode, this bit is fixed to 0 because there is no pointer. In buffer mode, this bit is fixed to 0 because the number of data packets is managed by CSIHnMCTL2.CSIHnND[7:0].

Table 11.17 CSIHnSTR0 Register Contents (2/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>15</td><td>CSIHnTMOE</td><td>Time-out error flag in FIFO mode
Indicates whether a time-out error was detected in FIFO mode.
0: No time out error is detected.
1: A time out error is detected.
For details, see Section 11.4.18.3, Time-out error.
This bit is cleared by CSIHnSTCR0.CSIHnTMOEC.
When this bit is set to 1 by the detection of time-out error and cleared to 0 by CSIHnSTCR0.CSIHnTMOEC simultaneously, setting to 1 is prioritized.
This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.</td></tr>
<tr><td>14</td><td>CSIHnOFE</td><td>Overflow error flag in FIFO mode
Indicates whether an overflow error was detected in FIFO mode.
0: No overflow error is detected.
1: An overflow error is detected.
For details, see Section 11.4.18.4, Overflow error.
This bit is cleared by CSIHnSTCR0.CSIHnOFEC.
When this bit is set to 1 by the detection of overflow error and cleared to 0 by CSIHnSTCR0.CSIHnOFEC simultaneously, setting to 1 is prioritized.
This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.</td></tr>
<tr><td>13 to 9</td><td>Reserved</td><td>When read, the value after a reset is read.</td></tr>
<tr><td>8</td><td>CSIHnHPST</td><td>Communication priority indication flag
0: Indicates low-priority communication is in progress.
1: Indicates high-priority communication is in progress.
This bit always reads 0 if CPU-controlled high-priority communication is disabled (CSIHnCTL1.CSIHnPHE = 0).</td></tr>
<tr><td>7</td><td>CSIHnTSF</td><td>Transfer status flag
0: Idle state
1: Communication is in progress or being prepared.

The timing to set or clear this bit is as follows:
<table>
<tr><th rowspan="2">Master Mode</th><th colspan="2">Timing to Set</th><th rowspan="2">Timing to Clear</th></tr>
<tr><th>Direct Access Mode, FIFO Mode</th><th>Dual Buffer Mode, Transmit-Only Buffer Mode</th></tr>
<tr><td>Transmit-only mode</td><td rowspan="3">Data is written to a transmit register (CSIHnTX0W/CSIHnTX0H)</td><td rowspan="3">Bit CSIHnMCTL2.CSIHnBTST is set</td><td rowspan="3">Within a half clock of the last serial clock edge</td></tr>
<tr><td>Transmit/receive mode</td></tr>
<tr><td>Receive-only mode</td></tr>
</table>
<table>
<tr><th rowspan="2">Slave Mode</th><th colspan="2">Timing to Set</th><th rowspan="2">Timing to Clear</th></tr>
<tr><th>Direct Access Mode, FIFO Mode</th><th>Dual Buffer Mode, Transmit-Only Buffer Mode</th></tr>
<tr><td>Transmit-only mode</td><td rowspan="2">Data is written to a transmit register (CSIHnTX0W/CSIHnTX0H)</td><td rowspan="3">Bit CSIHnMCTL2.CSIHnBTST is set</td><td rowspan="3">Within a half clock of the last serial clock edge</td></tr>
<tr><td>Transmit/receive mode</td></tr>
<tr><td>Receive-only mode</td><td>Input timing of CSIHTSCK</td></tr>
</table></td></tr>
<tr><td>6</td><td>Reserved</td><td>When read, the value after a reset is read.</td></tr>
</table>

Table 11.17 CSIHnSTR0 Register Contents (3/3)

Bit Position	Bit Name	Function
5	CSIHnFLF	A flag indicating that the buffer is full in FIFO mode 0: FIFO buffer is not full. 1: FIFO buffer is full. This bit is cleared to 0 by CSIHnSTCR0.CSIHnPCT. The FIFO buffer might be filled with unsent data or received data.
4	CSIHnEMF	A flag indicating that the buffer is empty in FIFO mode 0: FIFO buffer is not empty. 1: FIFO buffer is empty. This bit is set to 1 by CSIHnSTCR0.CSIHnPCT. This bit is set to 1 when value of “CSIHnSTR0.CSIHnSRP[7:0] + CSIHnSTR0.CSIHnSPF[7:0]” matches to 00_H. The FIFO buffer might be filled with unsent data or received data.
3	CSIHnDCE	Data consistency check error flag 0: No data consistency error is detected. 1: Data consistency error is detected. This bit is cleared to 0 by writing 1 to CSIHnSTCR0.CSIHnDCEC. When this bit is set to 1 by the detection of data consistency check error and cleared to 0 by CSIHnSTCR0.CSIHnDCEC simultaneously, setting to 1 is prioritized. This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.
2	Reserved	When read, the value after a reset is read.
1	CSIHnPE	Parity error flag 0: No parity error is detected. 1: Parity error is detected. This bit is cleared by writing 1 to CSIHnSTCR0.CSIHnPEC. When this bit is set to 1 by the detection of parity error and cleared to 0 by CSIHnSTCR0.CSIHnPEC simultaneously, setting to 1 is prioritized. This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.
0	CSIHnOVE	Overrun error flag (Fixed to 0 in dual buffer mode) 0: No overrun error is detected. 1: Overrun error is detected. This bit is cleared to 0 by writing 1 to CSIHnSTCR0.CSIHnOVEC. When this bit is set to 1 by overrun error and cleared to 0 by CSIHnSTCR0.CSIHnOVEC simultaneously, setting to 1 is prioritized. This bit is Initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.

Table 11.18 Behavior in Memory Mode

Bit Name	Bit Position	Direct Access Mode	FIFO Mode	Transmit-Only Buffer Mode	Dual Buffer Mode
CSIHnSRP[7:0]	31 to 24	Fixed to 0	Number of received words	Fixed to 0	Fixed to 0
CSIHnSPF[7:0]	23 to 16	Fixed to 0	Number of unsent data	Fixed to 0	Fixed to 0
CSIHnTMOE	15	Fixed to 0	0: No error is detected. 1: An error is detected.	Fixed to 0	Fixed to 0
CSIHnOFE	14	Fixed to 0	0: No error is detected. 1: An error is detected.	Fixed to 0	Fixed to 0
CSIHnTSF	7	0: Idle state 1: Communication is in progress or being prepared			
CSIHnFLF	5	Fixed to 0	0: FIFO is not full 1: FIFO is full	Fixed to 0	Fixed to 0
CSIHnEMF	4	Fixed to 1	0: FIFO is not empty 1: FIFO is empty	Fixed to 1	Fixed to 1
CSIHnDCE	3	0: No error is detected. 1: An error is detected.			
CSIHnPE	1	0: No error is detected. 1: An error is detected.			
CSIHnOVE	0	0: No error is detected. 1: An error is detected.	0: No error is detected. 1: An error is detected.	0: No error is detected. 1: An error is detected.	Fixed to 0

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.6 CSIHnSTCR0 — CSIHn Status Clear Register 0

This register clears the status flags of the CSIHnSTR0 status register.

Access: Readable/writable in 16-bit units.

When read, the value 0000_H is always returned.

Address: <CSIHn_base> + 0008_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHn TMOEC	CSIHn OFEC	—	—	—	—	—	CSIHn PCT	—	—	—	—	CSIHn DCEC	—	CSIHn PEC	CSIHn OVEC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R/W	R	R	R	R	R/W	R	R/W	R/W

Table 11.19 CSIHnSTCR0 Register Contents

Bit Position	Bit Name	Function
15	CSIHnTMOEC	Controls the time-out error flag clear command. 0: No operation. The read value is always 0. 1: Clears the time out error flag (CSIHnSTR0.CSIHnTMOE).
14	CSIHnOFEC	Controls the overflow error flag clear command. 0: No operation. The read value is always 0. 1: Clears the overflow error flag (CSIHnSTR0.CSIHnOFE).
13 to 9	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
8	CSIHnPCT	Controls the FIFO pointer clear command. 0: No operation. The read value is always 0. 1: Clears the FIFO buffer pointers below (in FIFO mode, dual buffer mode, and transmit-only buffer mode) and the status bits. FIFO Buffer Pointer / Status Bit: CSIHnMRWP0.CSIHnTRWA[6:0] — CSIHnSTR0.CSIHnSPF[7:0] CSIHnMRWP0.CSIHnRRA[6:0] — CSIHnSTR0.CSIHnSRP[7:0] CSIHnMCTL2.CSIHnSOP[6:0] — CSIHnSTR0.CSIHnFLF CSIHnSTR0.CSIHnTSF Also, the CSIHnSTR0.CSIHnEMF bit is set to 1 (FIFO empty) (only in FIFO mode).
7 to 4	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
3	CSIHnDCEC	Controls the data consistency error flag clear command. 0: No operation. The read value is always 0. 1: Clears the data consistency error flag (CSIHnSTR0.CSIHnDCE).
2	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
1	CSIHnPEC	Controls the parity error flag clear command. 0: No operation. The read value is always 0. 1: Clears the parity error flag (CSIHnSTR0.CSIHnPE).
0	CSIHnOVEC	Controls the overrun error flag clear command. 0: No operation. The read value is always 0. 1: Clears the overrun error flag (CSIHnSTR0.CSIHnOVE).

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.7 CSIHnMCTL0 — CSIHn Memory Control Register 0

This register selects the memory mode and the time-out setting.

Access: Readable/writable in 16-bit units.

Address: <CSIHn_base> + 1040_H

Value after reset: $001F_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CSIHnMMS[1:0]		—	—	—	CSIHnTO[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 11.20 CSIHnMCTL0 Register Contents

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
9, 8	CSIHnMMS[1:0]	Selects the memory mode. CSIHn MMS1 / CSIHn MMS0 / Description 0 / 0 / FIFO mode 0 / 1 / Dual buffer mode 1 / 0 / Transmit-only buffer mode 1 / 1 / Prohibited After a change of the memory mode, the respective buffer pointers must be cleared by setting the CSIHnSTCR0.CSIHnPCT bit to 1. In direct access mode, the setting of these bits is ignored.
7 to 5	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
4 to 0	CSIHnTO[4:0]	Selects the time-out setting in FIFO mode. CSIHnTO[4:0] / Description 00000_B / No time-out is detected 00001_B / Time-out is (1 × 8 × BRG output clocks) 00010_B / Time-out is (2 × 8 × BRG output clocks) : 11111_B / Time-out is (31 × 8 × BRG output clocks) **CAUTION:** Changing the time-out setting is only permitted when CSIHnCTL0.CSIHnPWR = 0. Set the CSIHnTO[4:0] bits to 00000_B in direct access mode, dual buffer mode, or transmit-only buffer mode (except FIFO mode). For details about time-out detection, see also **Section 11.4.18.3, Time-out error**.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.8 CSIHnMCTL1 — CSIHn Memory Control Register 1

This register selects the conditions to generate the interrupt requests, INTCSIHTIC and INTCSIHTIR in FIFO mode.

Access: Readable/writable in 32-bit units.

Address: <CSIHn_base> + 1000_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CSIHnFES[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CSIHnFFS[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.21 CSIHnMCTL1 Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
22 to 16	CSIHnFES[6:0]	Selects the conditions to generate the INTCSIHTIC interrupt (transmit data empty) in FIFO mode. When the number of unsent data to be transmitted in FIFO (checked by the CSIHnSTR0.CSIHnSPF[7:0] bit) and CSIHnMCTL1.CSIHnFES[6:0] match, the INTCSIHTIC interrupt request is generated.
15 to 7	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
6 to 0	CSIHnFFS[6:0]	Selects the conditions to generate the INTCSIHTIR interrupt (receive data full) in FIFO mode. When the number of received data in FIFO (checked by the CSIHnSTR0.CSIHnSRP[7:0] bit) and (128 - CSIHnMCTL1.CSIHnFFS[6:0]) match, the INTCSIHTIR interrupt request is generated.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.9 CSIHnMCTL2 — CSIHn Memory Control Register 2

This register controls the operation of the memory in dual buffer or transmit-only buffer mode and triggers to start communication in buffer mode.

Access: Readable/writable in 32-bit units.

Address: <CSIHn_base> + 1004_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHn BTST	—	—	—	—	—	—	—	CSIHnND[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CSIHnSOP[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.22 CSIHnMCTL2 Register Contents (1/2)

Bit Position	Bit Name	Function
31	CSIHnBTST	Provides a start trigger for buffer transfer. 0: No operation. 1: Issues the start transfer command. The read value is always 0. **CAUTION:** This bit can only be used in dual buffer mode and transmit-only buffer mode.
30 to 24	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
23 to 16	CSIHnND[7:0]	Specifies the number of data for each memory mode. The read value indicates the number of remaining communication data.

CSIHnND[7:0]	Dual Buffer Mode	Transmit-Only Buffer Mode	FIFO Mode	Direct Access Mode
00_H	Send 0 data	Send 0 data	No influence	No influence
01_H	Send 1 data	Send 1 data	No influence	No influence
:	:	:	No influence	No influence
$3F_H$	Send 63 data	Send 63 data	No influence	No influence
40_H	Send 64 data	Send 64 data	No influence	No influence
:	Prohibited	:	No influence	No influence
$7F_H$	Prohibited	Send 127 data	No influence	No influence
80_H	Prohibited	Send 128 data	No influence	No influence
Other than the above	Setting is prohibited.			

The values are automatically decremented after data transfer (Not decremented in direct access mode).

Table 11.22 CSIHnMCTL2 Register Contents (2/2)

Bit Position	Bit Name	Function
15 to 7	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
6 to 0	CSIHnSOP[6:0]	Selects the pointer of the data to be sent. If communication is forced to stop by setting CSIHnCTL0.CSIHnPWR to 0 or CSIHnSTCR0.CSIHnPCT to 1, these bits are cleared by hardware. In FIFO mode, these bits indicate the send address.

CSIHnSOP[6:0]	Dual Buffer Mode	Transmit-Only Buffer Mode	FIFO Mode	Direct Access Mode
00_H	0000_H	0000_H	0000_H	No influence
01_H	0004_H	0004_H	0004_H	No influence
:	:	:	:	No influence
$3F_H$	$00FC_H$	$00FC_H$	$00FC_H$	No influence
40_H	Prohibited	0100_H	0100_H	No influence
:	Prohibited	:	:	No influence
$7F_H$	Prohibited	$01FC_H$	$01FC_H$	No influence

CAUTION: In direct access mode, these bits are not incremented.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.10 CSIHnMRWP0 — CSIHn Memory Read/Write Pointer Register 0

This register sets the pointers for reading from and writing to the dual or transmit-only buffer.

Access: Readable/writable in 32-bit units.

Address: <CSIHn_base> + 1018_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CSIHnRRA[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CSIHnTRWA[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.23 CSIHnMRWP0 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
22 to 16	CSIHnRRA[6:0]	These bits select the read pointer of the receive buffer

CSIHnRRA[6:0]	Dual Buffer Mode	Transmit-Only Buffer Mode	FIFO Mode	Direct Access Mode
00_H	0000_H	No influence	0000_H	No influence
01_H	0004_H	No influence	0004_H	No influence
:	:	No influence	:	No influence
$3F_H$	$00FC_H$	No influence	$00FC_H$	No influence
40_H	Prohibited	No influence	0100_H	No influence
:	Prohibited	No influence	:	No influence
$7F_H$	Prohibited	No influence	$01FC_H$	No influence

These bits are automatically incremented when received data is read.

If an overrun error is generated while reading the CSIHnRX0W or CSIHnRX0H register, the read pointer is not incremented.

These bits are cleared when CSIHnSTCR0.CSIHnPCT is set to 1.

In direct access mode and transmit-only buffer mode, these bits are not incremented.

To perform write access in transmit-only buffer mode, set 0000_H to these bits.

In FIFO mode, these bits indicate the read address of the received data.

15 to 7	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.

Table 11.23 CSIHnMRWP0 Register Contents (2/2)

Bit Position	Bit Name	Function
6 to 0	CSIHnTRWA[6:0]	These bits select the read/write pointer of the transmit buffer.

CSIHn TRWA[6:0]	Dual Buffer Mode	Transmit-Only Buffer Mode	FIFO Mode	Direct Access Mode
00_H	0000_H	0000_H	0000_H	No influence
01_H	0004_H	0004_H	0004_H	No influence
:	:	:	:	No influence
$3F_H$	$00FC_H$	$00FC_H$	$00FC_H$	No influence
40_H	Prohibited	0100_H	0100_H	No influence
:	Prohibited	:	:	No influence
$7F_H$	Prohibited	$01FC_H$	$01FC_H$	No influence

These bits are automatically incremented when the transmission data is written or read.
These bits are cleared when CSIHnSTCR0.CSIHnPCT is set to 1.
In direct access mode, these bits are not incremented.
In FIFO mode, these bits indicate the read/write address of transmission data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.11 CSIHnCFGx — CSIHn Configuration Register x

These four registers specify for each chip select signal CSIHTCSSx prescaler, parity, data length, recessive configuration for broadcasting, serial data direction, clock and data phase, idle enforcement configuration, idle time, hold time, inter-data time and setup time.

Slave mode

In slave mode, the transmission protocol setting of the CSIHnCFG0 register is effective.

- CSIHnPSx[1:0]: parity usage
- CSIHnDLSx[3:0]: data length selection
- CSIHnDIRx: data direction
- CSIHnCKPx, CSIHnDAPx: clock and data phase

In slave mode, set 0 to all the bits other than above in the CSIHnCFG0 register, and the CSIHnCFG1 to CSIHnCFG3 registers.

Access: Readable/writable in 32-bit units.

Address: CSIHnCFG0: <CSIHn_base> + 1044_H
CSIHnCFG1: <CSIHn_base> + 1048_H
CSIHnCFG2: <CSIHn_base> + $104C_H$
CSIHnCFG3: <CSIHn_base> + 1050_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHnBRSSx[1:0]		CSIHnPSx[1:0]		CSIHnDLSx[3:0]				—	—	—	—	CSIHnRCBx	CSIHnDIRx	CSIHnCKPx	CSIHnDAPx
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnIDLx	CSIHnIDx[2:0]			CSIHnHDx[3:0]				CSIHnINx[3:0]				CSIHnSPx[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.24 CSIHnCFGx Register Contents (1/4)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>31, 30</td><td>CSIHnBRSSx [1:0]</td><td>Selects the baud rate setting register (CSIHnBRSy).
<table>
<tr><th>CSIHn BRSSx1</th><th>CSIHn BRSSx0</th><th>Baud Rate Setting Register Selection</th></tr>
<tr><td>0</td><td>0</td><td>The baud rate is set according to the CSIHnBRS0 setting</td></tr>
<tr><td>0</td><td>1</td><td>The baud rate is set according to the CSIHnBRS1 setting</td></tr>
<tr><td>1</td><td>0</td><td>The baud rate is set according to the CSIHnBRS2 setting</td></tr>
<tr><td>1</td><td>1</td><td>The baud rate is set according to the CSIHnBRS3 setting</td></tr>
</table>
The maximum value for setting the transfer clock frequency, combining the CSIHnCTL2.CSIHnPRS[2:0] setting, must be as follows:
Master mode: PCLK/8
Slave mode: PCLK/16</td></tr>
<tr><td>29, 28</td><td>CSIHnPSx[1:0]</td><td>Selects the parity for sending or receiving chip select signal x.
<table>
<tr><th>CSIHnPSx1</th><th>CSIHnPSx0</th><th>Transmission</th><th>Reception</th></tr>
<tr><td>0</td><td>0</td><td>No parity is transmitted</td><td>No parity is expected</td></tr>
<tr><td>0</td><td>1</td><td>Adds parity bit fixed to 0</td><td>Parity bit is expected but not judged</td></tr>
<tr><td>1</td><td>0</td><td>Adds odd parity only</td><td>Odd parity bit is expected</td></tr>
<tr><td>1</td><td>1</td><td>Adds even parity</td><td>Even parity bit is expected</td></tr>
</table></td></tr>
<tr><td>27 to 24</td><td>CSIHnDLSx [3:0]</td><td>Selects the data length for chip select signal x.
<table>
<tr><th>CSIHnDLSx[3:0]</th><th>Data Length</th></tr>
<tr><td>0000_B</td><td>16 bits</td></tr>
<tr><td>0001_B</td><td>1 bit</td></tr>
<tr><td>0010_B</td><td>2 bits</td></tr>
<tr><td>:</td><td>:</td></tr>
<tr><td>1111_B</td><td>15 bits</td></tr>
</table>
CAUTION: When CSIHnTX0W.CSIHnEDL = 1, the setting of this bit has no effect. (the data length is 16 bits)
When CSIHnTX0W.CSIHnEDL = 0, the setting of this bit is valid. Only when the previous transmit data is 16 bits while CSIHnEDL = 1, writing 1 to this bit is enabled.</td></tr>
<tr><td>23 to 20</td><td>Reserved</td><td>When read, the value after a reset is read. When writing, write the value after a reset.</td></tr>
<tr><td>19</td><td>CSIHnRCBx</td><td>Selects the recessive configuration for broadcasting for chip select x:
0: Dominant (higher priority)
1: Recessive (lower priority)
For details, see Section 11.4.9.1, Configuration Registers</td></tr>
<tr><td>18</td><td>CSIHnDIRx</td><td>Selects the serial data direction of chip select signal x.
0: Data is transmitted/received with MSB first.
1: Data is transmitted/received with LSB first.
For details, refer to Section 11.4.15, Serial Data Direction Selection.</td></tr>
</table>

Table 11.24 CSIHnCFGx Register Contents (2/4)

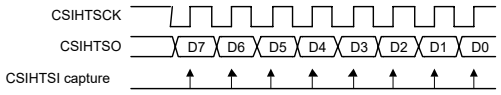
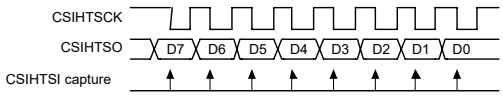
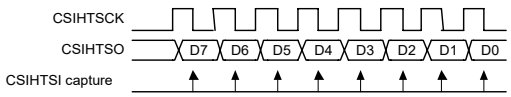
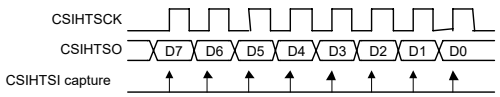
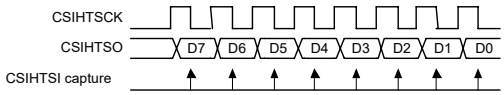
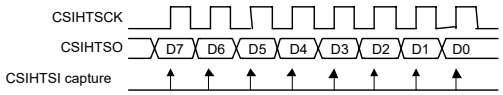
<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>17</td><td>CSIHnCKPx</td><td>CSIHnCKPx: Clock phase selection bit</td></tr>
<tr><td>16</td><td>CSIHnDAPx</td><td>CSIHnDAPx: Data phase selection bit
• CSIHnCTL1.CSIHnCKR = 0
<table>
<tr><th>CSIHn CKPx</th><th>CSIHn DAPx</th><th>Clock And Data Phase Selection</th></tr>
<tr><td>0</td><td>0</td><td> 
</td></tr>
<tr><td>0</td><td>1</td><td> 
</td></tr>
<tr><td>1</td><td>0</td><td> 
</td></tr>
<tr><td>1</td><td>1</td><td> 
</td></tr>
</table>
• CSIHnCTL1.CSIHnCKR = 1
<table>
<tr><th>CSIHn CKPx</th><th>CSIHn DAPx</th><th>Clock And Data Phase Selection</th></tr>
<tr><td>0</td><td>0</td><td> 
</td></tr>
<tr><td>0</td><td>1</td><td> 
</td></tr>
<tr><td>1</td><td>×</td><td>Setting prohibited</td></tr>
</table></td></tr>
<tr><td>15</td><td>CSIHnIDLx</td><td>Selects the idle enforcement configuration for chip select x:
0: If the CSIHnTX0W.CSIHnCSx setting of two consecutive transfers is different, an idle state is inserted between two transfers. If the CSIHnTX0W.CSIHnCSx setting of two consecutive transfers is the same, an idle state is not inserted between two transfers.
1: Regardless of the CSIHnTX0W.CSIHnCSx setting of two consecutive transfers, an idle state is inserted between two transfers.
This bit is only available in master mode.
For details about the enforced idle state, see Section 11.4.21, Enforced Chip Select Idle Setting.</td></tr>
</table>

Table 11.24 CSIHnCFGx Register Contents (3/4)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>14 to 12</td><td>CSIHnIDx[2:0]</td><td>Selects the idle time for chip select signal x.
<table>
<tr><th>CSIHnIDx[2:0]</th><th>Idle Time</th></tr>
<tr><td>000_B</td><td>0.5 transmission clock cycle</td></tr>
<tr><td>001_B</td><td>1.0 transmission clock cycle</td></tr>
<tr><td>010_B</td><td>1.5 transmission clock cycles</td></tr>
<tr><td>011_B</td><td>2.5 transmission clock cycles</td></tr>
<tr><td>100_B</td><td>3.5 transmission clock cycles</td></tr>
<tr><td>101_B</td><td>4.5 transmission clock cycles</td></tr>
<tr><td>110_B</td><td>6.5 transmission clock cycles</td></tr>
<tr><td>111_B</td><td>8.5 transmission clock cycles</td></tr>
</table>
These bits are only available in master mode.</td></tr>
<tr><td>11 to 8</td><td>CSIHnHDx[3:0]</td><td>Specifies the hold time for chip select signal x in transmission clock cycles.
<table>
<tr><th>CSIHnHDx [3:0]</th><th>Hold Time With CSIHnCTL1.CSIHnSIT = 0</th><th>Hold Time With CSIHnCTL1.CSIHnSIT = 1</th></tr>
<tr><td>0000_B</td><td>0.5 transmission clock cycle</td><td>1.0 transmission clock cycle</td></tr>
<tr><td>0001_B</td><td>1.0 transmission clock cycle</td><td>1.5 transmission clock cycles</td></tr>
<tr><td>0010_B</td><td>1.5 transmission clock cycles</td><td>2.0 transmission clock cycles</td></tr>
<tr><td>0011_B</td><td>2.5 transmission clock cycles</td><td>3.0 transmission clock cycles</td></tr>
<tr><td>0100_B</td><td>3.5 transmission clock cycles</td><td>4.0 transmission clock cycles</td></tr>
<tr><td>0101_B</td><td>4.5 transmission clock cycles</td><td>5.0 transmission clock cycles</td></tr>
<tr><td>0110_B</td><td>6.5 transmission clock cycles</td><td>7.0 transmission clock cycles</td></tr>
<tr><td>0111_B</td><td>8.5 transmission clock cycles</td><td>9.0 transmission clock cycles</td></tr>
<tr><td>1000_B</td><td>9.5 transmission clock cycles</td><td>10.0 transmission clock cycles</td></tr>
<tr><td>1001_B</td><td>10.5 transmission clock cycles</td><td>11.0 transmission clock cycles</td></tr>
<tr><td>1010_B</td><td>11.5 transmission clock cycles</td><td>12.0 transmission clock cycles</td></tr>
<tr><td>1011_B</td><td>12.5 transmission clock cycles</td><td>13.0 transmission clock cycles</td></tr>
<tr><td>1100_B</td><td>14.5 transmission clock cycles</td><td>15.0 transmission clock cycles</td></tr>
<tr><td>1101_B</td><td>16.5 transmission clock cycles</td><td>17.0 transmission clock cycles</td></tr>
<tr><td>1110_B</td><td>18.5 transmission clock cycles</td><td>19.0 transmission clock cycles</td></tr>
<tr><td>1111_B</td><td>20.5 transmission clock cycles</td><td>21.0 transmission clock cycles</td></tr>
</table>
These bits are only available in master mode.</td></tr>
</table>

Table 11.24 CSIHnCFGx Register Contents (4/4)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>7 to 4</td><td>CSIHnINx[3:0]</td><td>Specifies the inter-data time for chip select signal x in transmission clock cycles.
<table>
<tr><th>CSIHnINx[3:0]</th><th>Inter-data time when CSIHnCTL1.CSIHnSIT = 0</th><th>Inter-data time when CSIHnCTL1.CSIHnSIT = 1</th></tr>
<tr><td>0000_B</td><td>0.0 transmission clock cycle</td><td>0.5 transmission clock cycle</td></tr>
<tr><td>0001_B</td><td>0.5 transmission clock cycle</td><td>1.0 transmission clock cycle</td></tr>
<tr><td>0010_B</td><td>1.0 transmission clock cycle</td><td>1.5 transmission clock cycles</td></tr>
<tr><td>0011_B</td><td>2.0 transmission clock cycles</td><td>2.5 transmission clock cycles</td></tr>
<tr><td>0100_B</td><td>3.0 transmission clock cycles</td><td>3.5 transmission clock cycles</td></tr>
<tr><td>0101_B</td><td>4.0 transmission clock cycles</td><td>4.5 transmission clock cycles</td></tr>
<tr><td>0110_B</td><td>6.0 transmission clock cycles</td><td>6.5 transmission clock cycles</td></tr>
<tr><td>0111_B</td><td>8.0 transmission clock cycles</td><td>8.5 transmission clock cycles</td></tr>
<tr><td>1000_B</td><td>9.0 transmission clock cycles</td><td>9.5 transmission clock cycles</td></tr>
<tr><td>1001_B</td><td>10.0 transmission clock cycles</td><td>10.5 transmission clock cycles</td></tr>
<tr><td>1010_B</td><td>11.0 transmission clock cycles</td><td>11.5 transmission clock cycles</td></tr>
<tr><td>1011_B</td><td>12.0 transmission clock cycles</td><td>12.5 transmission clock cycles</td></tr>
<tr><td>1100_B</td><td>14.0 transmission clock cycles</td><td>14.5 transmission clock cycles</td></tr>
<tr><td>1101_B</td><td>16.0 transmission clock cycles</td><td>16.5 transmission clock cycles</td></tr>
<tr><td>1110_B</td><td>18.0 transmission clock cycles</td><td>18.5 transmission clock cycles</td></tr>
<tr><td>1111_B</td><td>20.0 transmission clock cycles</td><td>20.5 transmission clock cycles</td></tr>
</table>
These bits are only available in master mode.</td></tr>
<tr><td>3 to 0</td><td>CSIHnSPx[3:0]</td><td>Specifies the setup time for chip select signal x in transmission clock cycles.
<table>
<tr><th>CSIHnSPx[3:0]</th><th>Setup time</th></tr>
<tr><td>0000_B</td><td>0.5 transmission clock cycle</td></tr>
<tr><td>0001_B</td><td>1.0 transmission clock cycle</td></tr>
<tr><td>0010_B</td><td>1.5 transmission clock cycles</td></tr>
<tr><td>0011_B</td><td>2.5 transmission clock cycles</td></tr>
<tr><td>0100_B</td><td>3.5 transmission clock cycles</td></tr>
<tr><td>0101_B</td><td>4.5 transmission clock cycles</td></tr>
<tr><td>0110_B</td><td>6.5 transmission clock cycles</td></tr>
<tr><td>0111_B</td><td>8.5 transmission clock cycles</td></tr>
<tr><td>1000_B</td><td>9.5 transmission clock cycles</td></tr>
<tr><td>1001_B</td><td>10.5 transmission clock cycles</td></tr>
<tr><td>1010_B</td><td>11.5 transmission clock cycles</td></tr>
<tr><td>1011_B</td><td>12.5 transmission clock cycles</td></tr>
<tr><td>1100_B</td><td>14.5 transmission clock cycles</td></tr>
<tr><td>1101_B</td><td>16.5 transmission clock cycles</td></tr>
<tr><td>1110_B</td><td>18.5 transmission clock cycles</td></tr>
<tr><td>1111_B</td><td>20.5 transmission clock cycles</td></tr>
</table>
These bits are only available in master mode.</td></tr>
</table>

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.12 CSIHnTX0W — CSIHn Transmit Data Register 0 for Word Access

This register stores transmission data. In addition, it specifies the communication interrupt request, the end-of-job, the extended data length, and the chip select activation.

Access: Readable/writable in 32-bit units.

Address: <CSIHn_base> + 1008_H

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHnCIRE	CSIHnEOJ	CSIHnEDL	—	—	—	—	—	—	—	—	—	CSIHnCS3	CSIHnCS2	CSIHnCS1	CSIHnCS0
Value after reset	—	—	—	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnTX[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.25 CSIHnTX0W Register Contents (1/2)

Bit Position	Bit Name	Function
31	CSIHnCIRE	Enables the communication interrupt request INTCSIHTIC in dual buffer or transmit-only buffer mode, or the job completion interrupt INTCSIHTIJC request in FIFO mode. 0: No interrupt is requested. 1: An interrupt is requested. Generates interrupt INTCSIHTIC or INTCSIHTIJC after transmission. For details, see **Section 11.4.3, INTCSIHTIC (Communication Status Interrupt)** and **Section 11.4.6, INTCSIHTIJC (Job Completion Interrupt)**. **CAUTION:** This bit is only valid when job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
30	CSIHnEOJ	Specifies the end of a job. 0: Indicates that the job does not end. The job continues. 1: Indicates end-of-job data. **CAUTION:** This bit is only valid when job mode is enabled (CSIHnCTL1.CSIHnJE = 1). This bit must be set to 0 in slave mode.

Table 11.25 CSIHnTX0W Register Contents (2/2)

Bit Position	Bit Name	Function
29	CSIHnEDL	Specifies whether the associated data requires the extended data length (EDL) option. 0: Normal operation 1: Enables the extended data length. The associated data is transmitted as a 16-bit packet. No inter-data time or idle time will be inserted after the data is transmitted. If CSIHnCTL1.CSIHnEDLE = 1 and CSIHnTX0W.CSIHnEDL = 1, the subsequent data must have the same CS selection. If CS is modified for the subsequent data, the correct operation is not assured. **CAUTION:** This bit is only available if CSIHnCTL1.CSIHnEDLE = 1.
28 to 20	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
19 to 16	CSIHnCS[3:0]	Activates one or several chip select signals. 0: Activates chip select x for the associated transmission. 1: Deactivates chip select x for the associated transmission. Setting CSIHnTX0W.CSIHnCS[3:0] = F_H is prohibited. **CAUTION:** If several chip select signals are enabled for broadcasting, the configuration of one with CSIHnCFGx.CSIHnRCBx = 0 (dominant) is used. In this case, all dominant chip select signals must be set to precisely the same value. In slave mode, set the CSIHnCS[3:0] bits to E_H.
15 to 0	CSIHnTX[15:0]	Stores the transmission data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.13 CSIHnTX0H — CSIHn Transmit Data Register 0 for Half Word Access

This register stores the transmission data. This register is the same as bits 15 to 0 of register CSIHnTX0W.

The 16 high-order bits of CSIHnTX0W are applied for transfer. Set transmit data to CSIHnTX0W before using this register because the value of CSIHnTX0W is undefined after a reset.

Access: Readable/writable in 16-bit units.

Address: <CSIHn_base> + $100C_H$

Value after reset: Undefined

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnTX[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.26 CSIHnTX0H Register Contents

Bit Position	Bit Name	Function
15 to 0	CSIHnTX[15:0]	Stores the transmission data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.14 CSIHnRX0W — CSIHn Receive Data Register 0 for Word Access

This register stores the received data.

Access: Readable in 32-bit units.

Address: <CSIHn_base> + 1010_H

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CSIHn RPE	CSIHn TDCE	—	—	—	—	CSIHn CS3	CSIHn CS2	CSIHn CS1	CSIHn CS0
Value after reset	0	0	0	0	0	0	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnRX[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.27 CSIHnRX0W Register Contents

Bit Position	Bit Name	Function
31 to 26	Reserved	When read, the value after a reset is read.
25	CSIHnRPE	Indicates whether a reception data parity error was detected. 0: No parity error was detected on the associated reception data. 1: A parity error was detected on the associated reception data.
24	CSIHnTDCE	Indicates whether a transmission data consistency error was detected. 0: No consistency error was detected on the associated transmission data. 1: A consistency error was detected on the associated transmission data.
23 to 20	Reserved	When read, the value after a reset is read.
19 to 16	CSIHnCSx (x = 3 to 0)	Indicates which chip select signal was activated. 0: Chip select x was activated for the associated reception. 1: Chip select x was deactivated for the associated reception.
15 to 0	CSIHnRX [15:0]	Stores the received data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.15 CSIHnRX0H — CSIHn Receive Data Register 0 for Half Word Access

This register stores the received data. This register is the same as bits 15 to 0 of register CSIHnRX0W.

Access: Readable in 16-bit units.

Address: <CSIHn_base> + 1014_H

Value after reset: Undefined

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnRX[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.28 CSIHnRX0H Register Contents

Bit Position	Bit Name	Function
15 to 0	CSIHnRX [15:0]	Stores the received data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.16 CSIHnBRSy — CSIHn Baud Rate Setting Register y (y = 0 to 3)

This register sets the transfer clock frequency for each chip select signal.

With CSIHnCFG0 (to 3).CSIHnBRSSx[1:0] bits, one of the four types of transfer clock frequency settings can be selected for each chip select signal. For details of transfer clock frequency setting, refer to **Section 11.4.11, Transmission clock selection**.

Access: Readable/writable in 16-bit units.

Address: CSIHnBRS0: <CSIHn_base> + 1068_H

CSIHnBRS1: <CSIHn_base> + $106C_H$

CSIHnBRS2: <CSIHn_base> + 1070_H

CSIHnBRS3: <CSIHn_base> + 1074_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CSIHnBRS[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.29 CSIHnBRSy Register Contents

Bit Position	Bit Name	Function
15 to 12	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
11 to 0	CSIHnBRS [11:0]	0: BRG stopped 1: PCLK / (2^{α} × 1 × 2) 2: PCLK / (2^{α} × 2 × 2) 3: PCLK / (2^{α} × 3 × 2) 4: PCLK / (2^{α} × 4 × 2) . . . 4095: PCLK / (2^{α} × 4095 × 2) α is the value of CSIHnCTL2.CSIHnPRS[2:0].

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.4 Functions

11.4.1 Overview of Interrupt Function

CSIH can generate the interrupt requests listed in the table below:

Table 11.30 Interrupt Generation

Memory mode	Interrupt	Cause of Interrupt	
		Job Mode Disabled CSIHnCTL1.CSIHnJE = 0	Job Mode Enabled CSIHnCTL1.CSIHnJE = 1
FIFO	INTCSIHTIC (communication status interrupt)	Tx data empty*1	Tx data empty*1 except job abortion*4
	INTCSIHTIR (reception status interrupt)	Rx data full*2 and CSIHnCTL0.CSIHnRXE = 1	Rx data full*2 and CSIHnCTL0.CSIHnRXE = 1
	INTCSIHTIRE (communication error interrupt)	Error detected	Error detected
	INTCSIHTIJC*3 (job completion interrupt)	Not applicable	CSIHnTX0W.CSIHnCIRE = 1, (when it is not Tx data empty) or job abortion*4
Transmit-only buffer	INTCSIHTIC (communication status interrupt)	End of communication	CSIHnTX0W.CSIHnCIRE = 1 and (CSIHnCTL0.CSIHnJOBE = 0 or CSIHnTX0W.CSIHnEOJ = 0)
	INTCSIHTIR (reception status interrupt)	Data received and CSIHnCTL0.CSIHnRXE = 1	Data received and CSIHnCTL0.CSIHnRXE = 1
	INTCSIHTIRE (communication error interrupt)	Error detected	Error detected
	INTCSIHTIJC*3 (job completion interrupt)	Not applicable	Job abortion*4
Dual buffer	INTCSIHTIC (communication status interrupt)	End of communication	CSIHnTX0W.CSIHnCIRE = 1 and (CSIHnCTL0.CSIHnJOBE = 0 or CSIHnTX0W.CSIHnEOJ = 0)
	INTCSIHTIR (reception status interrupt)	End of communication and CSIHnCTL0.CSIHnRXE = 1	Data received and CSIHnCTL0.CSIHnRXE = 1
	INTCSIHTIRE (communication error interrupt)	Error detected	Error detected
	INTCSIHTIJC*3 (job completion interrupt)	Not applicable	Job abortion*4
Direct access	INTCSIHTIC (communication status interrupt)	1 data transferred	1 data transferred, (except the state of job abortion*4)
	INTCSIHTIR (reception status interrupt)	Data received and CSIHnCTL0.CSIHnRXE = 1	Data received and CSIHnCTL0.CSIHnRXE = 1
	INTCSIHTIRE (communication error interrupt)	Error detected	Error detected
	INTCSIHTIJC*3 (job completion interrupt)	Not applicable	Job abortion*4

Note 1. “Tx data empty” refers to the FIFO fill level, defined by CSIHnMCTL1.CSIHnFES[6:0].

Note 2. “Rx data full” refers to the FIFO fill level, defined by CSIHnMCTL1.CSIHnFFS[6:0].

Note 3. INTCSIHTIJC is not available in slave mode.

Note 4. Job abortion condition: CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1.
During high priority communication in transmit-only buffer mode, the operation is the same as that in direct access mode.

The communication error interrupt INTCSIHTIRE is generated when an error is detected. The generation of the other interrupts depends on the memory mode, the job mode, and - in case of the job completion interrupt INTCSIHTIJC - also the operating mode.

The job completion interrupt INTCSIHTIJC is only generated when job mode is enabled (CSIHnCTL1.CSIHnJE = 1). It is not available in slave mode.

11.4.2 Interrupt Delay

In master mode, all interrupts generated by the master can be delayed by one half period of the transmission clock, CSIHTSCK. This is not possible in slave mode.

The delay is specified by setting bit CSIHnCTL1.CSIHnSIT = 1.

The following example illustrates the interrupt delay function, assuming a setting of CSIHnCTL1.CSIHnSIT = 1 (interrupt delay enabled), CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0 (Normal clock and data phases), and CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B (data length 8 bits).

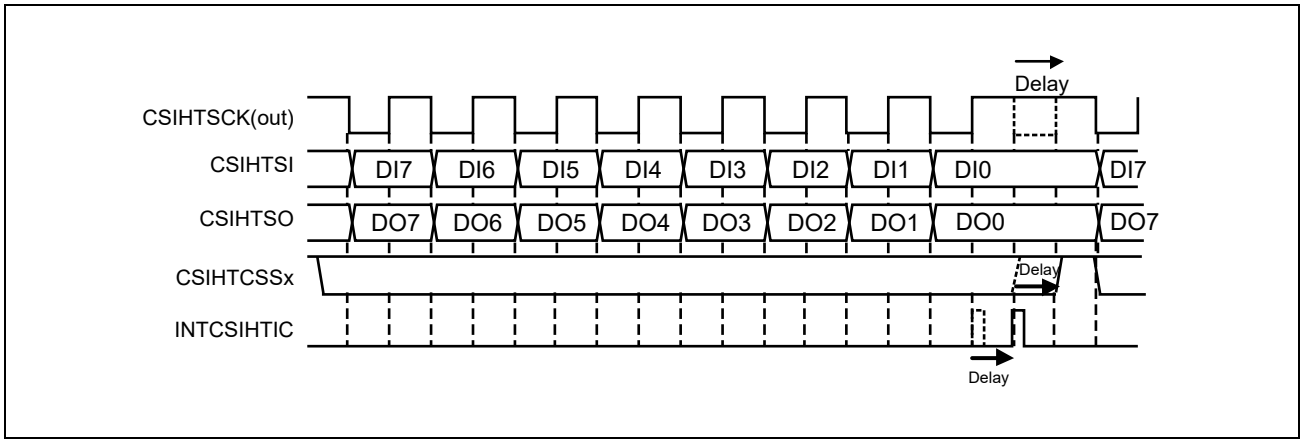


Figure 11.2 Interrupt Delay Function (CSIHnCTL1.CSIHnSIT = 1)

Setting CSIHnCTL1.CSIHnSIT = 1 adds half period delay to the transmission clock. This delays also the end of the present chip select signal (CSIHTCSSx).

11.4.3 INTCSIHTIC (Communication Status Interrupt)

Depending on the memory mode and the job mode, this interrupt is generated according to the conditions shown in the following table.

Table 11.31 INTCSIHTIC Interrupt Generation

Memory Mode	Cause of Interrupt	
	Job Mode Disabled CSIHnCTL1.CSIHnJE = 0	Job Mode Enabled CSIHnCTL1.CSIHnJE = 1
FIFO	This interrupt is generated when transmission data is about to be missing in the FIFO, indicating to the application that new data should be added. INTCSIHTIC is generated, if the number of data to be sent remaining in the FIFO (CSIHnSTR0.CSIHnSPF[7:0]) equals CSIHnMTCL1.CSIHnFES[6:0].	Similar to “when CSIHnJE is 0”, an interrupt is generated when the number of transmit data remained in the FIFO CSIHnSTR0.CSIHnSPF[7:0] is the same number as CSIHnMCTL1.CSIHnFES[6:0]. At the time of job abortion, no interrupt is generated.
Transmit-only buffer, Dual buffer	Generated at the End of communication. (Specified by the CSIHnMTLC2.CSIHnND[7:0] bit)	Generated when data with CSIHnTX0W.CSIHnCIRE = 1 is sent. Note that if data with CSIHnTX0W.CSIHnCIRE = 1 and job abortion*1 are sent, the INTCSIHTIJC interrupt is generated instead of INTCSIHTIC.
Direct access	Generated after every data transfer.	Generated after every data transfer, except when the communication was aborted.

Note 1. Job abortion condition: CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1.
During high priority communication in transmit-only buffer mode, the operation is the same as that in direct access mode.

11.4.3.1 INTCSIHTIC in direct access mode

The examples below show the INTCSIHTIC behavior in direct access mode.

The examples assume:

- Master mode
- Direct access mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)

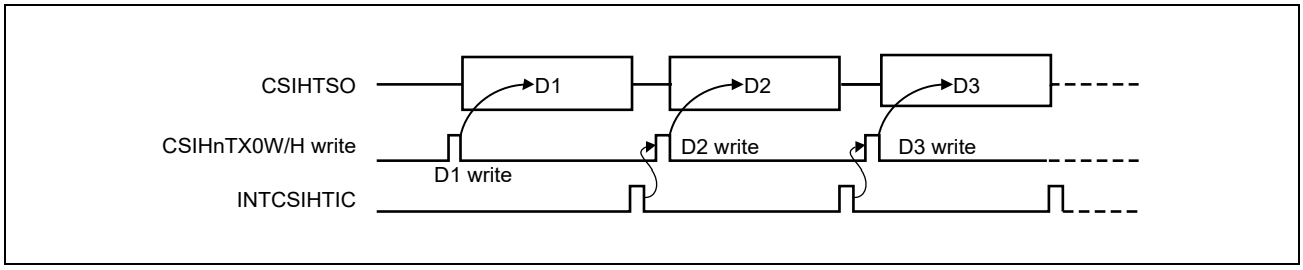


Figure 11.3 Generation of INTCSIHTIC after Transfer (CSIHnCTL1.CSIHnSLIT = 0)

If job mode is enabled (CSIHnCTL1.CSIHnJE = 1) and a job ends because data is sent with CSIHnTX0W.CSIHnEOJ = 1 and communication stop is requested (CSIHnCTL0.CSIHnJOBE = 1), then INTCSIHTIC is replaced by the job completion interrupt INTCSIHTIJC.

INTCSIHTIC can also be set up to occur as soon as the CSIHnTX0W/H register is free for the next data.

This is specified by setting CSIHnCTL1.CSIHnSLIT = 1.

The effect is illustrated in the figure below.

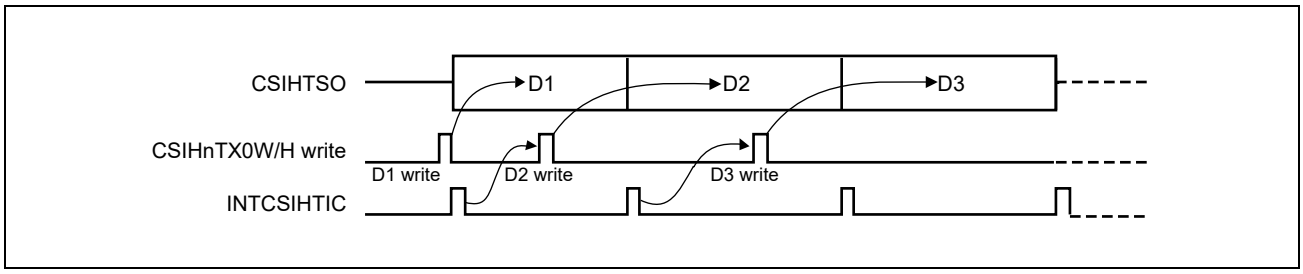


Figure 11.4 Immediate Generation of INTCSIHTIC (CSIHnCTL1.CSIHnSLIT = 1)

Thus, the new data can be written in advance.

NOTE

During high priority communication in transmit-only buffer mode, the operation is the same as that in direct access mode.

11.4.3.2 INTCSIHTIC in FIFO mode

The example below shows the INTCSIHTIC behavior in FIFO mode.

The example assumes:

- Master mode
- FIFO mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)

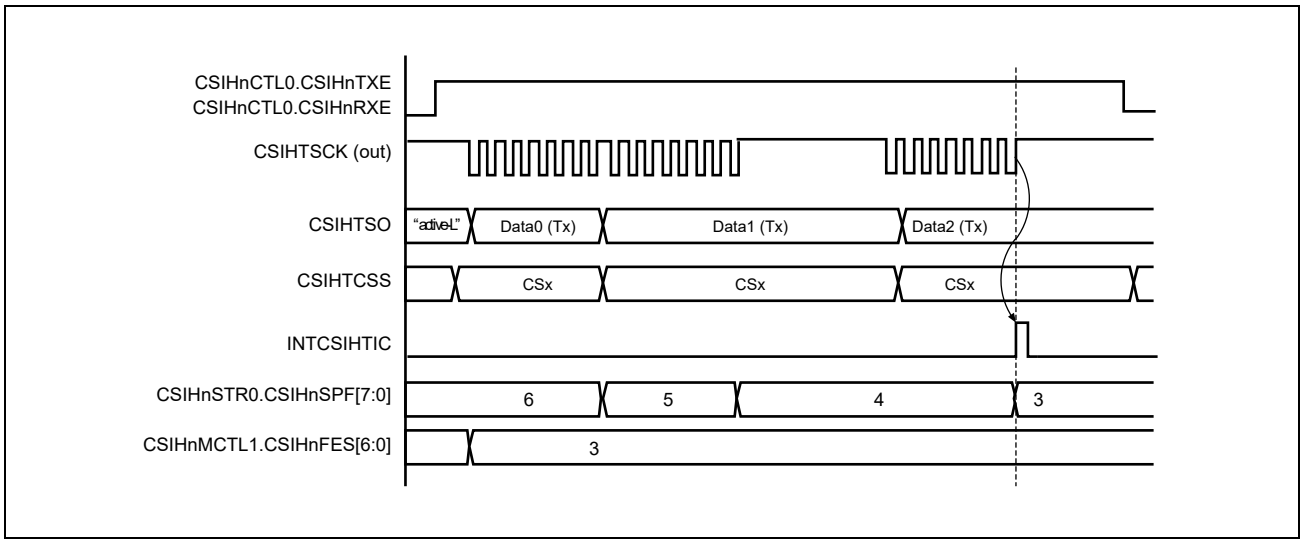


Figure 11.5 Generation of INTCSIHTIC in FIFO Memory Mode

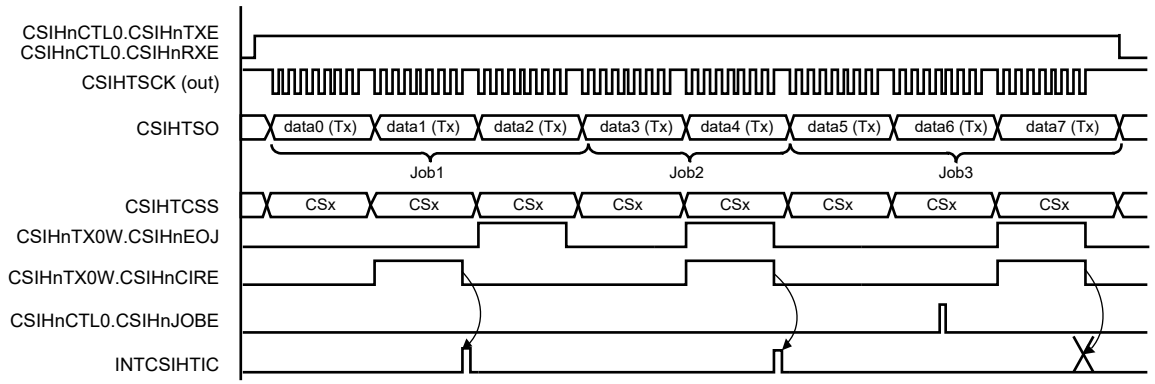
The condition for “FIFO empty” is specified in CSIHnMCTL1.CSIHnFES[6:0]. In the example of the diagram above, the number of unsent data in FIFO is set to 3. CSIHnSTR0.CSIHnSPF[7:0] indicates the number of unsent data. When both match, the interrupt INTCSIHTIC occurs.

11.4.3.3 INTCSIHTIC in job mode

The example below shows the INTCSIHTIC behavior in job mode.

The example assumes:

- Master mode
- Job mode enabled (CSIHnCTL1.CSIHnJE = 1)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)



Note: CSx can be changed per data transfer.

Figure 11.6 Generation of INTCSIHTIC in Job Mode

The rules for generating INTCSIHTIC in job mode are shown in the following table.

Table 11.32 Generation of INTCSIHTIC in Job Mode

CSIHnTX0W. CSIHnEOJ	CSIHnTX0W. CSIHnCIRE	INTCSIHTIC
0	0	Not generated
0	1	Generated
1	0	Not generated
1	1	CSIHnCTL0.CSIHnJOBE = 0: Generated
		CSIHnCTL0.CSIHnJOBE = 1: Not generated, replaced by interrupt INTCSIHTIJC

11.4.4 INTCSIHTIR (Reception Status Interrupt)

Depending on the memory mode and the job mode, this interrupt is generated according to the conditions below.

Table 11.33 INTCSIHTIR Interrupt Generation

Memory Mode	Cause of Interrupt	
	Job Mode Disabled CSIHnCTL1.CSIHnJE = 0	Job Mode Enabled CSIHnCTL1.CSIHnJE = 1
FIFO	This interrupt occurs when CSIHnCTL0.CSIHnRXE is 1 and the FIFO buffer is almost full with received data, indicating to the application that the FIFO must be emptied. INTCSIHTIR is generated, if the number of received data in the FIFO (CSIHnSTR0.CSIHnSRP[7:0]) equals (128 – CSIHnMCTL1.CSIHnFFS[6:0]).	
Dual buffer	Generated when the communication has finished (as specified by the CSIHnMCTL2.CSIHnND[7:0] bit) and CSIHnCTL0.CSIHnRXE = 1.	Generated after every data transfer.
Transmit-only buffer, Direct access	Generated after every data transfer.	

11.4.4.1 INTCSIHTIR in direct access mode

The example below shows the INTCSIHTIR behavior in direct access mode.

The example below assumes

- Master mode
- Direct access mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)

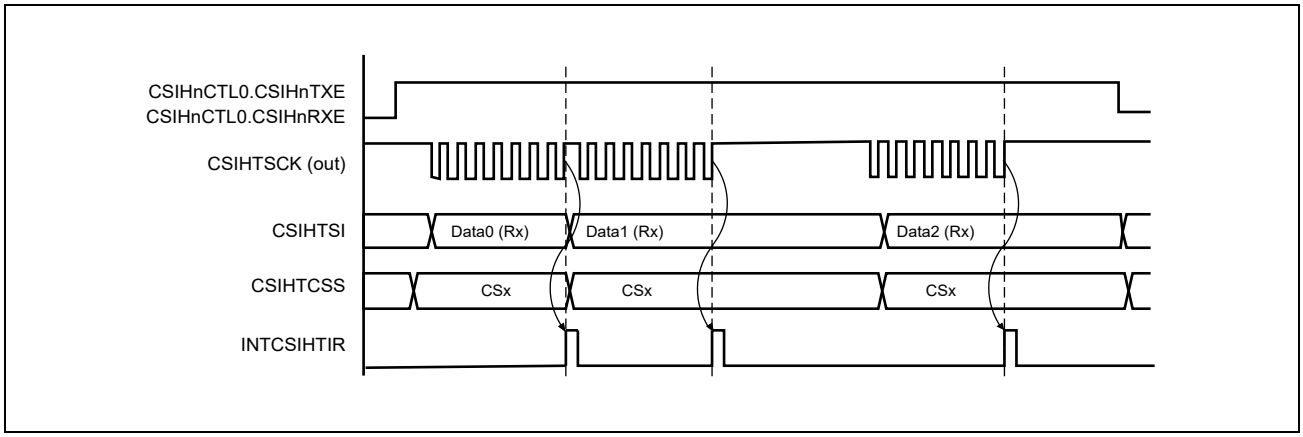


Figure 11.7 Generation of INTCSIHTIR in Direct Access Memory Mode

11.4.4.2 INTCSIHTIR in dual buffer mode

The example below shows the INTCSIHTIR behavior in dual buffer mode.

The example assumes:

- Master mode
- Dual buffer mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Default clock and data phase (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)

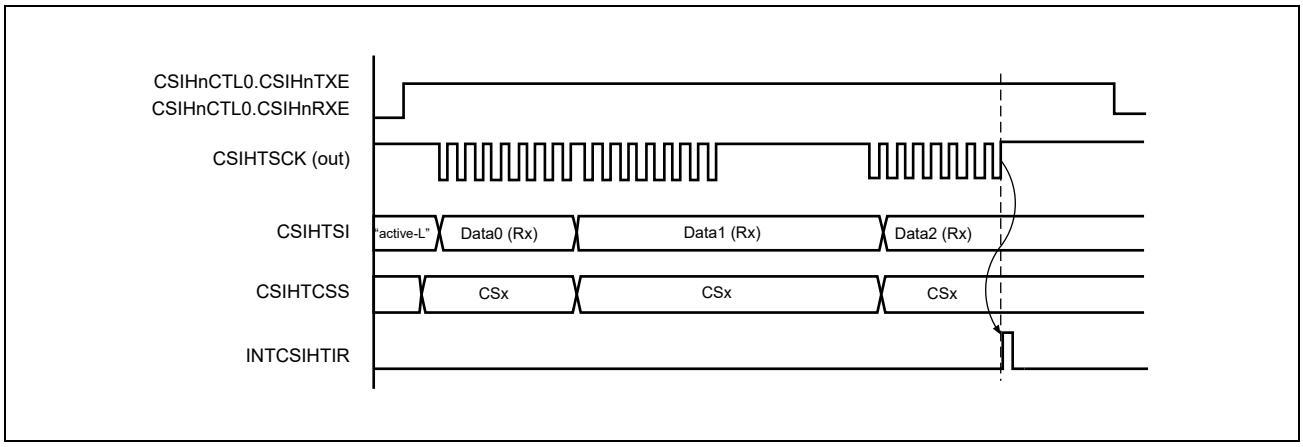


Figure 11.8 Generation of INTCSIHTIR in Dual Buffer Mode

11.4.5 IINTCSIHTIRE (Reception Error Interrupt)

This interrupt is generated whenever an error is detected.

For details about generation interruption timing, see **Section 11.4.18, Error Detection**.

Table 11.34 Data Error Types

Error Type	Communication Status after Communication Error Interrupt	Comment
FIFO overflow error	Interrupt is generated and communication continues	The data are not written to the FIFO buffer and the overflow of data is lost, but communications started before the error is continued.
Parity error	Interrupt is generated and communication continues.	—
Data consistency error	Interrupt is generated and communication continues.	—
Time-out error	Interrupt is generated and communication continues.	—
Overrun error	Condition for errors 1: In FIFO mode, when the number of received data is 0 and CPU reads the CSIHnRX0W/H register, an interrupt is generated and communication continues.	—
	Condition for errors 2: In slave mode, when CSIHnCTL1.CSIHnHSE = 0 (handshake function disabled): (1) In direct access mode or transmit-only buffer mode, when reception is completed while the previous received data is remained in the CSIHnRX0W/H register, an interrupt is generated, and communication continues. (2) In FIFO mode, when reception by the FIFO buffer is completed and the buffer is in the full state, an interrupt is generated. Communication continues.	In slave mode, when CSIHnCTL1.CSIHnHSE = 1 (handshake function enabled), communication is suspended due to handshake, an overrun error is not generated.

The type of error that caused the generation of INTCSIHTIRE is flagged in register CSIHnSTR0.

Additionally a parity error flag and a data consistency error flag are attached to the received data in CSIHnRX0W.

For details about the various error types, see **Section 11.4.18, Error Detection**.

11.4.6 INTCSIHTIJC (Job Completion Interrupt)

This interrupt supports the handling of jobs, see **Section 11.4.9.3, Job Concept**. This interrupt is only available in master mode.

Job mode is enabled by setting CSIHnCTL1.CSIHnJE = 1. When CSIHnCTL1.CSIHnJE = 0, INTCSIHTIJC is not generated.

Depending on the memory mode, this interrupt is generated according to the condition shown in the following table.

Table 11.35 INTCSIHTIJC Interrupt Generation

Memory Mode	Job Mode Disabled CSIHnCTL1.CSIHnJE = 0	Job Mode Enabled CSIHnCTL1.CSIHnJE = 1
FIFO	Not applicable	Indicates that the communication has stopped at the end of a job after a job abortion*1 was triggered. If FIFO empty is not detected and when CSIHnCIRE is 1, INTCSIHTIJC is generated.
Transmit-only buffer		Indicates that the communication has stopped at the end of a job after a job abortion*1 was triggered
Dual buffer		
Direct access		

Note 1. Job abortion condition: CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1

11.4.7 Operating Modes (Master/Slave)

For a particular CSIH module, the master or slave mode determines the source of the serial clock.

11.4.7.1 Master mode

In master mode, the serial transmission clock is generated by the internal baud rate generator (BRG) and provided to the slave(s) by signal CSIHTSCK.

Master mode is enabled by setting CSIHnCTL2.CSIHnPRS[2:0] to any value other than 111_B. In master mode, the BRG frequency can be set by combining the CSIHnCTL2.CSIHnPRS[2:0] bits and the CSIHnBRSy.CSIHnBRS[11:0] bits.

(1) Chip select signals

In master mode, one or several chip select signals can be used. If several slaves are connected to the master, the chip select signals can be used to address one or several of the slaves. Only a selected slave is then enabled for communication.

The communication protocol as well as additional parameters are stored separately for each chip select signal. This makes it possible to adapt the data transfer individually to the requirements of each slave.

For details, see **Section 11.4.9, Chip Selection (CS) Features**.

(2) Clock default setting

The default level of CSIHTSCK depends on the CSIHTSCK clock inversion function bit: It is high when CSIHnCTL1.CSIHnCKR = 0, and is low when CSIHnCTL1.CSIHnCKR = 1.

The example below shows the communication in master mode for data length 8 bits, CSIHnCTL1.CSIHnCKR = 0, CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0, and MSB first:

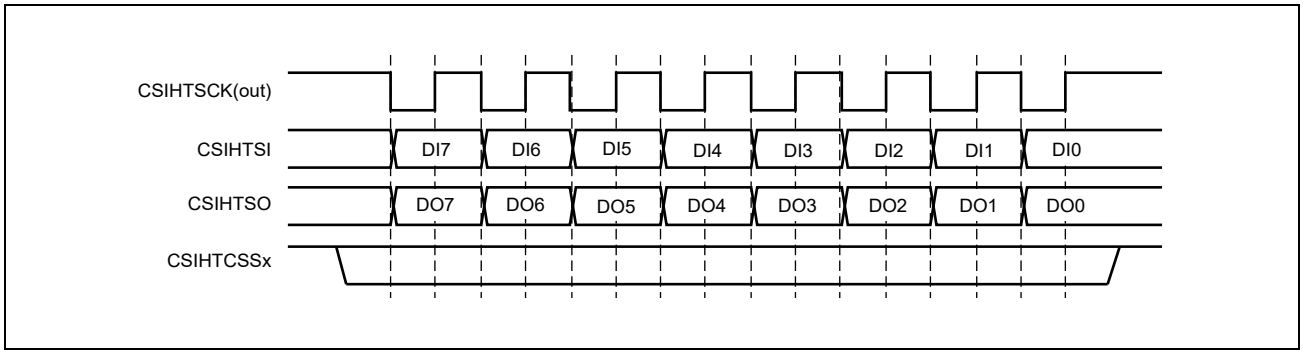


Figure 11.9 Transmission/Reception in Master Mode

11.4.7.2 Slave mode

In slave mode, another device is the communication master and provides the transmission clock. Send/receive operation normally starts as soon as a clock signal is detected.

Slave mode is selected by setting the CSIHnCTL2.CSIHnPRS[2:0] bits to 111_B.

In slave mode, the transmission protocol setting of the CSIHnCFG0 register is enabled (setting of the CSIHnCFG1 to CSIHnCFG3 registers is disabled).

- CSIHnPSx[1:0]: parity usage
- CSIHnDLSx[3:0]: data length selection
- CSIHnDIRx: data direction
- CSIHnCKPx, CSIHnDAPx: clock and data phase

NOTE

When using slave mode, disable the baud rate generator (BRG) by setting the CSIHnBRSy.CSIHnBRS[11:0] bits to 000_H. However, if you are using a time-out error, set the CSIHnBRSy.CSIHnBRS[11:0] bits to a value other than 000_H.

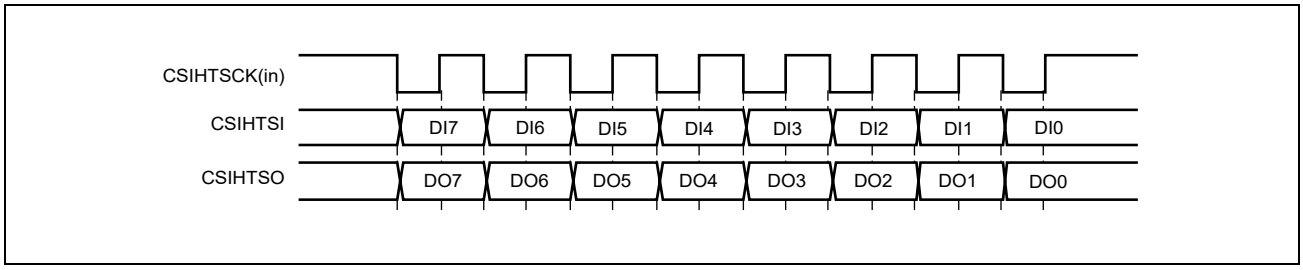


Figure 11.10 Transmission/Reception in Slave Mode

11.4.8 Master/Slave Connections

11.4.8.1 One master and one slave

The following figure illustrates the connections between one master and one slave.

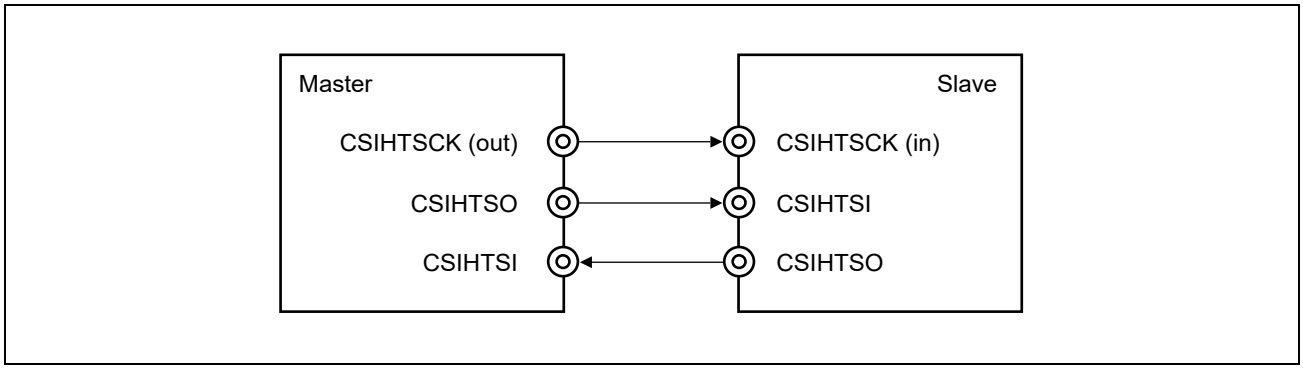


Figure 11.11 Direct Master/Slave Connection

11.4.8.2 One master and multiple slaves

The following figure illustrates the connections between one master and multiple slaves. In this example, the master provides one chip select (CS) signal to each of the slaves. This signal is connected to the slave select input $\overline{\text{CSIHTSSI}}$ of the slave.

The $\overline{\text{CSIHTSSI}}$ signal can be enabled or disabled by using the CSIHnCTL1.CSIHnSSE bit.

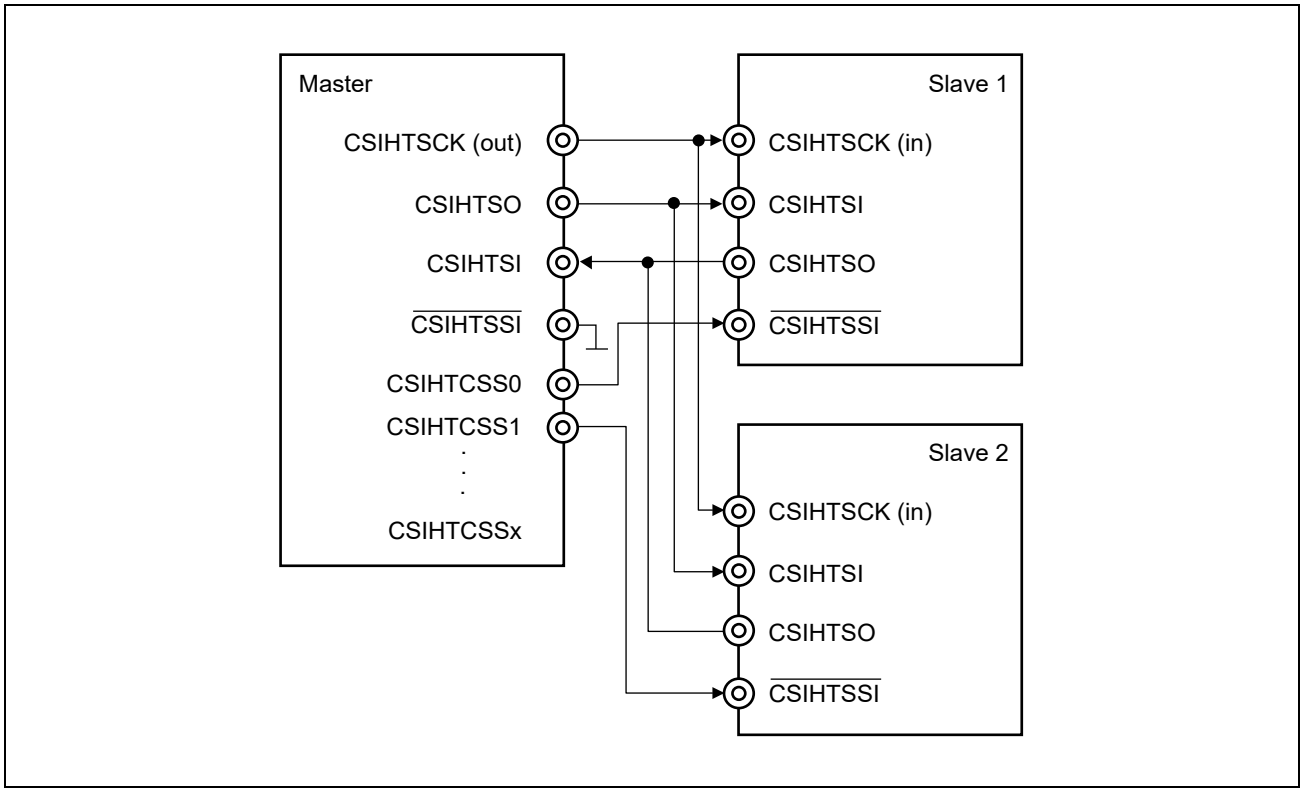


Figure 11.12 Master to Multiple Slaves Connection

By default, the chip select level is active low. That means, a slave is selected (enabled) as a CSIH slave when its $\overline{\text{CSIHTSSI}}$ signal has low level. However, to adapt the CS to other devices, the output level of each chip select signal can also be programmed to be active high.

If a slave is not selected, it will neither receive nor transmit data. In addition, its output CSIHTSO of a slave that is not selected is set to input mode in order to avoid interference with the output of another slave that was selected, in transmit-only mode or transmit/receive mode (CSIHnCTL0.CSIHnTXE = 1).

11.4.9 Chip Selection (CS) Features

The chip select signal, CSIHTCSSx can be used by the master to select one or several slaves for communication.

11.4.9.1 Configuration Registers

The parameters for each chip select signal CSIHTCSSx are defined in the corresponding configuration register CSIHnCFGx. The parameters include the communication protocol and additional CS parameters.

The communication protocol specifies:

- Data length: The number of bits to be sent or received. (CSIHnCFGx.CSIHnDLSx[3:0])
- Transfer direction: MSB or LSB first. (CSIHnCFGx.CSIHnDIRx)
- Parity usage: Odd, even, 0 parity or none. (CSIHnCFGx.CSIHnPSx[1:0])
- Clock phase and data phase: (CSIHnCFGx.CSIHnCKPx, CSIHnCFGx.CSIHnDAPx)

Additional parameters for each chip select signal that is only available in master mode are:

- Prescaler selection of the baud-rate generator separately for each chip select signal (CSIHnCFGx.CSIHnBRSSx[1:0])
- Chip select priority: Separates between “dominant” and “recessive” chip select signals. The priority applies if two or more chip selects signals with different configurations are simultaneously activated for message broadcasting. In this case, the configuration that is set as dominant is used. (CSIHnCFGx.CSIHnRCBx)

The principle is also called “Recessive Configuration for Broadcasting” (RCB).

CAUTION

It is forbidden to specify several chip select signals as dominant with different configurations unless all dominant chip selects have the same configuration.

- Chip select timing:
  - Setup time T_{setup}: The time from setting the CS signal active to starting data output. (CSIHnCFGx.CSIHnSPx[3:0])
  - Inter-data time T_{inter}: The time between one data and the next following data while the CS signal is active. (CSIHnCFGx.CSIHnINx[3:0])
  - Hold time T_{hold}: Hold time of CS signal active level before changing the CS signal. (CSIHnCFGx.CSIHnHDx[3:0])
  - Idle time T_{idle}: Inactive time after every data transfer is completed. (CSIHnCFGx.CSIHnIDx[2:0])

The CS timings of the setup time, the inter-data time, the hold time, and the idle time are illustrated in the figure below. When the CSIHnCFGx.CSIHnIDLx bit set to 1, idle time is inserted per transfer regardless of the CS signal.

Figure 11.13 provides an example when the default CSIHTCSS1 and CSIHTCSS2 signals are active low (CSIHnCTL1.CSIHnCSL1 bit = 0, CSIHnCTL1.CSIHnCSL2 bit = 0). The active level can be specified individually for each CS.

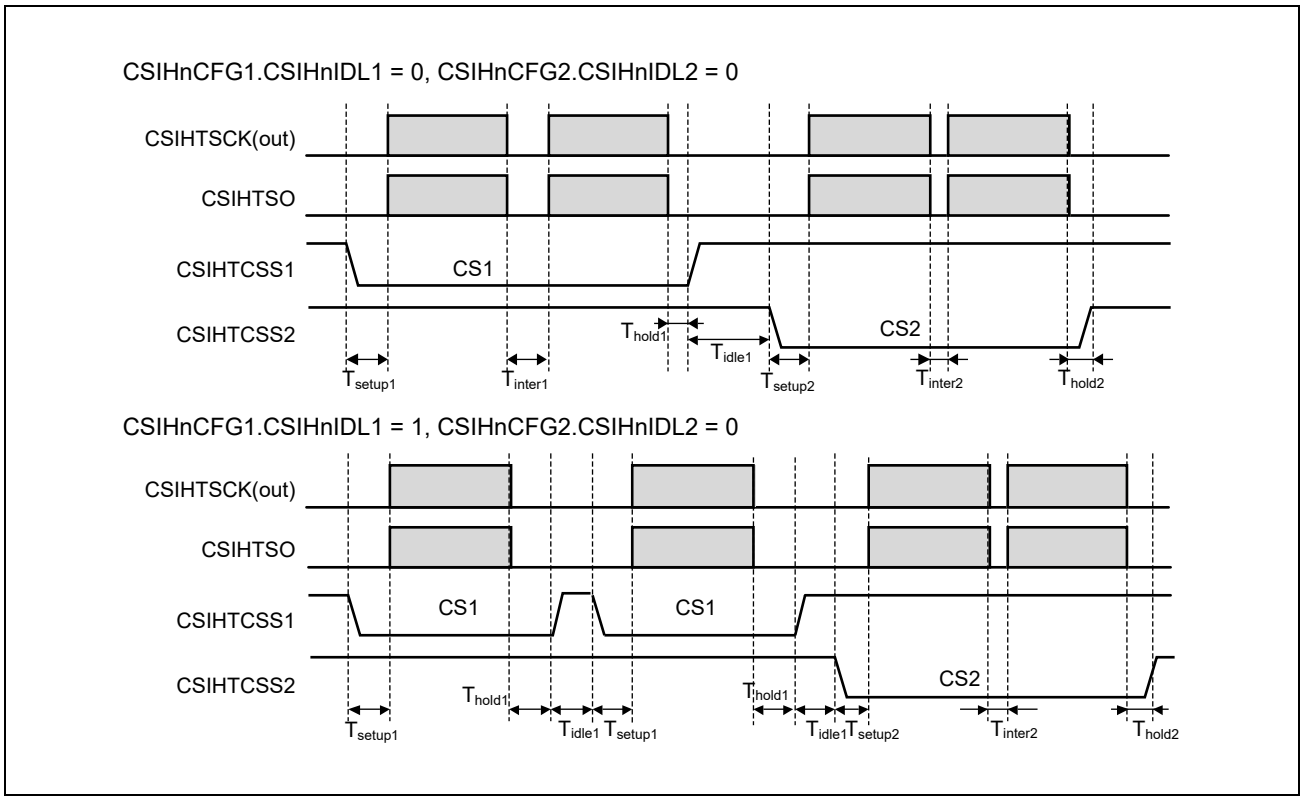


Figure 11.13 Chip Select Timings

Note that each CS signal can have a different value for setup, inter-data time, hold time, and idle time.

A particular chip select signal is activated by setting the appropriate bit in the transmission register CSIHnTX0W.CSIHnCSx.

CSIHnRX0W.CSIHnCSx in the reception register indicates the chip select signal associated with the received data.

CAUTION

If high priority communication function controlled by CPU is enabled (CSIHnCTL1.CSIHnPHE = 1), when mode is switched from low priority communication mode to high priority communication mode or from high priority communication mode to low priority communication mode, idle state is inserted regardless of the setting of the CSIHnCFGx.CSIHnIDLx bit.

11.4.9.2 CS Example

The following figure shows an example of two consecutive data transmissions.

The first communication uses CS0 to communicate with one single slave. The second enables CS0 and CS1 to broadcast a message to two slaves. The priority of CS0 is set to “recessive: low priority”, the priority of CS1 to “dominant: high priority”. Therefore, the second communication uses CS1 set in dominant to communicate.

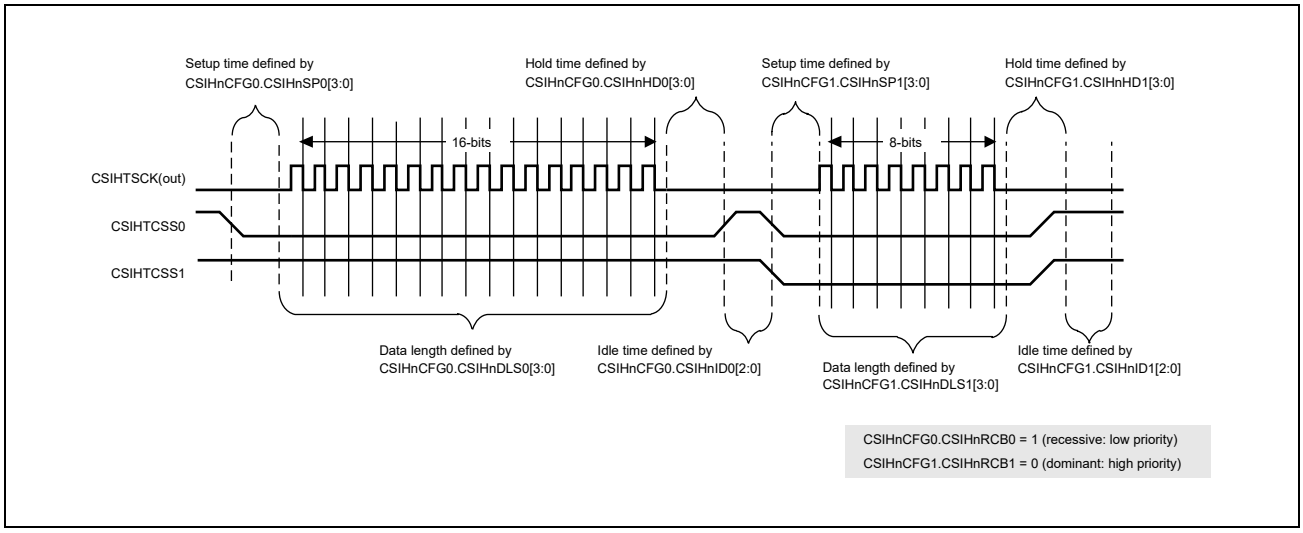


Figure 11.14 Chip Select and RCB Example

11.4.9.3 Job Concept

In terms of CSIH, a job consists of a number of data that are transferred.

Enabling job mode

The job mode can only be enabled in master mode. The job mode is enabled and disabled by CSIHnCTL1.CSIHnJE, while the CSIH is disabled by CSIHnCTL0.CSIHnPWR = 0.

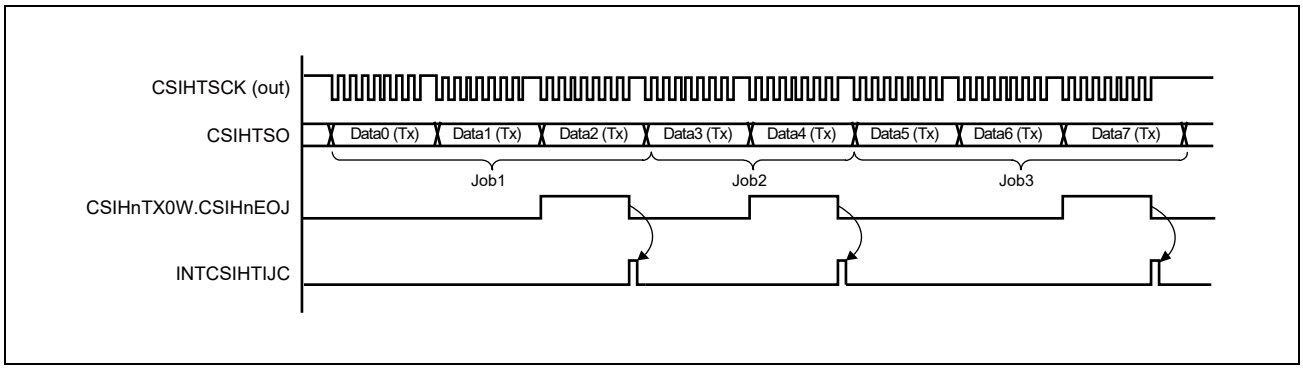


Figure 11.15 Job Examples

A job ends when data set with CSIHnTX0W.CSIHnEOJ = 1 is transmitted.

A communication stop can be specified to occur after a job has finished. This is done by setting CSIHnCTL0.CSIHnJOBE. When CSIHnJOBE is set, the communication continues until data is sent, for which the

CSIHnEOJ bit was set. After this data is sent, the communication is stopped and the interrupt INTCSIHTIJC is generated.

11.4.10 Chip Select Timing Details

11.4.10.1 Changing the Clock Phase

The serial clock level specified by CSIHnCFGx.CSIHnCKPx may be changed when communication is disabled. The minimum value of an idle time is one period of transmission clock (CSIHTSCK (out)).

If the idle time is set to 0.5 transmission clock periods (in CSIHnCFGx.CSIHnIDx[2:0]) and two consecutive data are sent with different CSIHnCFGx.CSIHnCKPx configuration, the idle time is extended to one period of CSIHTSCK (out).

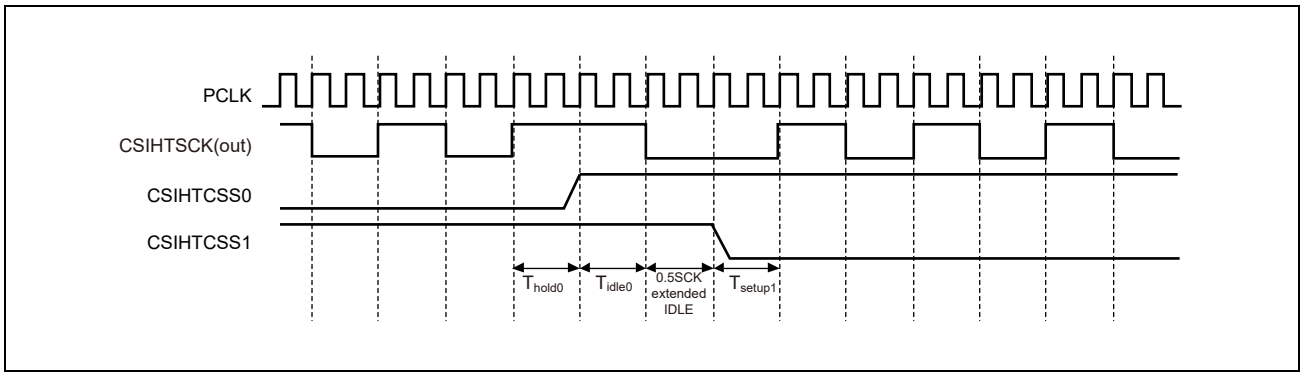


Figure 11.16 Clock Phase Timing
PCLK/4, T_{hold0} = T_{setup1} = 0.5CSIHTSCK, T_{idle0} = 0.5CSIHTSCK, CSIHnCFG0.CSIHnCKP0 = 0 (CSIHTCSS0) → CSIHnCFG1.CSIHnCKP1 = 1 (CSIHTCSS1)

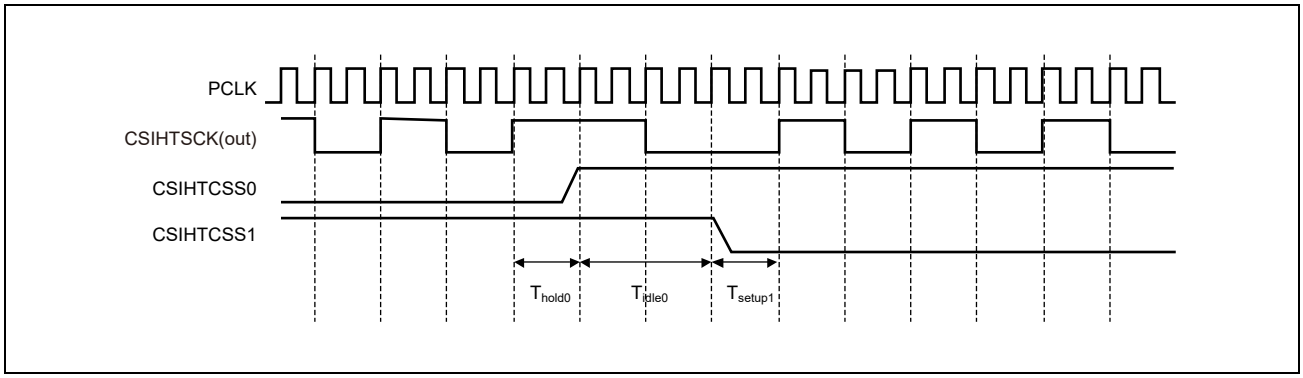


Figure 11.17 Clock Phase Timing
PCLK/4, T_{hold0} = T_{setup1} = 0.5CSIHTSCK, T_{idle0} = 1CSIHTSCK, CSIHnCFG0.CSIHnCKP0 = 0 (CSIHTCSS0) → CSIHnCFG1.CSIHnCKP1 = 1 (CSIHTCSS1)

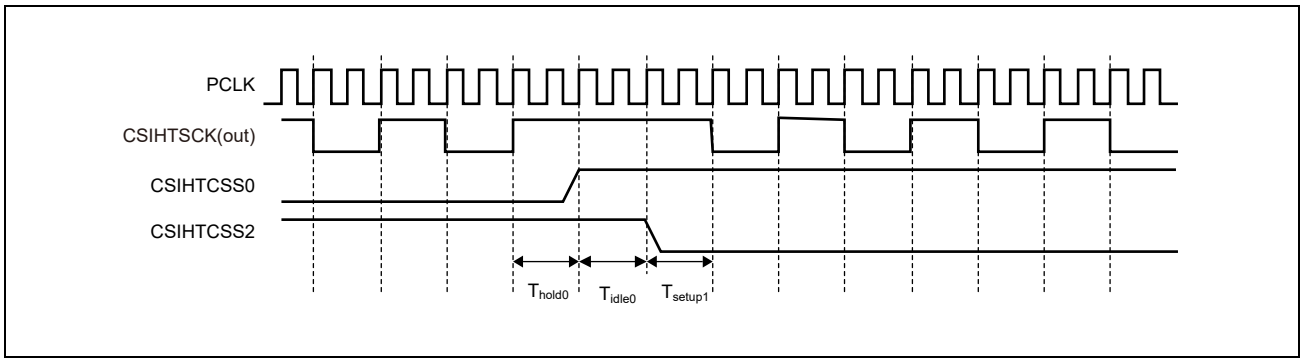


Figure 11.18 Clock Phase Timing
PCLK/4, $T_{hold0} = T_{setup1} = 0.5$CSIHTSCK, $T_{idle0} = 0.5$CSIHTSCK, CSIHnCFG0.CSIHnCKP0 = 0 (CSIHTCSS0) → CSIHnCFG2.CSIHnCKP2 = 0 (CSIHTCSS2)

11.4.10.2 Changing the Data Phase

The CSIHnCFGx.CSIHnDAPx bit defines the phase of the data bits relative to the clock.

The value of the CSIHnCFGx.CSIHnDAPx bit affects the hold and setup times as described below.

The hold time is the time from the last edge of the serial clock (CSIHTSCK) to the point where CSIHTCSSx goes to the inactive level regardless of the CSIHnCFGx.CSIHnDAPx bit setting.

The setup time is the time from the point where CSIHTCSSx goes to the active level until output of the data for transmission (CSIHTSO).

Therefore, timing of the output of the next edge of the serial clock (CSIHTSCK) varies by 0.5 CSIHTSCK according to the setting of CSIHnCFGx.CSIHnDAPx.

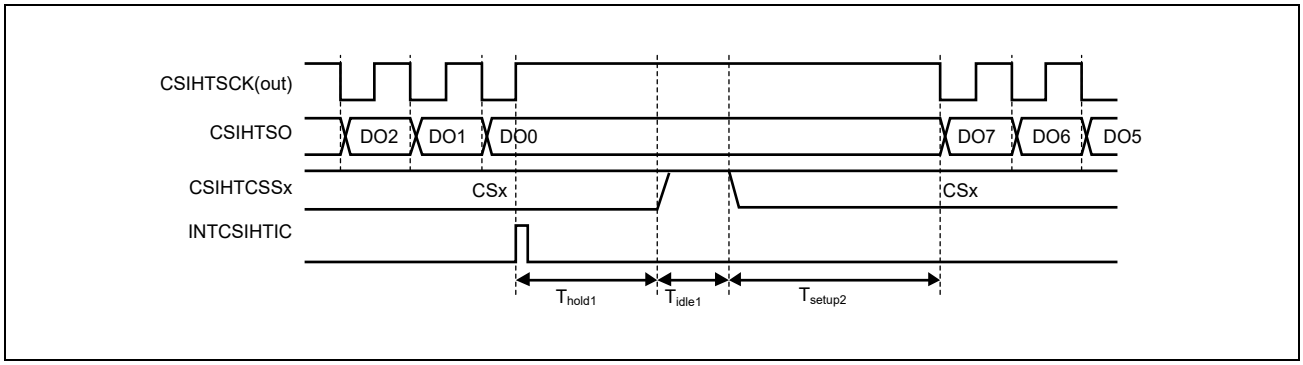


Figure 11.19 Data Phase Timing with
CSIHnCFG1.CSIHnCKP1 = 0, CSIHnCFG1.CSIHnDAP1 = 0 and
CSIHnCFG2.CSIHnCKP2 = 0, CSIHnCFG2.CSIHnDAP2 = 0

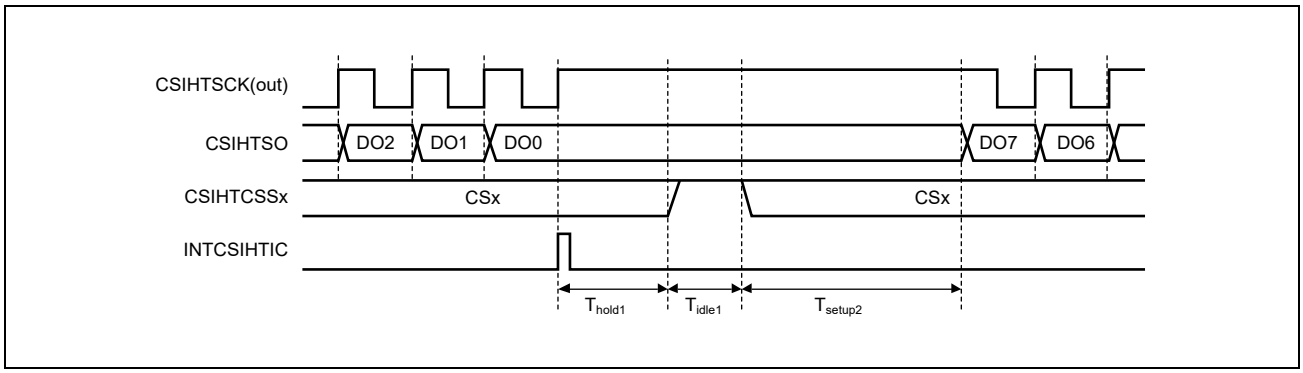


Figure 11.20 Data Phase Timing with
CSIHnCFG1.CSIHnCKP1 = 1, CSIHnCFG1.CSIHnDAP1 = 0 and
CSIHnCFG2.CSIHnCKP2 = 0, CSIHnCFG2.CSIHnDAP2 = 1

11.4.11 Transmission clock selection

In master mode, the transfer clock frequency is selectable using the following bits:

- CSIHnCTL2.CSIHnPRS[2:0]
- CSIHnBRSy.CSIHnBRS[11:0]
- CSIHnCFGx.CSIHnBRSSx[1:0]

The transfer clock frequency of transmission clock CSIHTSCK is determined by the setting of the CSIHnCTL2.CSIHnPRS[2:0] bits and the setting of the CSIHnBRSy.CSIHnBRS[11:0] bits, but any one of CSIHnBRS3 to CSIHnBRS0 can be selected for each chip select signal with the CSIHnCFGx.CSIHnBRSSx[1:0] bits.

The following figure shows a block diagram of the baud-rate generator.

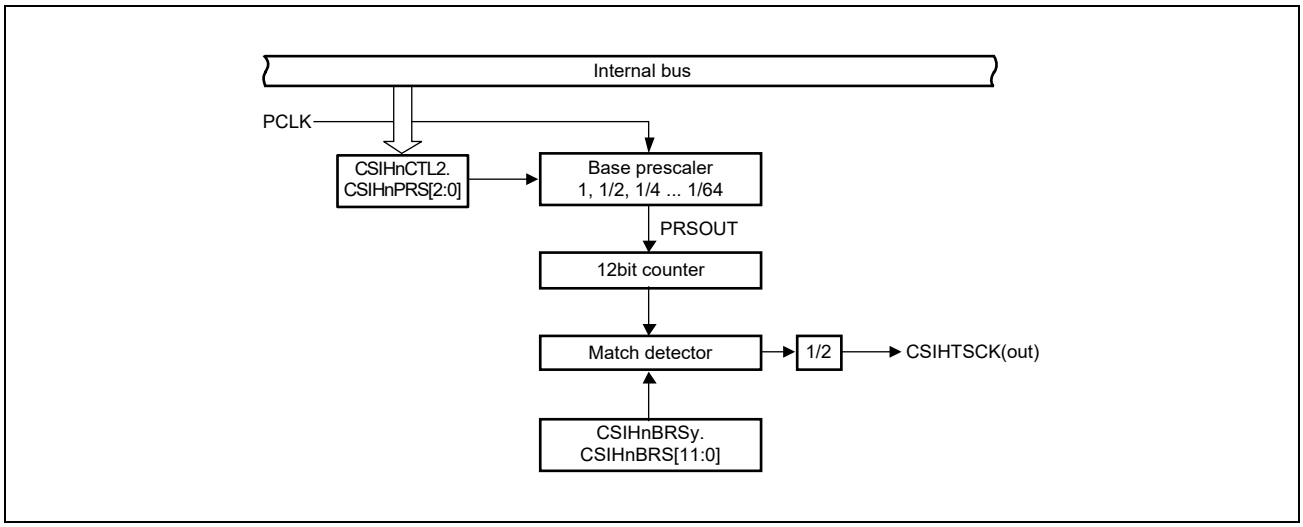


Figure 11.21 Baud Rate Generator Block Diagram

Setting CSIHnBRSy.CSIHnBRS[11:0] to 000_H disables the baud-rate generator, and thus all CSIHTSCK are stopped.

Transfer clock frequency calculation

The transfer clock frequency in master mode is calculated as:

Transfer clock frequency (CSIHTSCK) = PCLK / (division ratio of PCLK) = PCLK / ($2^{\alpha} \times k \times 2$),

where

α = CSIHnCTL2.CSIHnPRS[2:0] = 0 to 6

k = CSIHnBRS0.CSIHnBRS0[11:0] = 1 to 4095 (when CSIHnCFGx.CSIHnBRSSx[1:0] = 0)
CSIHnBRS1.CSIHnBRS1[11:0] = 1 to 4095 (when CSIHnCFGx.CSIHnBRSSx[1:0] = 1)
CSIHnBRS2.CSIHnBRS2[11:0] = 1 to 4095 (when CSIHnCFGx.CSIHnBRSSx[1:0] = 2)
CSIHnBRS3.CSIHnBRS3[11:0] = 1 to 4095 (when CSIHnCFGx.CSIHnBRSSx[1:0] = 3)

Transfer clock frequency upper and lower limits

When setting the transfer clock frequency, please note:

- The minimum transfer clock frequency in both master and slave mode is PCLK / 524160.
- The maximum transfer clock frequency is as below:
  - In master mode: PCLK / 8
  - In slave mode: PCLK / 16

11.4.12 CSIH Buffer Memory

The CSIH has a configurable RAM that can be used for buffered I/O. The size is 128 words. One word is comprised of 32 bits data plus 7 bits ECC.

The following configurations are available:

Mode	CSIHnCTL0.CSIHnMBS	CSIHnMCTL0.CSIHnMMS[1:0]
FIFO mode	0	00_B
Dual buffer mode		01_B
Transmit-only buffer mode		10_B
Direct access mode	1	X

11.4.12.1 FIFO mode

In FIFO mode, data can be written to the CSIHnTX0W register without waiting for completion of the transmission, and data can be received without reading the CSIHnRX0W register immediately, provided the FIFO is not full.

Data to be transmitted is stored to the FIFO memory. Transmission and reception occur simultaneously – one data is sent, one data is received. That means, received data overwrites the transmitted data in the FIFO.

The CSIH automatically updates the respective FIFO memory pointers when a data is rewritten to, read from, sent to, or received from the FIFO memory:

Table 11.36 FIFO Mode

Pointer Description	Control Bit*1	Range
Number of unsent words	CSIHnSTR0.CSIHnSPF[7:0]	0 to 128
Number of words received and stored in the FIFO	CSIHnSTR0.CSIHnSRP[7:0]	0 to 128
Address of data to be written/read	CSIHnMRWP0.CSIHnTRWA[6:0]	0000_H to $01FC_H$
Address of received data to be read	CSIHnMRWP0.CSIHnRRA[6:0]	0000_H to $01FC_H$
Address to be sent	CSIHnMCTL2.CSIHnSOP[6:0]	0000_H to $01FC_H$

Note 1. The value is automatically updated after each read/write or data transmit/receive operation.

The CSIH status register contains also two FIFO status flags:

- CSIHnSTR0.CSIHnFLF: FIFO full
- CSIHnSTR0.CSIHnEMF: FIFO empty

When this mode is started, bit CSIHnSTCR0.CSIHnPCT must be set. Only CSIHnSTR0.CSIHnEMF is set, but not reset.

All FIFO pointers and FIFO flags excluding CSIHnSTR0.CSIHnEMF are reset and CSIHnSTR0.CSIHnEMF is set.

11.4.12.2 Dual buffer mode

In this mode, the memory is divided into two parts of equal size – this means 64 words for transmit data and 64 words for received data. In dual buffer mode, the respective buffer pointers indicate the values shown in the following table.

Table 11.37 Dual Buffer Mode

Pointer Description	Pointer*1	Range
Address for data written to or read from the transmit buffer	CSIHnMRWP0.CSIHnTRWA[6:0]	0000_H to $00FC_H$
Address of data read from the receive buffer	CSIHnMRWP0.CSIHnRRA[6:0]	0000_H to $00FC_H$
The number of transmit data remained in the transmit buffer	CSIHnMCTL2.CSIHnND[7:0]	0 to 64
Address to which data is sent to	CSIHnMCTL2.CSIHnSOP[6:0]	0000_H to $00FC_H$

Note 1. Both pointers are automatically incremented after each read/write.

11.4.12.3 Transmit-only buffer mode

In this mode, the entire memory is used to save transmission data.

Received data must be read directly from CSIHnRX0W/H.

In transmit-only buffer mode, the respective buffer pointer indicates the values shown in the following table.

Table 11.38 Transmit-only Buffer Mode

Pointer Description	Pointer*1	Range
Address for data written to or read from the transmit buffer	CSIHnMRWP0.CSIHnTRWA[6:0]	0000_H to $01FC_H$
The number of transmit data remained in the transmit buffer	CSIHnMCTL2.CSIHnND[7:0]	0 to 128
Address to which data is sent to	CSIHnMCTL2.CSIHnSOP[6:0]	0000_H to $01FC_H$

Note 1. The pointers are automatically incremented after each read/write.

11.4.12.4 Direct access mode

In direct access mode, the CSIH memory is completely bypassed:

- Transmission data provided by the CPU to the transmission register CSIHnTX0W or CSIHnTX0H is directly copied to the shift register.
- Reception data is directly copied from the shift register to the reception register CSIHnRX0W or CSIHnRX0H.

11.4.13 Data Transfer Modes

11.4.13.1 Transmit-only mode

Setting CSIHnCTL0.CSIHnTXE = 1 and CSIHnCTL0.CSIHnRXE = 0 puts the CSIH in transmit-only mode. Start of transmission depends on the memory mode:

- In case of FIFO or direct access mode, transmission starts when transmit data is written to the CSIHnTX0W or CSIHnTX0H register.
- In case of dual buffer or transmit-only buffer mode, transmission starts when the CSIHnMCTL2.CSIHnBTST bit is set.

11.4.13.2 Receive-only mode

Setting CSIHnCTL0.CSIHnTXE = 0 and CSIHnCTL0.CSIHnRXE = 1 puts the CSIH in receive-only mode.

In master mode, start of reception depends on the memory mode:

- In case of FIFO or direct access mode, reception starts when dummy data is written in the CSIHnTX0W or CSIHnTX0H register.

In slave mode, reception starts as soon as the transmission clock CSIHTSCK from the master is received. It is not necessary to write data to the CSIHnTX0W or CSIHnTX0H register of the slave.

- In case of dual buffer mode or transmit-only buffer mode, reception starts when the CSIHnMCTL2.CSIHnBTST bit is set.

11.4.13.3 Transmit/receive mode

Setting CSIHnCTL0.CSIHnTXE = 1 and CSIHnCTL0.CSIHnRXE = 1 puts the CSIH in transmit/receive mode.

Start of communication (transmission and reception) depends on the memory mode:

- In case of FIFO or direct access mode, communication starts when transmit data is written to the CSIHnTX0W or CSIHnTX0H register.
- In case of dual buffer or transmit-only buffer mode, communication starts when the CSIHnMCTL2.CSIHnBTST bit is set.

11.4.13.4 Summary

The following table summarizes this section. It shows how data transfer is started in the various memory, operating, and transfer modes.

Table 11.39 Start of Data Transfer

Memory and Operating Mode		Transfer Mode	
		Transmit-Only Transmit/Receive	Receive-Only
FIFO, direct access	Master	Writing to the CSIHnTX0W register or the CSIHnTX0H register	Writing to the CSIHnTX0W register or the CSIHnTX0H register
	Slave	Reception of clock from the master	Incoming clock from the master
Transmit-only buffer, dual buffer	Master	CSIHnMCTL2.CSIHnBTST = 1	CSIHnMCTL2.CSIHnBTST = 1
	Slave	Reception of clock from the master	Incoming clock from the master

11.4.14 Data Length Selection

11.4.14.1 Data length between 2 and 16 bits

The length of a data packet is selectable for each chip select signal from 2 to 16 bits by using CSIHnCFGx.CSIHnDLSx[3:0]. The examples below show the communication with MSB first (CSIHnCFGx.CSIHnDIRx = 0).

Data length = 16 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 0000_B)

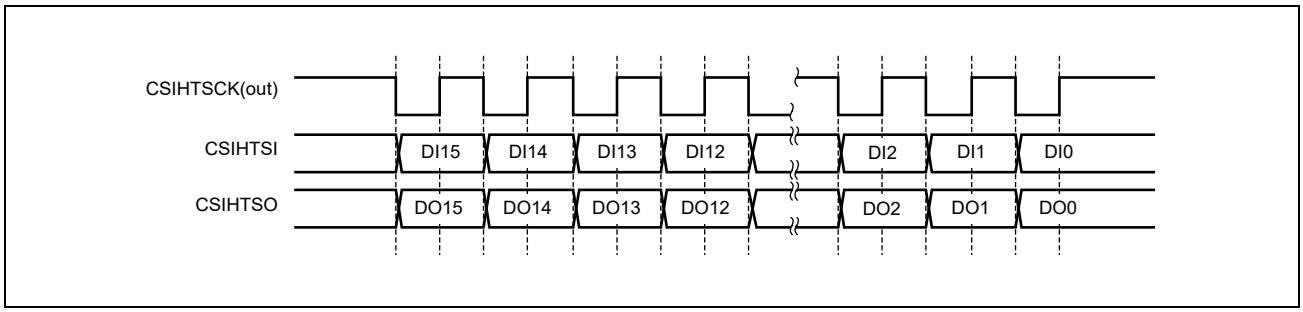


Figure 11.22 16 Bit Data Length, MSB First

Data length = 14 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1110_B)

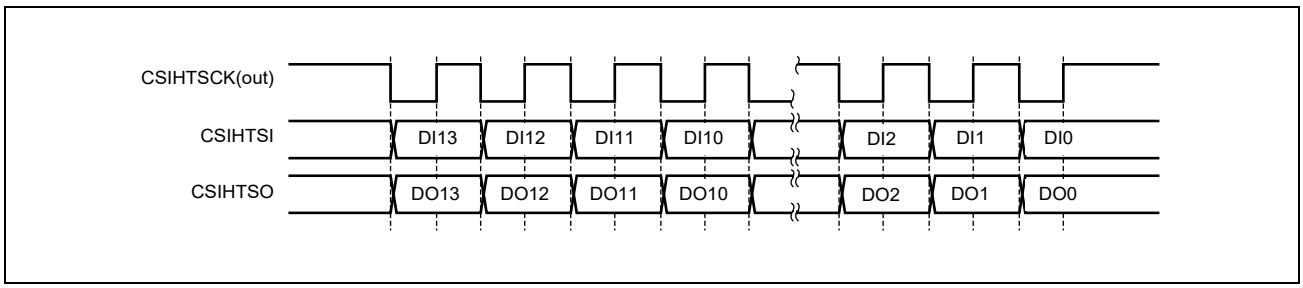


Figure 11.23 14 Bit Data Length, MSB First

11.4.14.2 Data length greater than 16 bits

If the data to be sent/received exceeds 16 bits, the extended data length (EDL) feature can be used.

EDL function is enabled by setting the CSIHnCTL1.CSIHnEDLE bit to 1.

EDL function works as follows:

- The data has to be broken into 16-bit blocks plus remainder. For example, 42-bit data would be broken into two 16-bit blocks plus 10 bits.
- The bit length of the remainder is set as “data length” in CSIHnCFGx.CSIHnDLSx[3:0].
- For transmitting the 16-bit blocks, CSIHnTX0W.CSIHnEDL must be set to 1. In this case, the data written to CSIHnTX0W is sent as a 16-bit data length regardless of the CSIHnCFGx.CSIHnDLSx[3:0] bit setting.
- The transfer is complete after a block with the specified data length (the remainder of data specified with CSIHnTX0W.CSIHnEDL = 0) has been sent.

Example

Example for sending 40-bit data ($123456789A_H$) to CS0:

40 bits are split into 2 to 16 bits plus 8 bits.

- Initialize CSIHnCFG0.CSIHnDLS0[3:0] = 8.
- To send $123456789A_H$with MSB first, write the following sequence to CSIHnTX0W:
  - $20FE\ 1234_H$ (CSIHnTX0W.CSIHnEDL = 1)
  - $20FE\ 5678_H$ (CSIHnTX0W.CSIHnEDL = 1)
  - $00FE\ 009A_H$ (CSIHnTX0W.CSIHnEDL = 0)

The following figure illustrates the timing.

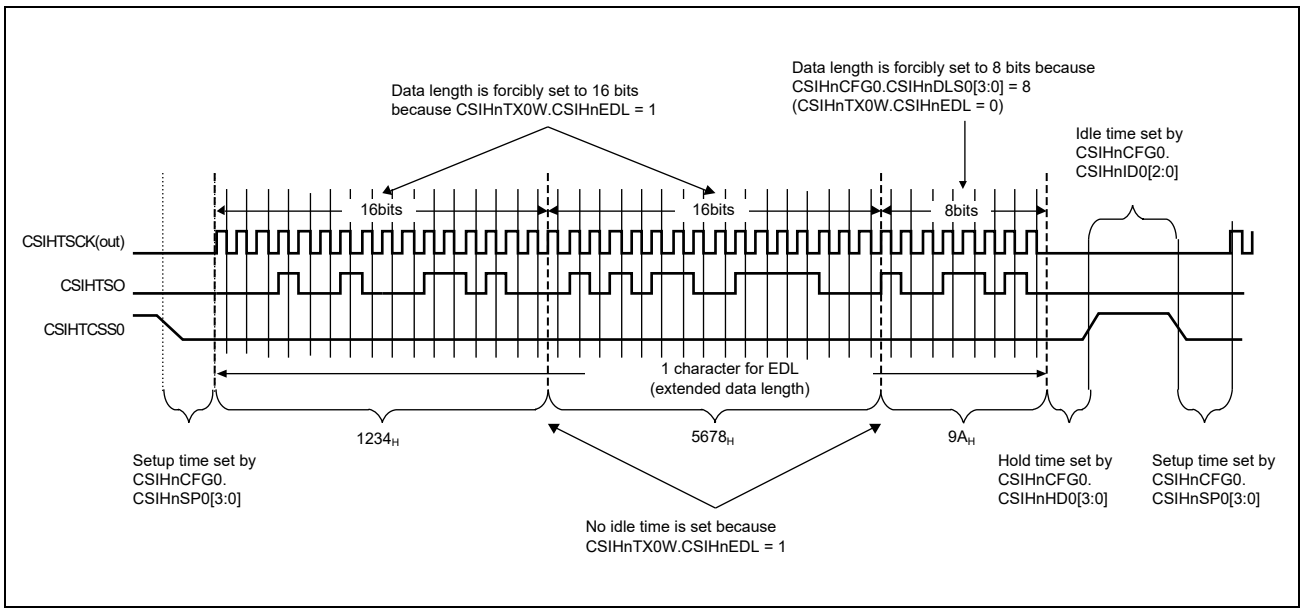


Figure 11.24 EDL Timing Diagram

NOTES

1. A data length of 1 bit can only be selected when EDL mode is in use.
2. It is not possible to send two consecutive data with a data length of less than 2 bits.
3. If parity is enabled, the parity bit is added after the last bit.
4. When extended data length (EDL) function is used, use the same chip selection signal.
5. To consider the data direction, pay attention to the following example:
   - Data to be sent: 12 3456_H
   - MSB first:
     Set CSIHnCFGx.CSIHnDIRx = 0
     Write CSIHnTX0W = 20FE 1234_H(EDL bit = 1)
     Write CSIHnTX0W = 00FE 0056_H(EDL bit = 0)
   - LSB first:
     Set CSIHnCFGx.CSIHnDIRx = 1
     Write CSIHnTX0W = 20FE 3456_H(EDL bit = 1)
     Write CSIHnTX0W = 00FE 0012_H(EDL bit = 0)
6. If CSIHnTX0W.CSIHnEOJ = 1 and CSIHnTX0W.CSIHnEDL = 1 are set simultaneously while CSIHnCTL1.CSIHnJE = 1 and CSIHnCTL1.CSIHnEDLE = 1, the behavior and performance are not guaranteed.
7. EDL mode cannot be used in receive-only mode of slave mode.(CSIHnCTL2.CSIHnPRS[2:0] = 111_B, CSIHnCTL0.CSIHnTXE = 0, CSIHnCTL0.CSIHnRXE = 1)

11.4.15 Serial Data Direction Selection

The serial data direction is selectable for each chip select signal by using the CSIHnDIRx bit in the CSIHnCFGx register.

The examples below show communication for a data length of 8 bit (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).

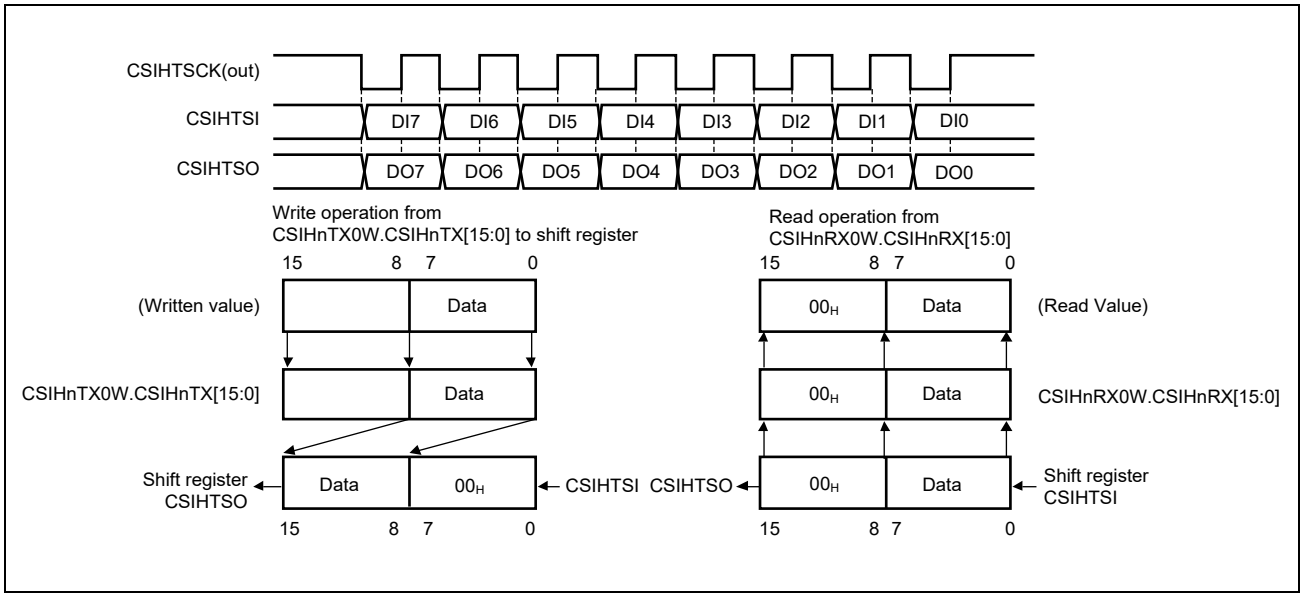


Figure 11.25 Serial Data Direction Select Function - MSB First (CSIHnDIRx = 0)

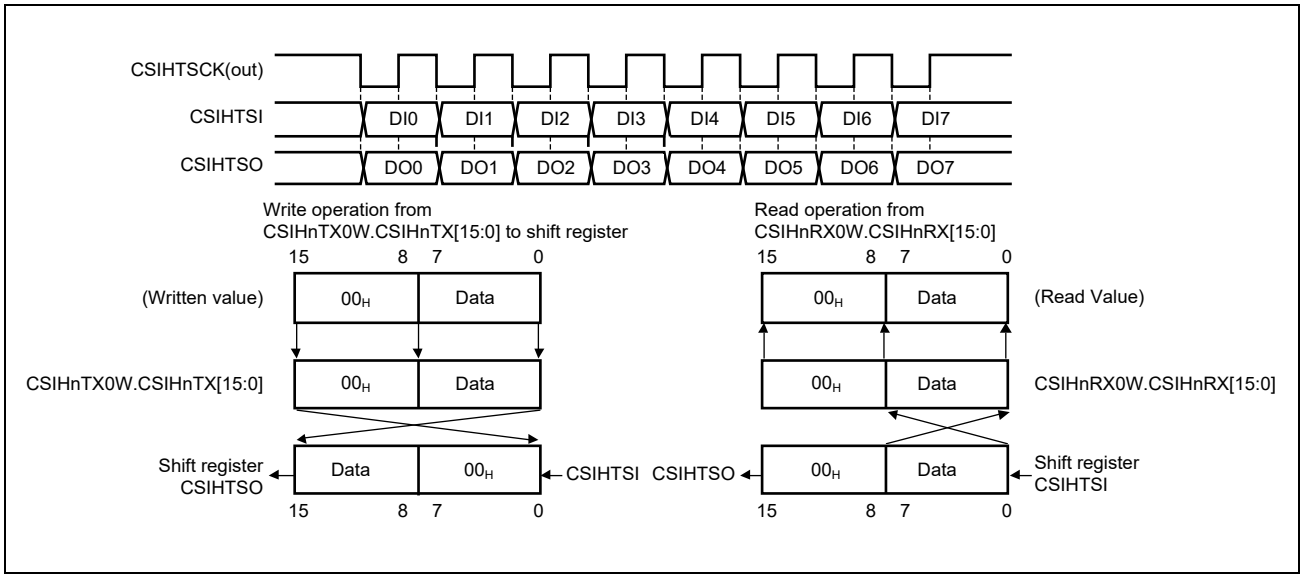


Figure 11.26 Serial Data Direction Select Function - LSB First (CSIHnDIRx = 1)

11.4.16 SS (Slave Select) Function

The SS function realizes communication between one master and multiple slaves.

In master mode, the master device outputs the slave select signal (CSIHTCSSx) to select a single slave. Communication by a device in slave mode is enabled when the slave input select signal ($\overline{\text{CSIHTSSI}}$) is at the low level.

Refer to the **Section 11.4.8, Master/Slave Connections**, for an example of a connection using the SS function.

11.4.16.1 Communication timing using SS function

The following figure illustrates the communication signal using the SS function and timings. In slave mode, the data transfer configuration is determined by the CSIHnCFG0 register.

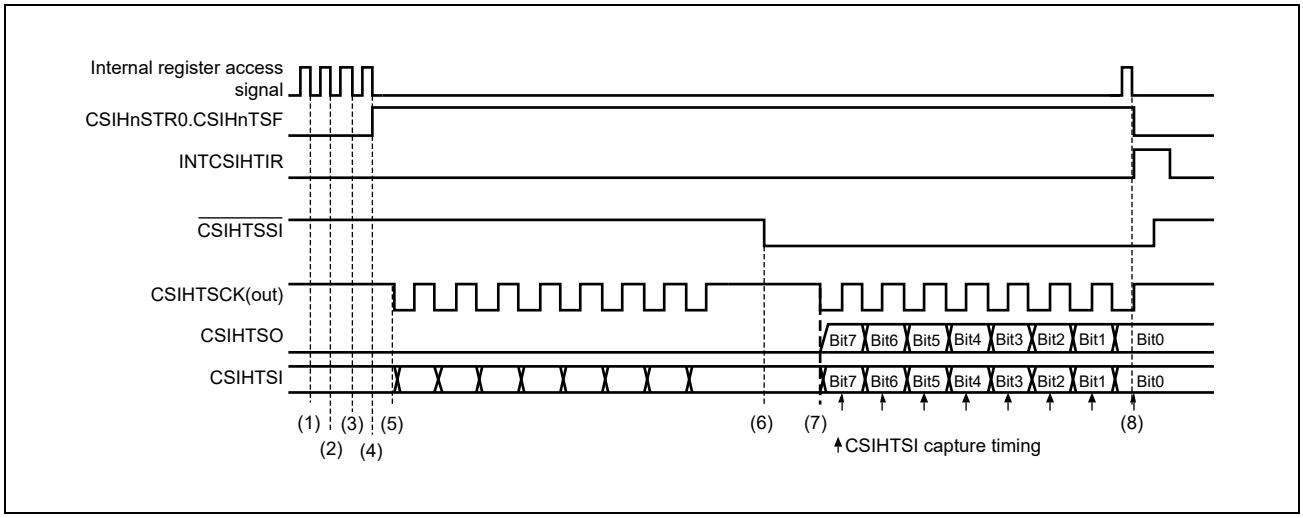


Figure 11.27 Tx/Rx Timing of Communication Using SS Function

(1) CSIH is put into slave mode by setting CSIHnCTL2.CSIHnPRS[2:0] = 111_B. CSIHnCFG0.CSIHnCKP0 and CSIHnCFG0.CSIHnDAP0 are 0.

(2) The data length is 8 bits (CSIHnCFG0.CSIHnDLS0[3:0] = 1000_B). The data direction is MSB first (CSIHnCFG0.CSIHnDIR0 = 0).

(3) The transmit/receive mode is set (CSIHnCTL0.CSIHnTXE = 1, CSIHnCTL0.CSIHnRXE = 1, CSIHnCTL0.CSIHnPWR = 1). Communication start is permitted.

(4) The transfer status flag CSIHnSTR0.CSIHnTSF is automatically set when transfer data is written to the CSIHnTX0W or CSIHnTX0H transmission register during direct access mode or FIFO mode.

(5) As long as signal $\overline{\text{CSIHTSSI}}$ is high, transmission/reception is not started, even if an external transmission clock CSIHTSCK is applied. Input at CSIHTSI is ignored.

(6) When $\overline{\text{CSIHTSSI}}$ falls to low level, indicating that CSIHTSO is enabled, transmission is enabled.

(7) Now, as soon as the external clock signal CSIHTSCK appears, the slave transmits data to CSIHTSO and simultaneously captures data from CSIHTSI.

(8) Interrupt INTCSIHTIR indicates when the reception is complete. The CSIHnRX0W/H register can be read.

11.4.16.2 CSIHTSSO operation

CSIHnPWR	CSIHnTXE	CSIHnRXE	CSIHnSSE	CSIHTSSO
0	—	—	—	H
1	—	—	0	H
	0		1	H
	1		1	Reversed value of $\overline{\text{CSIHTSSI}}$ level

The CSIHTSSO pin is a signal to control the I/O function of the chip's SO pin in case of using the SS function.

The CSIHTSO pin is enabled when the CSIHTSSO pin is "High" (the chip's SO pin is being driven).

The CSIHTSO pin is disabled when the CSIHTSSO pin is "Low" (the chip's SO pin is not being driven).

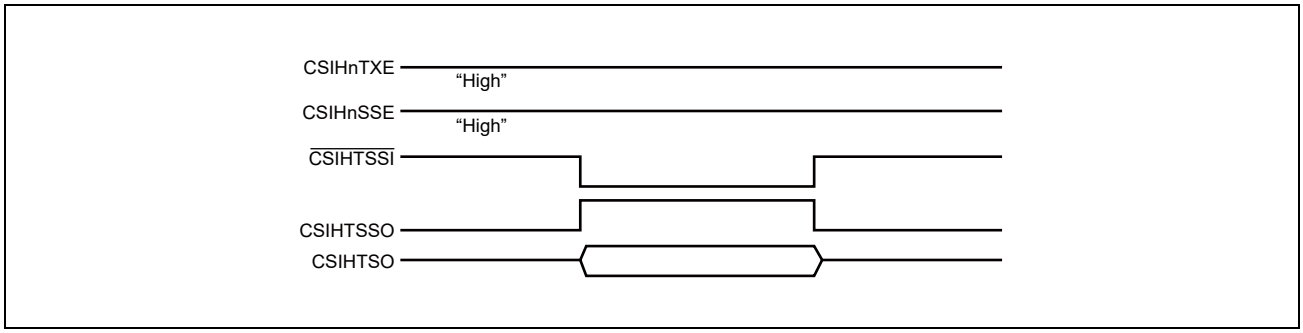


Figure 11.28 Operation of CSIHTSSO

CAUTION

If $\overline{\text{CSIHTSSI}}$ pin is changed during communication (CSIHnSTR0.CSIHnTSF = 1), current communication is not assured.

11.4.17 Handshake Function

CSIH features a handshake function to synchronize the master and the slave devices. This function can be enabled/disabled by the CSIHnCTL1.CSIHnHSE bit. For handshake, the signals CSIHTRYI and CSIHTRYO are used.

The busy timing depends on the data phase selection bit CSIHnCFGx.CSIHnDAPx.

11.4.17.1 Slave mode

When CSIHnCTL1.CSIHnHSE = 1 and the slave is busy, the CSIHTRYO signal outputs low level.

This can happen in two cases:

1. When the next data to be sent is not ready:
   When the slave is in transmit-only mode or transmit/receive mode (CSIHnCTL0.CSIHnTXE = 1) and is in the states listed below, the CSIHTRYO output indicates the busy state (is at the low level).

Table 11.40 Memory Mode and Slave Transfer State

Memory Mode	Transmission State of Slave
Direct access mode	No data for the next transfer
FIFO mode	No data for the next transfer (CSIHnSTR0.CSIHnEMF = 1)
Dual buffer mode	When CSIHnMCTL2.CSIHnBTST is not set to 1
Transmit-only buffer mode	

The example below is on the assumption of an eight-bit data length.

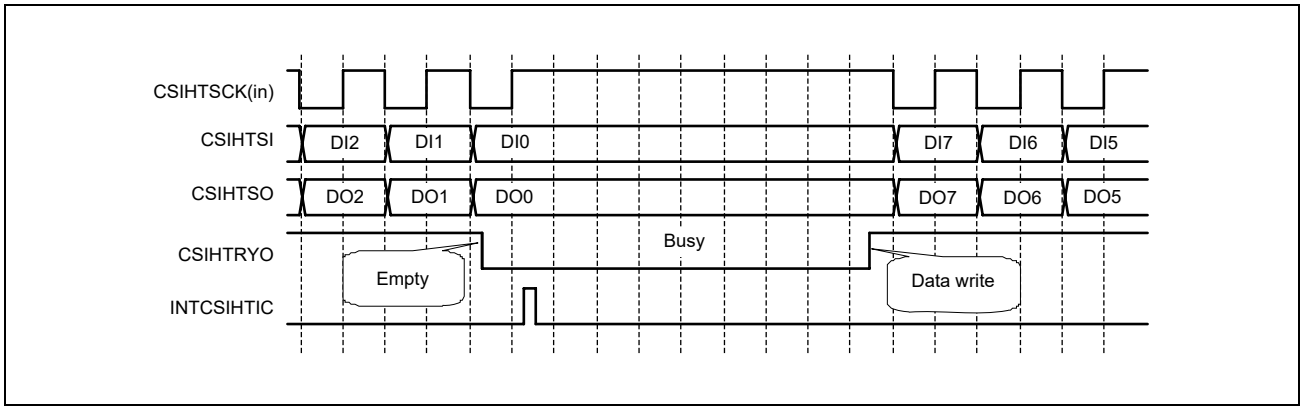


Figure 11.29 Busy Signal from the Slave (FIFO Mode; CSIHnCFGx.CSIHnDAPx = 0)

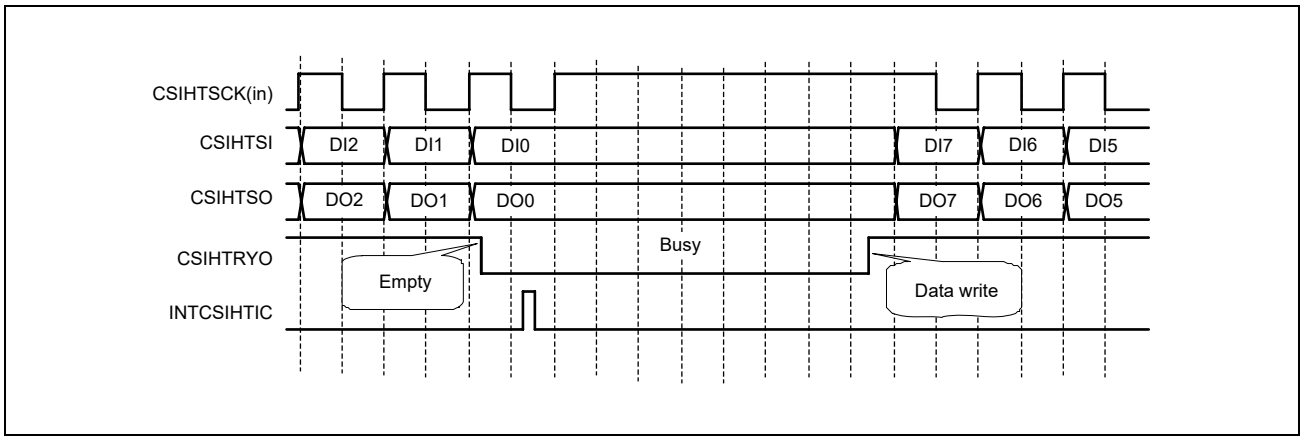


Figure 11.30 Busy Signal from the Slave (FIFO Mode; CSIHnCFGx.CSIHnDAPx = 1)

2. When transmit register is full:
   When slave is set in receive-only mode or transmit/receive mode (CSIHnCTL0.CSIHnRXE = 1), and new data cannot be copied from a shift register to CSIHnRX0W/H (CSIHnRX0W/H is full) because the previously received data is still in the CSIHnRX0W/H register. When CSIHnCTL0.CSIHnRXE is 1 and is in the following states, CSIHTRYO outputs busy state (low level).

Table 11.41 Memory Mode and Slave Transfer State

Memory Mode	Transmission State of Slave
Direct access mode	When CSIHnRX0W or CSIHnRX0H is full
FIFO mode	Receive data remains in the buffer (when CSIHnSTR0.CSIHnFLF = 1)
Dual buffer mode	No applicable state
Transmit-only buffer mode	When CSIHnRX0W or CSIHnRX0H is full

The example below is on the assumption of an eight-bit data length.

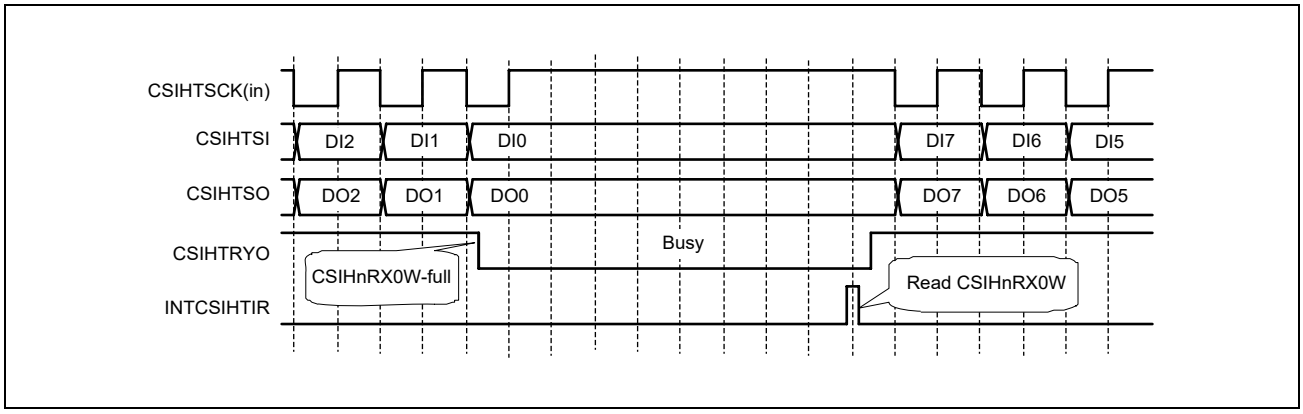


Figure 11.31 Busy Signal from the Slave (Direct Access Mode; CSIHnCFGx.CSIHnDAPx = 0)

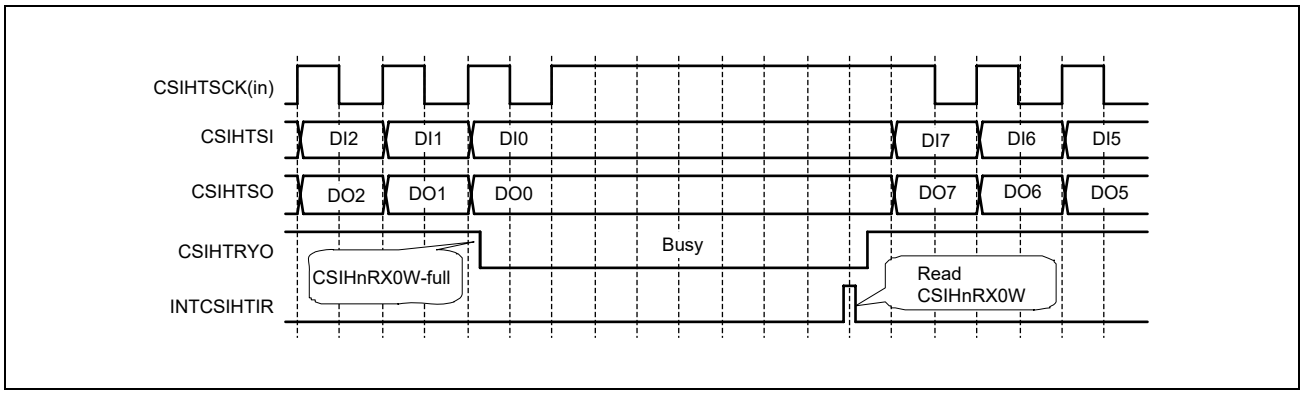


Figure 11.32 Busy Signal from the Slave (CSIHnCFGx.CSIHnDAPx = 1)

11.4.17.2 Master mode

When the master detects CSIHTRYI = 0 while CSIHnCTL1.CSIHnHSE = 1, the following transfer is put on hold, and the master goes into wait status. It suspends the CSIHTSCK clock.

The CSIHTRYI level is checked at each half clock cycle of CSIHTSCK.

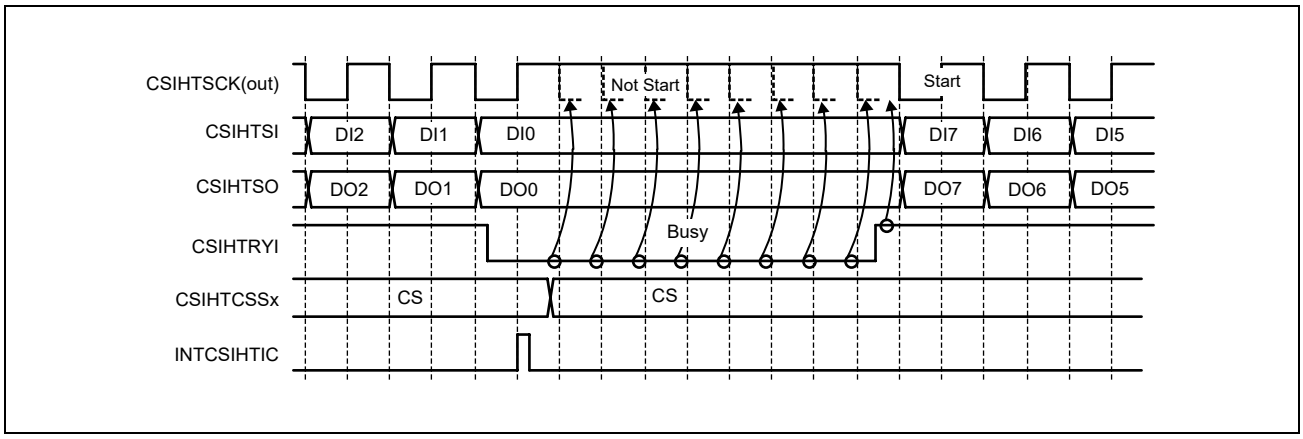


Figure 11.33 Master's Reaction on CSIHTSHSG (CSIHnCFGx.CSIHnDAPx = 0)

The CSIHTRYI signal must be pulled down by the slave before the next transfer starts. If this is done on the slave while data transfer is in progress, the serial clock from the master is suspended after the transfer is complete.

The master resumes the communication as soon as CSIHTRYI becomes high level (the slave is "ready").

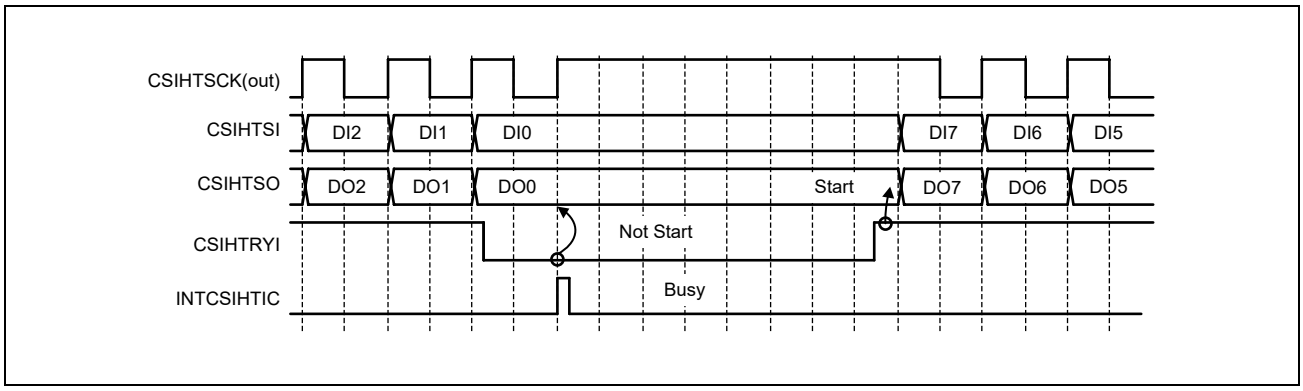


Figure 11.34 Master's Reaction on CSIHTRYI (CSIHnCFGx.CSIHnDAPx = 1)

CAUTIONS

1. If multiple slaves are connected, the master must only detect the CSIHTRYI signal from the slave it has selected for communication.
2. Even when the CSIHTRYI pin of the master detects a CSIHTRYO signal from the slave during data transfer, the communication is not made to wait but continues until the data transfer is completed.

11.4.18 Error Detection

CSIH can detect five error types:

- Data consistency error (transmission data)
- Parity error (received data)
- Overrun error (received data)
- Time-out error (in FIFO mode)
- Overflow error (in FIFO mode)

Check for parity, data consistency and time-out errors can be enabled/disabled individually.

If one of these errors is detected, the interrupt request, INTCSIHTIRE is generated and the corresponding flags are set.

11.4.18.1 Data consistency check

The purpose of the data consistency check is to ensure that the data physically sent as output signal is identical to the original data that was copied to the shift register.

The data consistency check can be enabled/disabled by the CSIHnCTL1.CSIHnDCS bit. When checking data consistency, always set the PIPCn.PIPCn_m to 1. It is not active if data transmission is disabled (CSIHnCTL0.CSIHnTXE = 0).

When the data consistency check is active, the data transferred from CSIHnTX0W or CSIHnTX0H to the shift register is copied to a separate register.

After completion of the transmission, the sent data is compared with the original transmission data.

Mismatch is considered as a data consistency error:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnDCE bit is set.

Additionally, CSIHnRX0W.CSIHnTDCE of data that contains the error is set.

The data consistency check function is illustrated in the following block diagram.

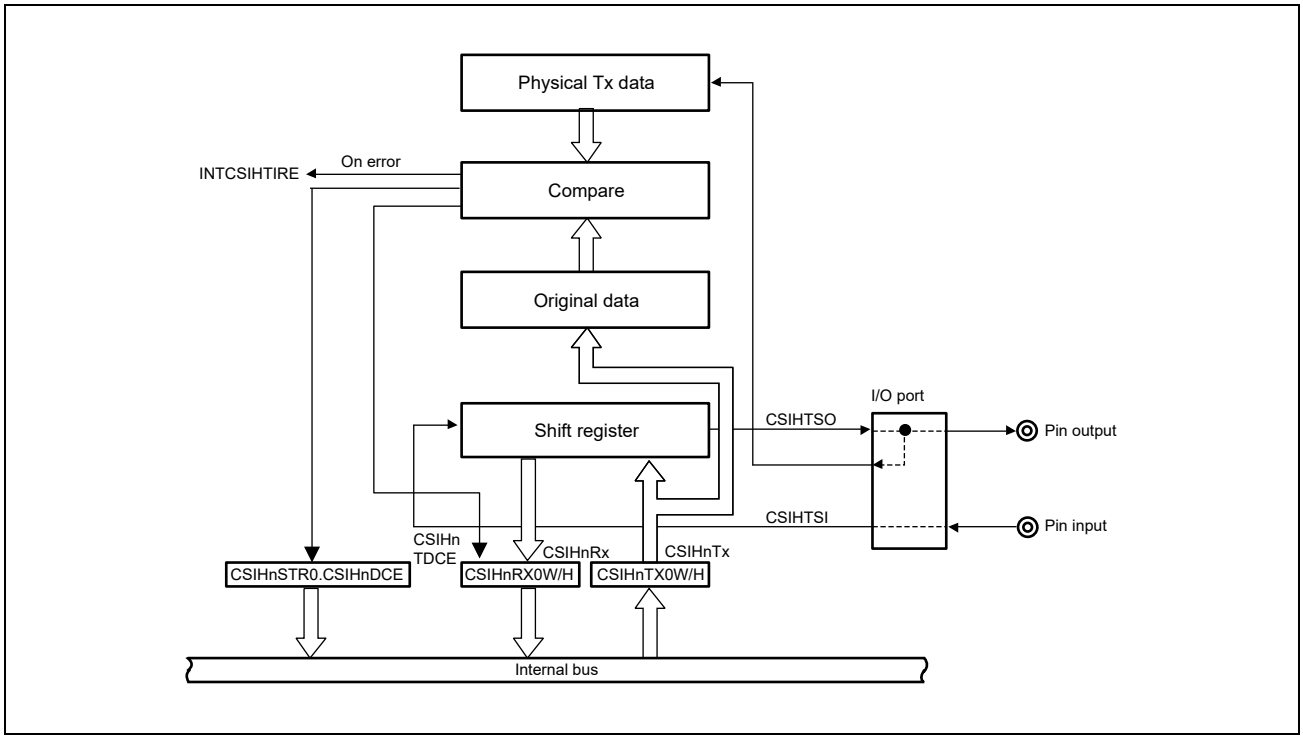


Figure 11.35 Data Consistency Check Functional Block Diagram

11.4.18.2 Parity check

CSIH can append a parity bit to the last data bit (even if extended data length is used). The use and type of parity is specified in CSIHnCFGx.CSIHnPSx[1:0].

Parity check is enabled if CSIHnCFGx.CSIHnPSx[1] = 1.

The parity bit is checked after a reception is complete. In case of parity error:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnPE bit is set.

Additionally, CSIHnRX0W.CSIHnRPE of data that contains the error is set. The figure below shows an example.

- Data length is 8 bits.
- The data to be transmitted is 05_H and 35_H.
- Data direction is LSB first.
- Parity type is odd.

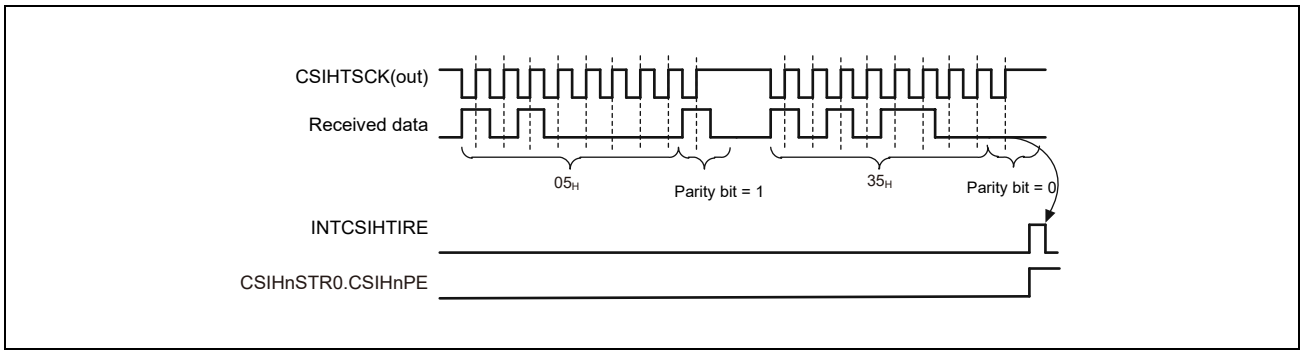


Figure 11.36 Parity Check Example

The parity bit of the first data is 1. There is no parity error, because the total number of ones (including the parity bit) is odd.

The parity bit of the second data is 0. This is detected as a parity error, because the total number of ones (including the parity bit) is even.

If the EDL (extended data length) function is used, the parity bit is added after the last bit of the data.

11.4.18.3 Time-out error

Time-out errors can be checked only in slave FIFO mode.

This error occurs if neither of the following occurred within a certain period of time:

- Received data in FIFO is read
- FIFO receives data from CSIHTSI

The time is defined in CSIHnMCTL0.CSIHnTO[4:0] in multiples of 8 times the transmission clock, CSIHTSCK. A time-out error occurs when the specified time is exceeded (The time-out time is not detected when CSIHnMCTL0.CSIHnTO[4:0] = 00000_B).

A dedicated time-out counter measures the time between the last and the next read operation.

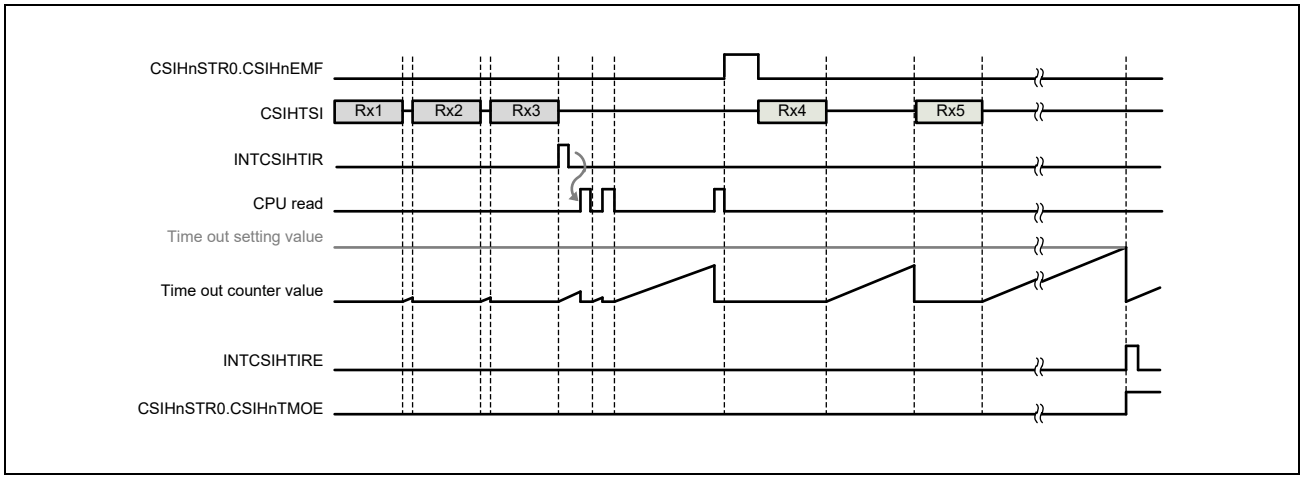


Figure 11.37 Time-Out Check Functional Timing Diagram

The start timing of the time-out counter is as follows:

- When reception is completed
- When data read from the CPU completes (The counter does not start if the buffer is empty.)
- When a time-out error is detected

After a time-out error is detected, if data is still available in FIFO, the time-out counter restarts.

If the value set by bit CSIHnMCTL0.CSIHnTO[4:0] is reached again, the INTCSIHTIRE interrupt is output again.

The timeout counter continues to count until received data is read. To stop the counter, read all received data or set CSIHnSTCR0.CSIHnPCT to 1. Note that the pointer is cleared if you perform the latter.

The counter is reset at the following timing:

- Data is read once.
- New data is received.
- A timeout error is detected.
- The CSIHnSTCR0.CSIHnPCT bit is set to 1.

If a timeout error occurs, the following occurs:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnTMOE bit is set.

The dedicated time-out counter is set by the CSIHnCTL2.CSIHnPRS[2:0] and CSIHnBRSy.CSIHnBRS[11:0] bits. If the value of the CSIHnBRSy.CSIHnBRS[11:0] bits is left as 000_H, the dedicated time-out counter does not operate.

11.4.18.4 Overflow error

An overflow error can happen in FIFO mode. It occurs when transmission data is written to the CSIHnTX0W register while the FIFO buffer is filled with received data.

Example

100 data have been transmitted. That means, the FIFO contains 100 received data. The application starts to read the received data.

While the read operation is in progress, the application begins to write another set of 50 transmission data to the FIFO. However, only 10 received data have been read up to now, 90 are still in the FIFO.

In this case, only 38 cells are available for new transmission data packets. When the CPU tries to write the 39th data, an overflow error happens.

This is illustrated in the following figure.

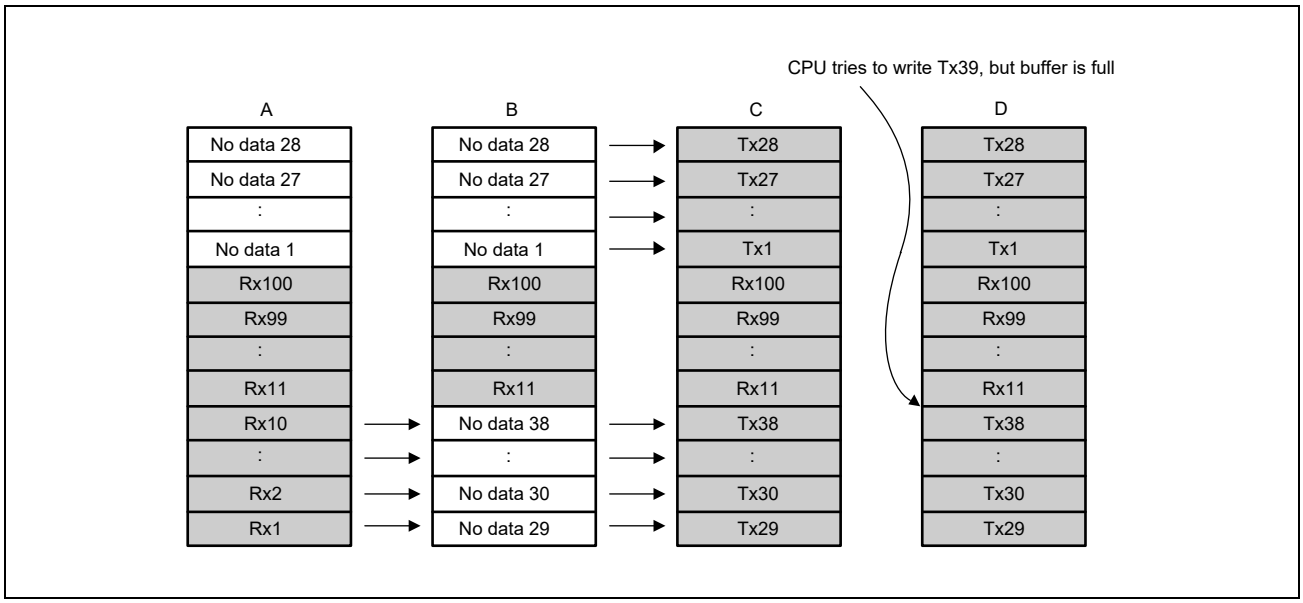


Figure 11.38 FIFO Overview

The data after 39 are discarded. The figure below shows the overflow timing.

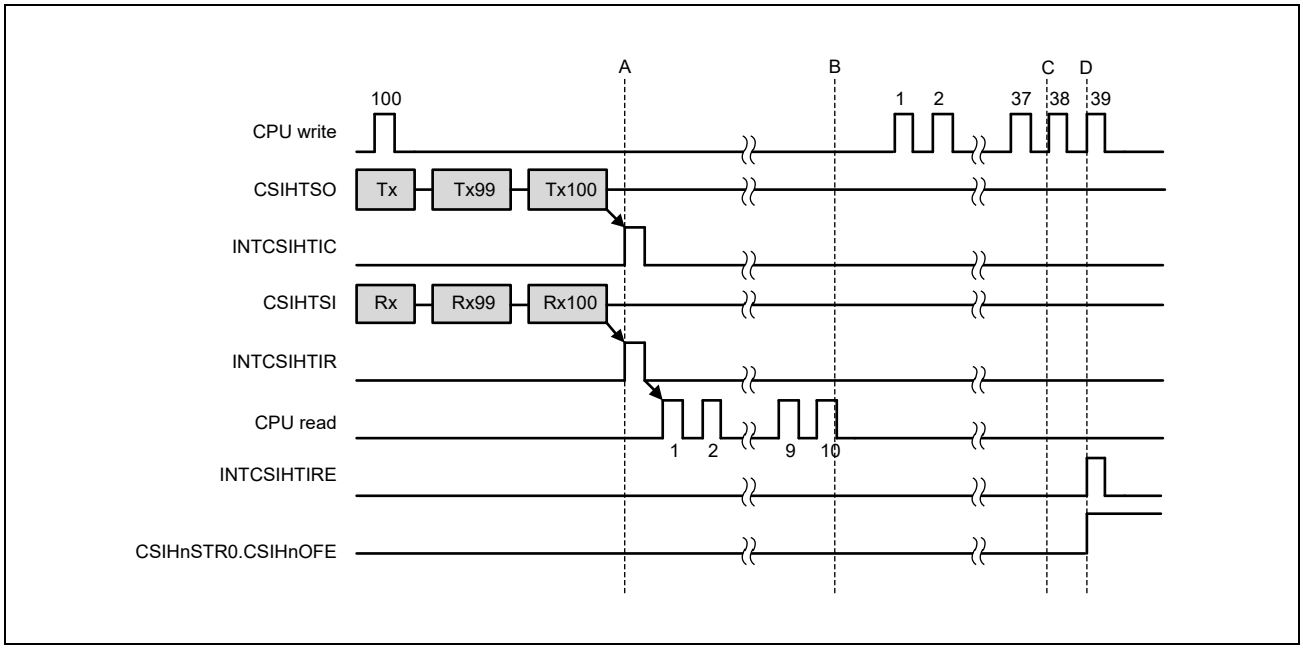


Figure 11.39 FIFO Overflow Timing

In case of overflow error:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnOFE bit is set.

11.4.18.5 Overrun error

An overrun error can happen in direct access, transmit-only buffer, and FIFO modes. It cannot happen in dual buffer mode. The overrun error is not generated if data reception is disabled (CSIHnCTL0.CSIHnRXE = 0).

There are two conditions for overrun errors.

Condition for errors 1

- In FIFO mode, while the number of received data is 0 and CPU reads the CSIHnRX0W/H register

Condition for errors 2

- In slave mode, when CSIHnCTL1.CSIHnHSE = 0 (handshake function disabled):
  - In direct access mode or transmit-only buffer mode, when reception is completed while the previous received data is remained in the CSIHnRX0W/H register.
  - In FIFO mode, when FIFO buffer completes receiving data in the full state.

(1) Direct access/transmit-only buffer

In direct access and transmit-only buffer mode, this error occurs when newly received data cannot be transferred from the shift register to the reception register CSIHnRX0W/H. This happens when CSIHnRX0W/H was not read and therefore contains previous reception data.

The following figure illustrates the function.

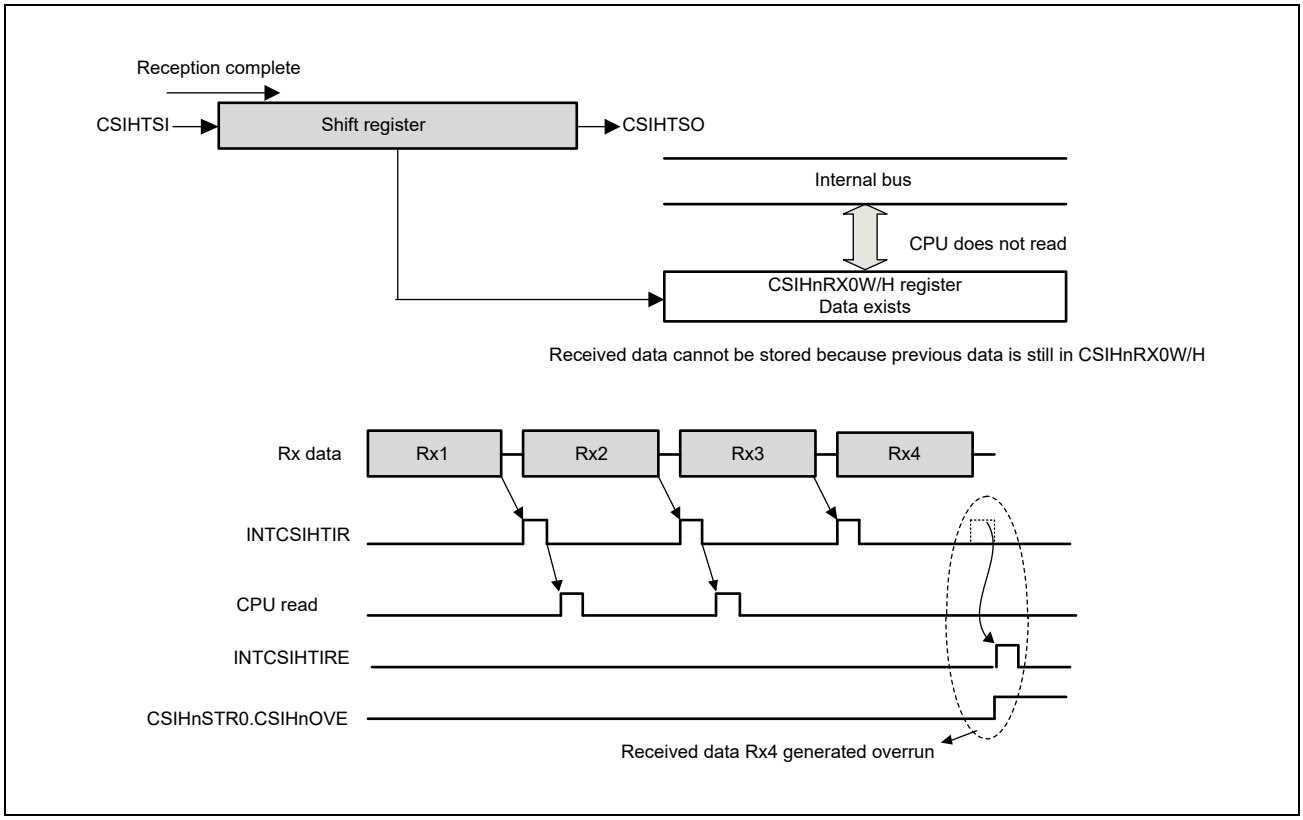


Figure 11.40 Overrun Error Detection in Direct Access and Transmit-only Buffer Mode

NOTE

An overrun error can be avoided in slave mode by using the handshake function.

When handshake is used in slave mode, the receiver (slave) signals to the transmitter (master) that it is busy. The transmitter then waits until the receiver has read its reception register and is ready again.

(2) FIFO mode

In FIFO mode, this error occurs if:

1. Newly received data cannot be transferred from the shift register to the FIFO because the FIFO is full.

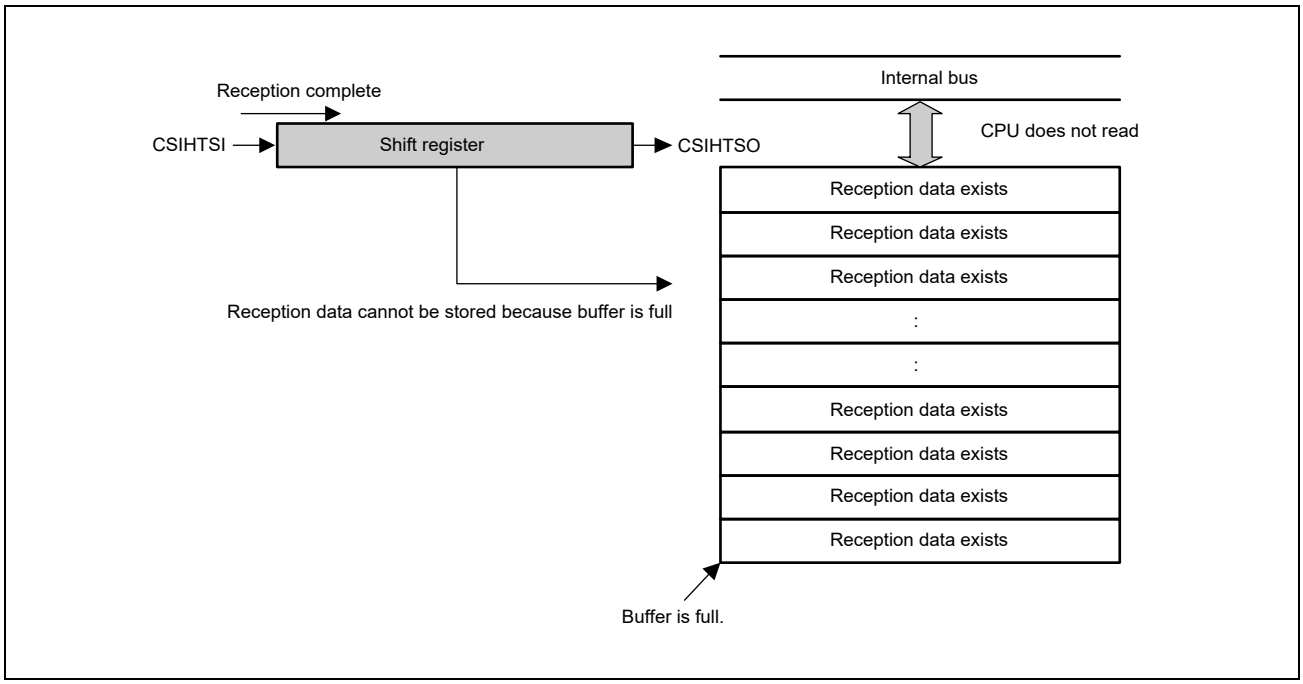


Figure 11.41 Overrun Error Detection in FIFO Mode (FIFO Full)

NOTE

An overrun error can be avoided in slave mode by using the handshake function.

When handshake is used in slave mode, the receiver (slave) signals to the transmitter (master) that it is busy. The transmitter then waits until the receiver has read its reception register and is ready again.

2. The CPU attempts to read non existing reception data.

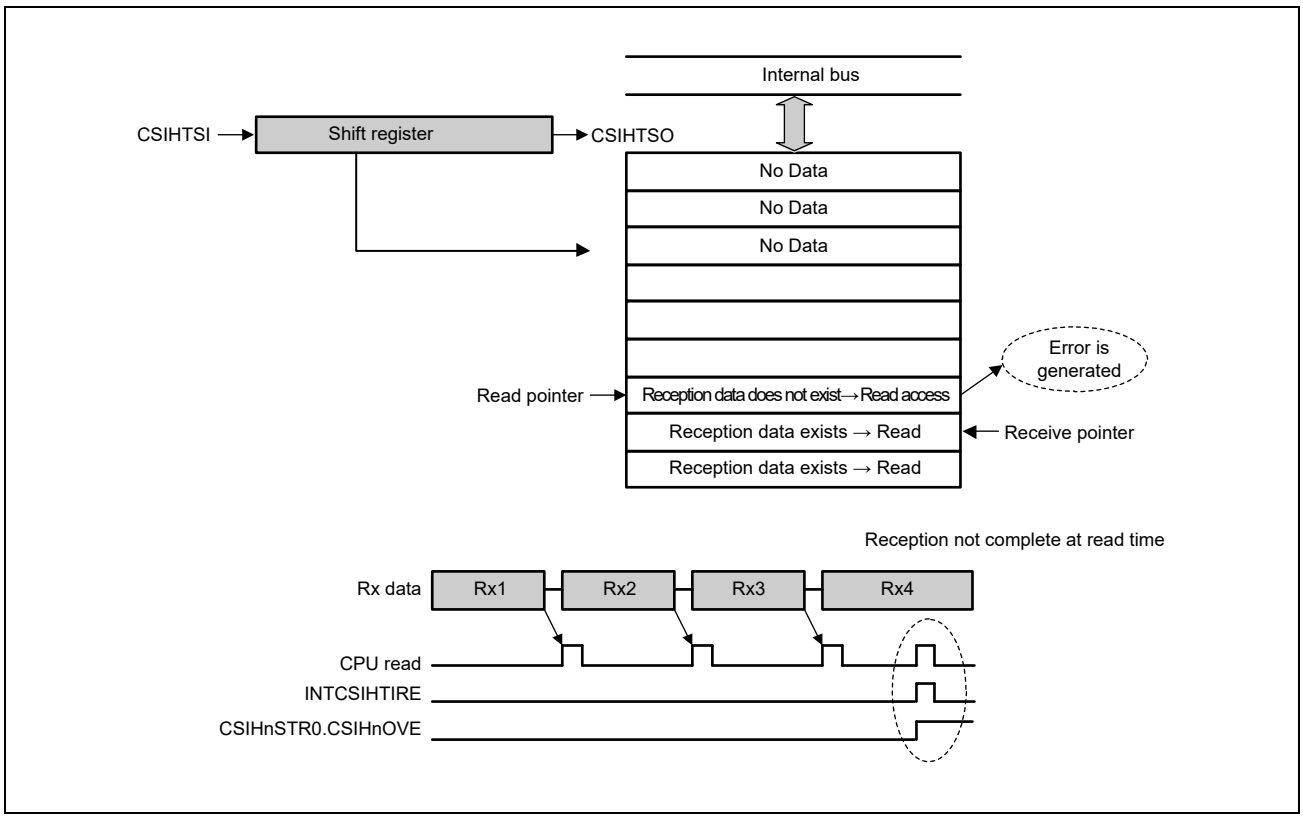


Figure 11.42 Overrun Error Detection in FIFO Mode (No Data)

In case of overrun error:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnOVE bit is set.
- Received data is overwritten and the communication continues.
  (When the CPU tries to read non-existent data, the CPU starts reading again after a wait until reception is completed.)

For details see **Section 11.4.17, Handshake Function**.

11.4.19 Loop-back Mode

Loop-back mode is a special mode for self-test. This feature is only available in master mode.

When this mode is active (CSIHnCTL1.CSIHnLBM = 1), CSIHTCSSx is fixed to the inactive level (the active level is defined by the CSIHnCTL1.CSIHnCSLx value). The transmit and receive signals are internally connected, as shown in the figures below. The signals CSIHTSCK, CSIHTSO, CSIHTSI, and CSIHTCSSx are disconnected from the ports. In addition, the CSIHTSO output level is fixed to low, and CSIHTSCK is set to reset level (High) regardless of the value of the CSIHnCFGx.CSIHnCKPx. The rest of CSIH works as in normal operation. Loop-back test has no effect on the device connected to.

In order to test CSIH, put it in loop-back mode and carry out normal transfer operations. Then check that the received data is the same as the transmitted data.

Table 11.42 Pin Output Level in Loop-back Mode

Pin Name	Output Level
CSIHTSCK (out)	High level
CSIHTCSS[x]	Inactive level
CSIHTSO	Low level (does not depend on the previous values)
Interrupt	Normal function
CSIHTRYO	Normal function (low level)

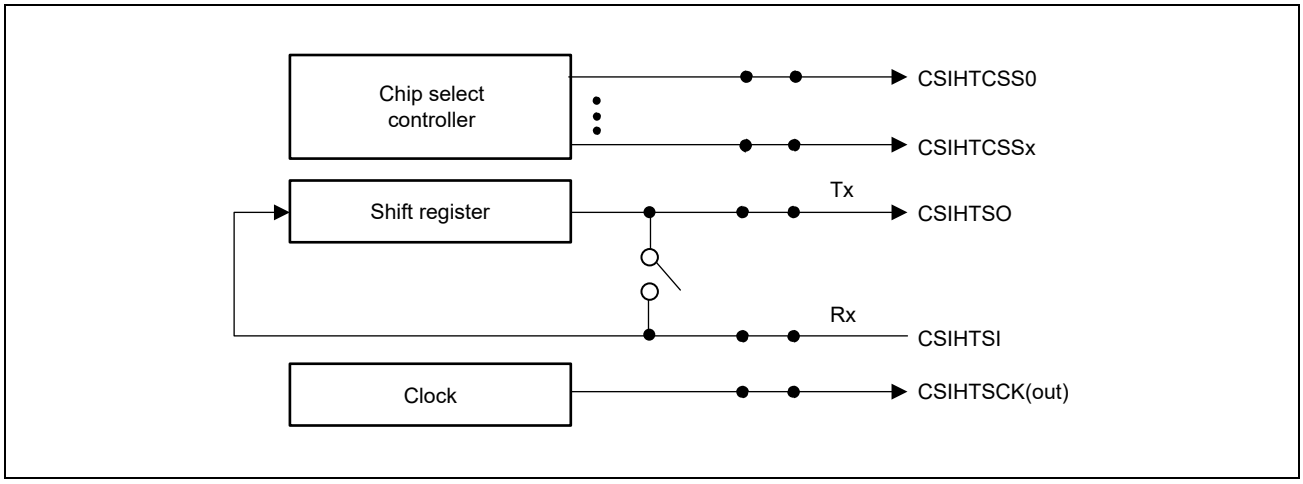


Figure 11.43 Normal Operation

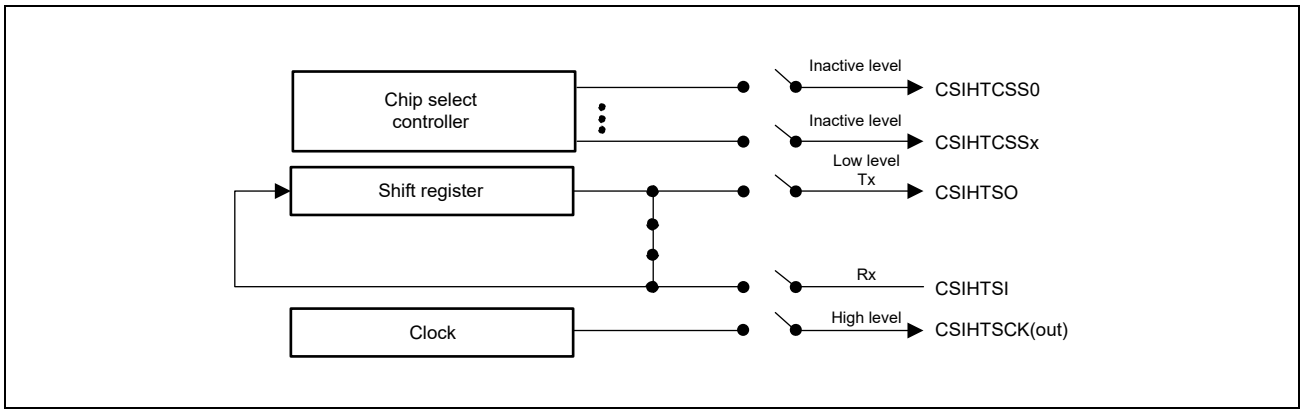


Figure 11.44 Loop-Back Operation

11.4.20 CPU-controlled High Priority Communication Function

CSIH has a function to abort low priority communication to perform high priority communication if it receives a high-priority communication request from the CPU while low-priority communication is being used. This function supports transmit-only buffer mode as low priority communication and direct access mode as high-priority communication only.

To enable this function, CSIHnCTL1.CSIHnPHE and CSIHnCTL1.CSIHnJE must be set to 1.

The following figure illustrates CPU-controlled high-priority communication.

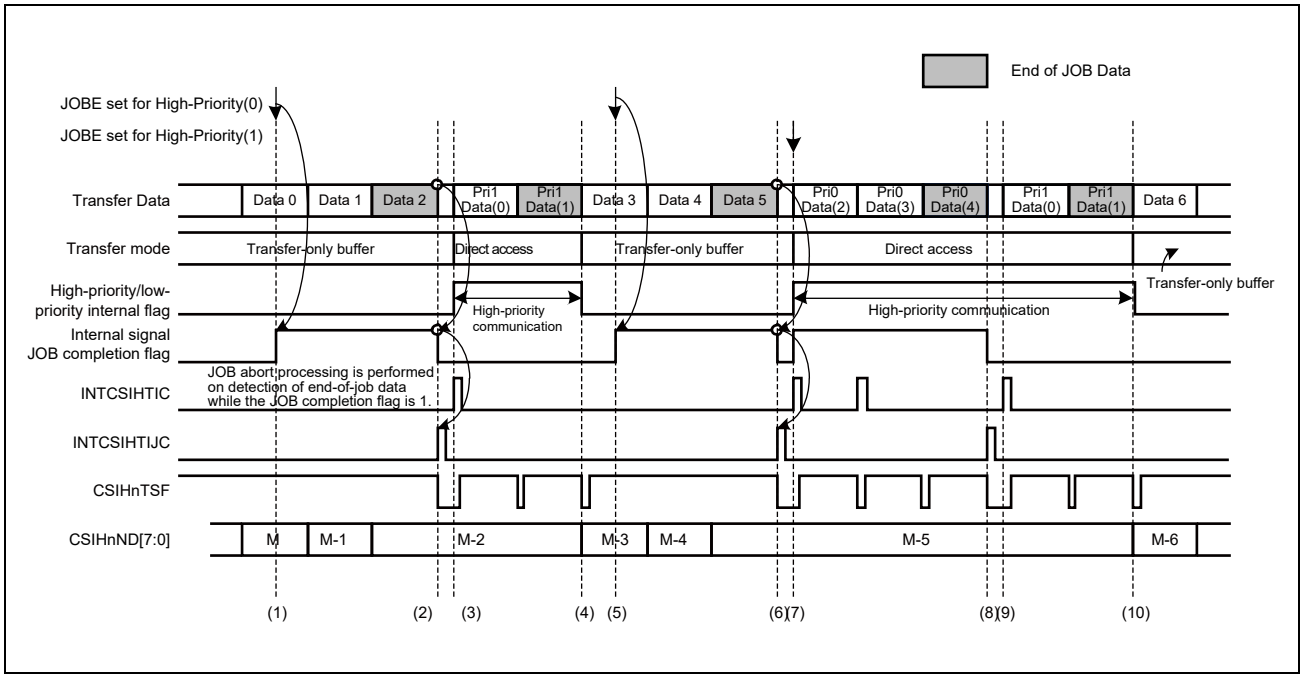


Figure 11.45 Example of CPU-Controlled High-Priority Communications, when CSIHnCTL1.CSIHnSLIT = 1

(1) By setting CSIHnCTL0.CSIHnJOBE = 1 during low-priority communication, start of high-priority communication following end-of-job data is notified, and the internal signal flag is set.

(2) When end-of-job data is detected, the current low-priority communication is aborted and the INTCSIHTIJC interrupt occurs. An internal signal, the JOB completion flag is cleared due to the abortion of communication, and memory mode is automatically switched to direct access mode for the subsequent high-priority communication.

(3) The CPU detects the interrupt and starts communication by writing the first transmission data of high-priority communication to CSIHnTX0W or CSIHnTX0H.

(4) When end-of-job data is detected, communication is aborted. At this time, because the internal signal, end-of-job flag is set to 0, the CSIH determines that the next communication is low priority and switches memory mode to transmit-only buffer mode automatically, and then resumes the aborted low-priority communication.

(5) Same as (1) above.

(6) Same as (2) above.

(7) The CPU detects an interrupt and starts communication by writing the first transmission data of high-priority communication to CSIHnTX0W or CSIHnTX0H. The CPU sets CSIHnCTL0.CSIHnJOBE = 1 again to notify that the next communication is high-priority.

(8) When end-of-job data is detected, communication is aborted and the INTCSIHTIJC interrupt is generated. At this time, the CPU determines that the subsequent communication is high-priority because the internal signal JOB completion flag is 1, and waits for communication to start.

(9) Same as (3) above.

(10) Same as (4) above.

CAUTION

Memory mode is switched automatically when communication is changed from low priority to high priority (switch from transmit-only buffer mode to direct access mode) and from high priority to low priority (switch from direct access mode to transmit-only buffer mode).

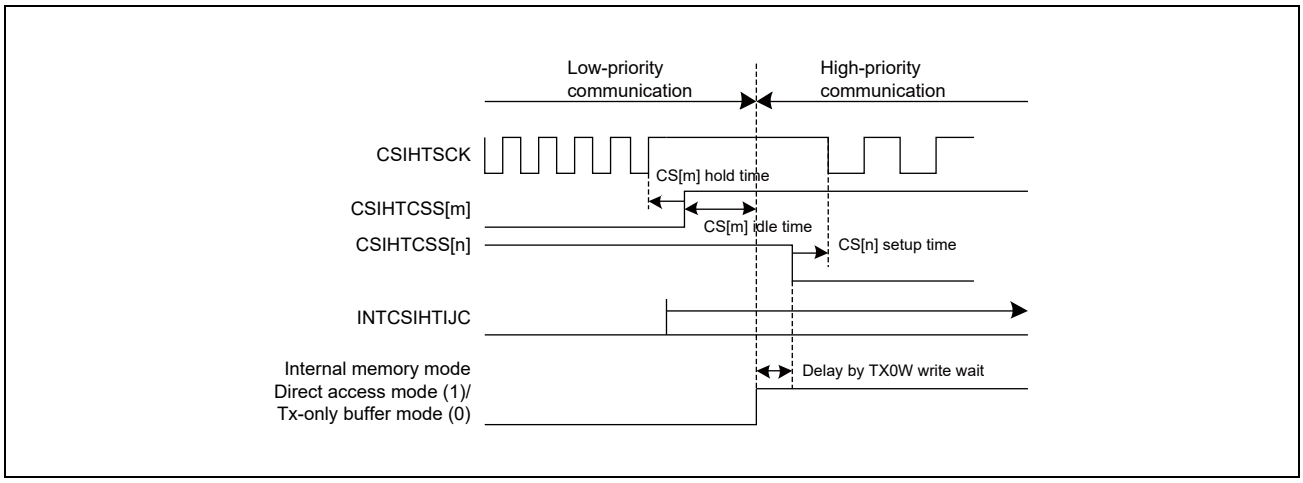


Figure 11.46 Transition from Low-Priority Mode to High-Priority Mode

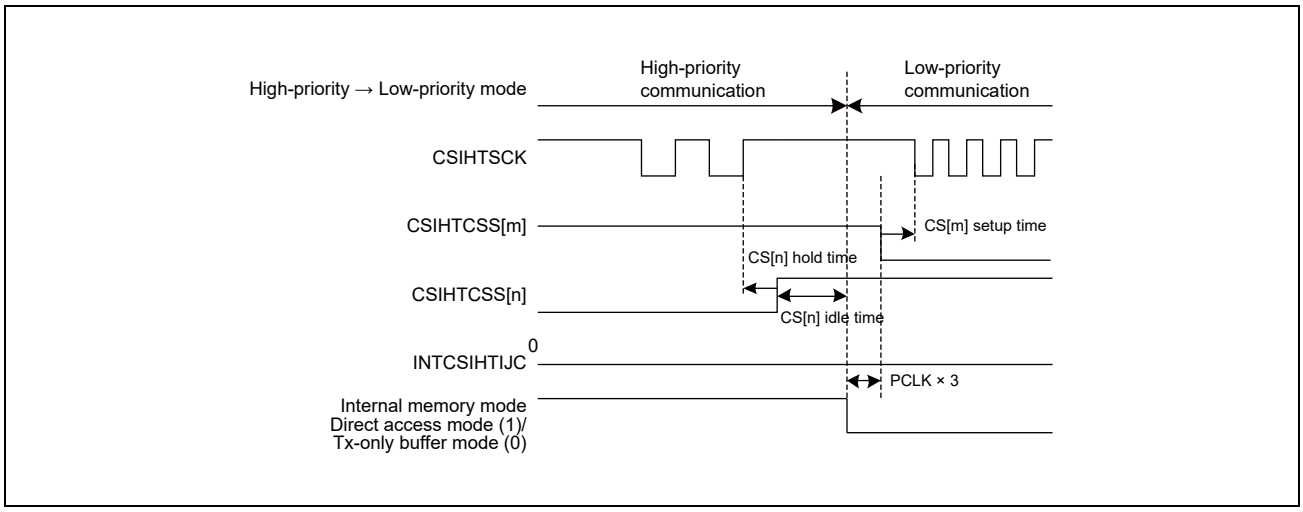


Figure 11.47 Transition from High-Priority Mode to Low-Priority Mode

Do not conduct write operation of communication data or CSIHnCTL0.CSIHnJOBE bit operation during setting prohibit period to switch low and high priority communication mode correctly.

CSIHnTX0W register write inhibited period:

- Period from when CSIHnJOBE bit is set for switching to high priority communication mode to when the INTCSIHTIJC interrupt is detected.
- Period from when the last data of high priority communication (End of JOB data) is written to when the CSIHnHPST state = 0 is detected.

CSIHnJOBE register write inhibited period:

- Period from when CSIHnJOBE bit is set for switching to high priority communication mode to when the INTCSIHTIJC interrupt is detected.

During high communication mode period, there is no setting prohibit period for CSIHnJOBE bit. It is possible to set CSIHnJOBE bit before writing communication data. For example, to communicate multiple JOB data in high priority mode, it is possible to set CSIHnJOBE bit before writing the first communication data.

CAUTION

When CSIHnJOBE bit is set right before the last communication of high priority communication ends, different operations are required depending on the internal detection timing of CSIHnJOBE bit setting. When CSIHnJOBE bit setting is detected before the last bit communication, high priority communication mode continues.

When setting of the CSIHnJOBE bit is detected after the transfer of the last bit is completed, the mode temporarily returns to low priority communications. After detection of End of JOB data in low priority communications, the mode changes back to high priority communications.

11.4.21 Enforced Chip Select Idle Setting

This macro is able to insert an idle state between the two consecutive transfer data by the setting of CSIHnCFGx.CSIHnIDLx. Detail is as follows.

1. When CSIHnCFGx.CSIHnIDLx = 0:
   If a next CSIHTCSSx is the same as the previous one, an idle state is not inserted and an inter-data time is inserted.
   An idle state is always inserted if a next CSIHTCSSx is different from the previous one.
2. When CSIHnCFGx.CSIHnIDLx = 1:
   An idle state is always inserted even if a next CSIHTCSSx is same with the previous one.

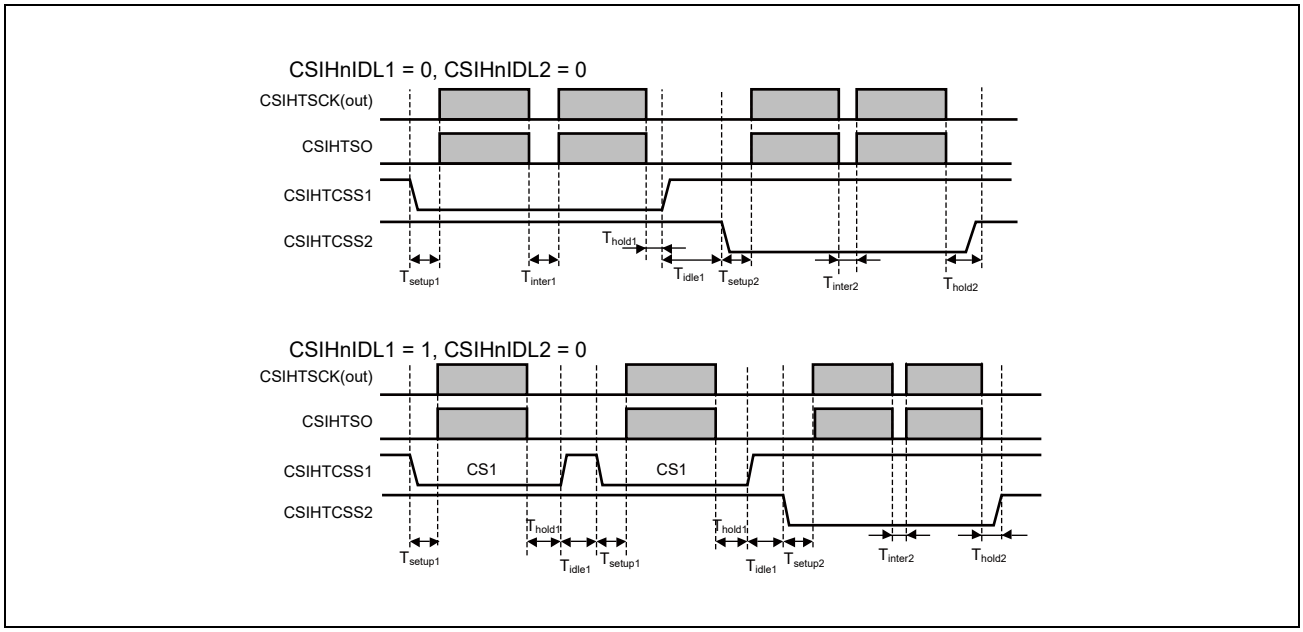


Figure 11.48 Enforced Chip Select Idle Example

CAUTION

If high priority communication function controlled by CPU is validated (CSIHnCTL1.CSIHnPHE = 1), when switch from low priority communication mode to high priority communication mode or switch from high priority communication mode to low priority communication mode, idle state is inserted regardless of the setting of CSIHnCFGx.CSIHnIDLx bit.

11.5 Procedures

The examples and procedures below are described according to the memory mode in the following order:

- Direct access mode
- Transmit-only buffer mode
- Dual buffer mode
- FIFO mode

11.5.1 Procedures in Direct Access Mode

Two examples for a master are provided, one with job mode being disabled, and the other with job mode being enabled.

11.5.1.1 Transmission/reception in master mode when job mode is disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- Transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Direct access mode (CSIHnCTL0.CSIHnMBS = 1)

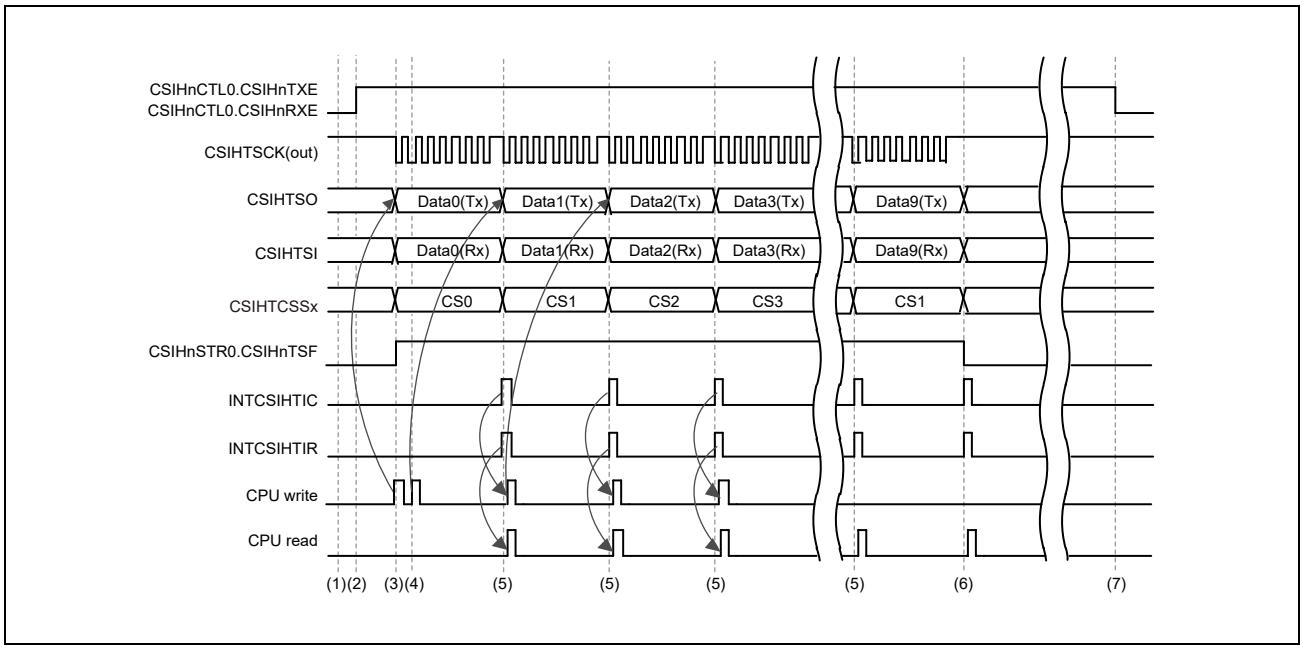


Figure 11.49 Master in Direct Access Mode, CSIHnCTL1.CSIHnJE = 0

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CSIHTCSS0 to CSIHTCSS3.
   Specify the transfer mode and job mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), CSIHnRXE = 1 (permits reception), and CSIHnMBS = 1 (selects direct access mode).
3. Write the first data to be sent to the transmission register, CSIHnTX0W. Within the same write operation, activate CS0. Transmission starts automatically when the first data becomes available.
4. Write the second data to CSIHnTX0W. If required, you can change the CS to address a different device. Writing the second data immediately after the first one avoids unnecessary delays between the data.
5. After every transmission/reception of a data the interrupts INTCSIHTIC and INTCSIHTIR are generated:
   - INTCSIHTIC indicates that the next data can be written to CSIHnTX0W.
   - INTCSIHTIR indicates that the reception register, CSIHnRX0W must be read.
6. No further actions of writing are required after the completion of transmission of data 8. Writing of data 9 (the last data) will have been completed at the time of the transmission completed interrupt for data 7. However, reception register CSIHnRX0W must be read after the completion of writing data 8 and data 9.
7. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.1.2 Transmission/reception in master mode when job mode is enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- Transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0).
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0).
- Direct access mode (CSIHnCTL0.CSIHnMBS = 1).
- Two jobs, each of them sends three data.

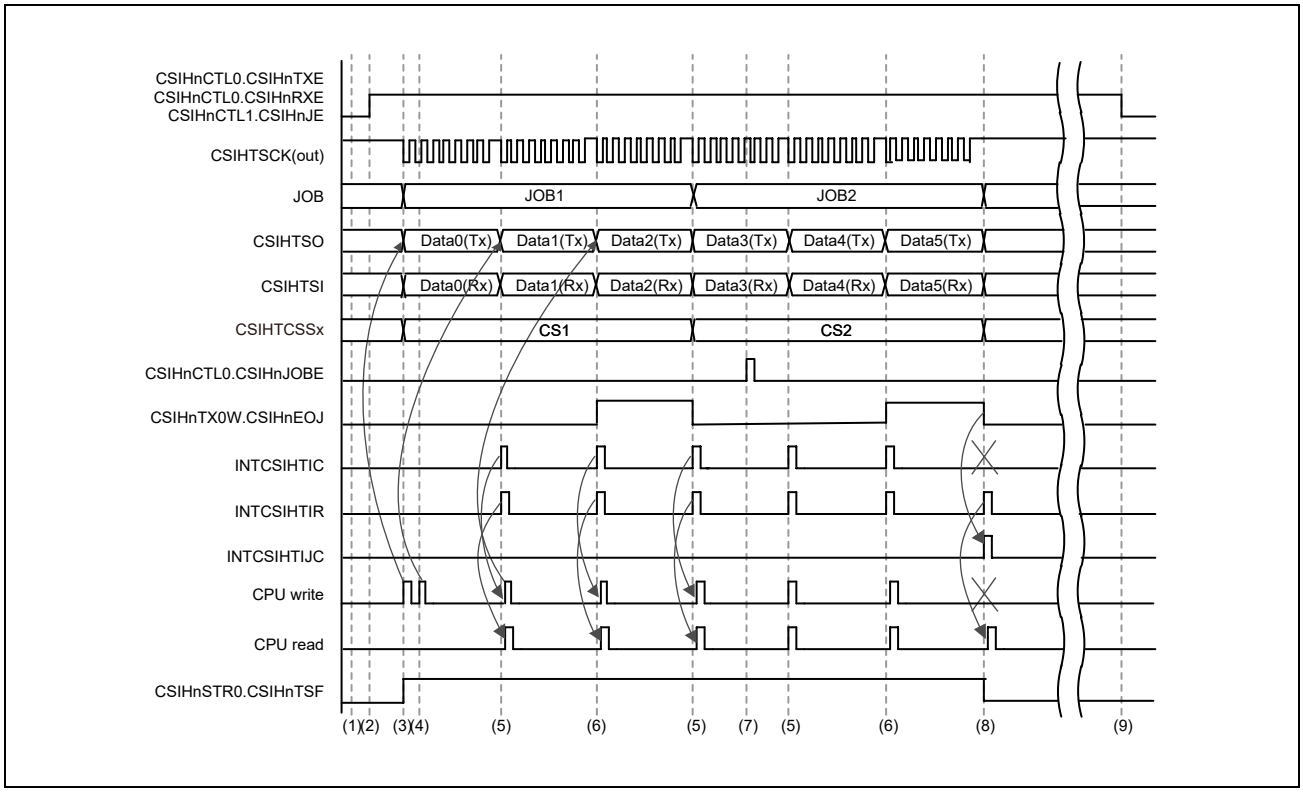


Figure 11.50 Master in Direct Access Mode, CSIHnCTL1.CSIHnJE = 1

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CSIHTCSS1 and CSIHTCSS2.
   Specify the transfer mode and job mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), CSIHnRXE = 1 (permits the reception), and CSIHnMBS = 1 (selects direct access mode).
3. Write the first data to be sent to the transmission register CSIHnTX0W. Transmission starts automatically when the first data becomes available.
   The CSIHnSTR0.CSIHnTSF flag indicates that communication is in progress.
4. Write the second data to CSIHnTX0W. Writing the second data immediately after the first one avoids unnecessary delays between the data.
5. After every data transmission/reception, the interrupt requests, INTCSIHTIC and INTCSIHTIR are generated.
   - INTCSIHTIC indicates that the next data can be written to CSIHnTX0W.
   - INTCSIHTIR indicates that the reception register, CSIHnRX0W must be read.
6. Setting CSIHnTX0W.CSIHnEOJ = 1 indicates that the last data of the current job is sent. After that, the next job may begin.
7. By setting CSIHnCTL0.CSIHnJOBE = 1, communication is forced to stop at the end of the current job (JOB2).
8. After the forced stop of communication, the interrupt request, INTCSIHTIC is replaced by INTCSIHTIJC. INTCSIHTIR is generated as usual.
   The interrupt request, INTCSIHTIJC indicates a forced stop of communication at the end of the current job.
   The interrupt request, INTCSIHTIC is not generated. Additionally, the transmission data available in the CSIHnTX0W register is not sent.
9. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.
   To start another transmission without stopping communication, perform steps 3 and later.

11.5.2 Procedures in Transmit-only Buffer Mode

Two examples for a master is provided, one with job mode being disabled, and the other with job mode being enabled.

11.5.2.1 Transmission/reception in master mode when job mode is disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- The number of data is 9 (CSIHnMCTL2.CSIHnND[7:0] = 09_H).
- The transfer start address is 10_H (CSIHnMCTL2.CSIHnSOP[6:0] = 10_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Transmit-only buffer mode (CSIHnCTL0.CSIHnMBS = 0,CSIHnMCTL0.CSIHnMMS[1:0] = 10_B)

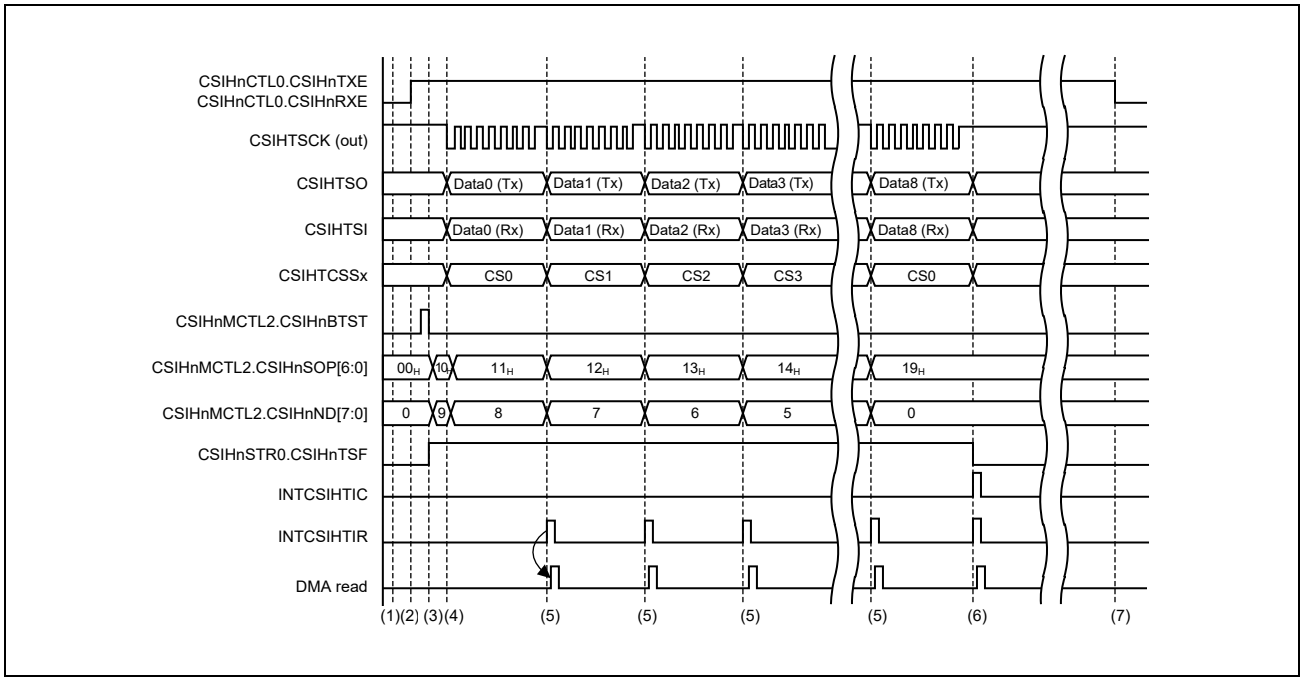


Figure 11.51 Master in Transmit-Only Buffer Mode, CSIHnCTL1.CSIHnJE = 0

NOTE

The procedure of writing the data into the buffer is not described here.

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CSIHTCSS0 to CSIHTCSS3.
   Specify the transfer and operating modes by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   CSIHnMCTL0.CSIHnMMS[1:0] is set to 10_B (transmit-only buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission) and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Configure the send pointer and the number of data by setting the CSIHnMCTL2.CSIHnSOP[6:0] and CSIHnMCTL2.CSIHnND[7:0] bits. Start the buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission/reception is started. The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data packet transmission.
5. After every data reception, the interrupt request, INTCSIHTIR is generated. INTCSIHTIR indicates necessity of reading reception register CSIHnRX0W.
6. When all transmissions are complete, the interrupt request, INTCSIHTIC is generated.
7. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.2.2 Transmission/reception in master mode when job mode is enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- The number of data is 8 (CSIHnMCTL2.CSIHnND[7:0] = 08_H).
- The transfer start address is 10_H (CSIHnMCTL2.CSIHnSOP[6:0] = 10_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Transmit-only buffer mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 10_B)

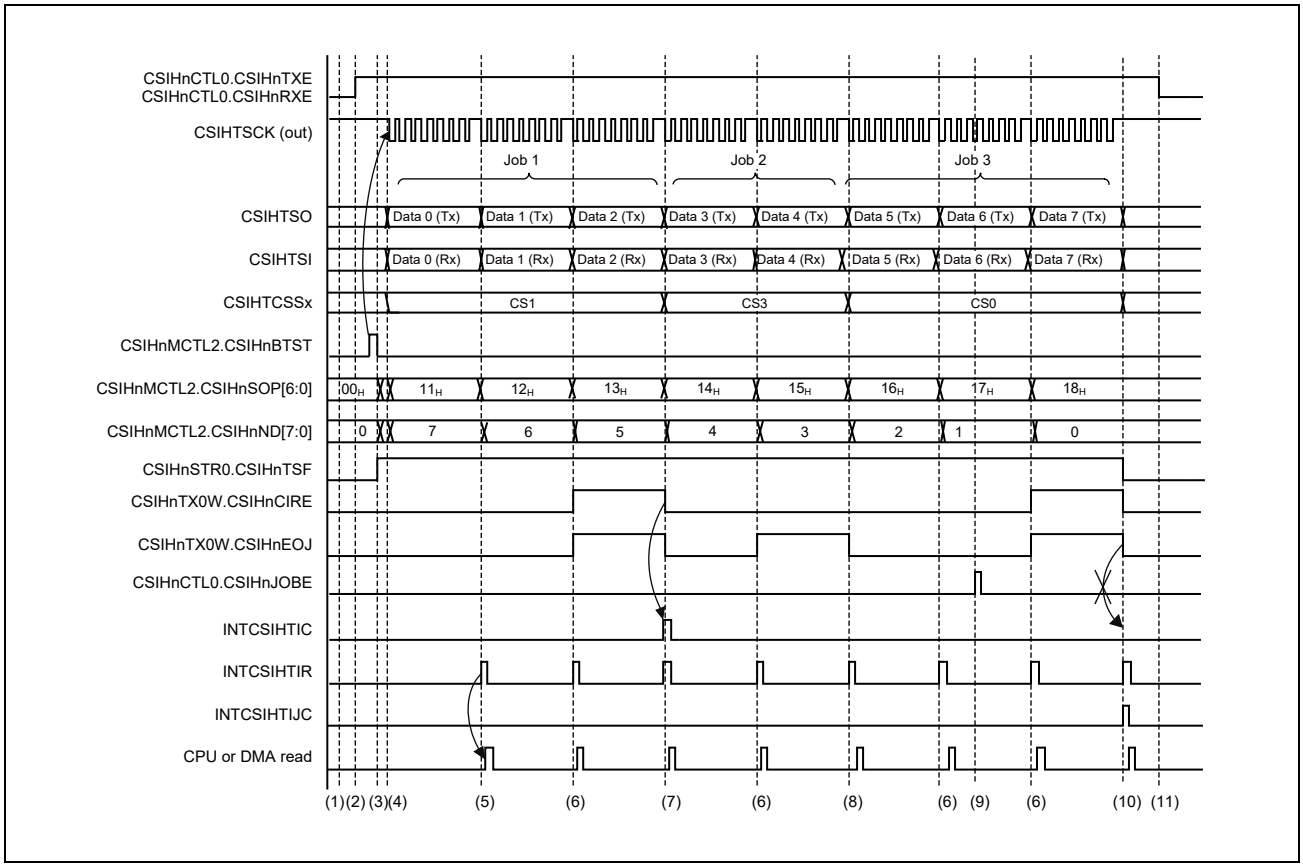


Figure 11.52 Master in Transmit-Only Buffer Mode, CSIHnCTL1.CSIHnJE = 1

NOTE

The process of writing the data into the buffer is not described here.

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register.
   Specify the transfer mode and job mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   Set CSIHnMCTL0.CSIHnMMS[1:0] = 10_B (transmit-only buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Configure the send pointer and the number of data by setting the CSIHnMCTL2.CSIHnSOP[6:0] and CSIHnMCTL2.CSIHnND[7:0] bits. Start the buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission/reception is started. The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data transmission.
5. After every data reception, the interrupt request, INTCSIHTIR is generated. INTCSIHTIR indicates necessity of reading reception register CSIHnRX0W.
6. The CSIHnTX0W.CSIHnEOJ = 1 setting indicates that the last data of the current job is sent.
7. The interrupt request INTCSIHTIC is generated. INTCSIHTIC indicates that the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 1.
8. The INTCSIHTIC interrupt request is not generated because the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 0.
9. By setting CSIHnCTL0.CSIHnJOBE = 1, communication is forced to stop at the end of JOB3.
10. After the forced stop of communication, interrupt requests INTCSIHTIJC and INTCSIHTIR are generated at the end of job3.
    The INTCSIHTIJC interrupt request indicates a forced stop of communication at the end of the current job.
    The INTCSIHTIC interrupt request is not generated because the INTCSIHTIJC interrupt request is generated instead of the INTCSIHTIC interrupt request. Additionally, the transmission data available in the CSIHnTX0W register is not sent.
11. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.3 Procedures in Dual Buffer Mode

Examples when job mode is enabled in master mode, disabled in master mode, and disabled in slave mode are provided below.

11.5.3.1 Transmission/reception in master mode when job mode is disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Default clock and data phase (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- The number of data is 9 (CSIHnMCTL2.CSIHnND[7:0] = 09_H).
- The transfer start address is 10_H (CSIHnMCTL2.CSIHnSOP[6:0] = 10_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Dual buffer mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 01_B)

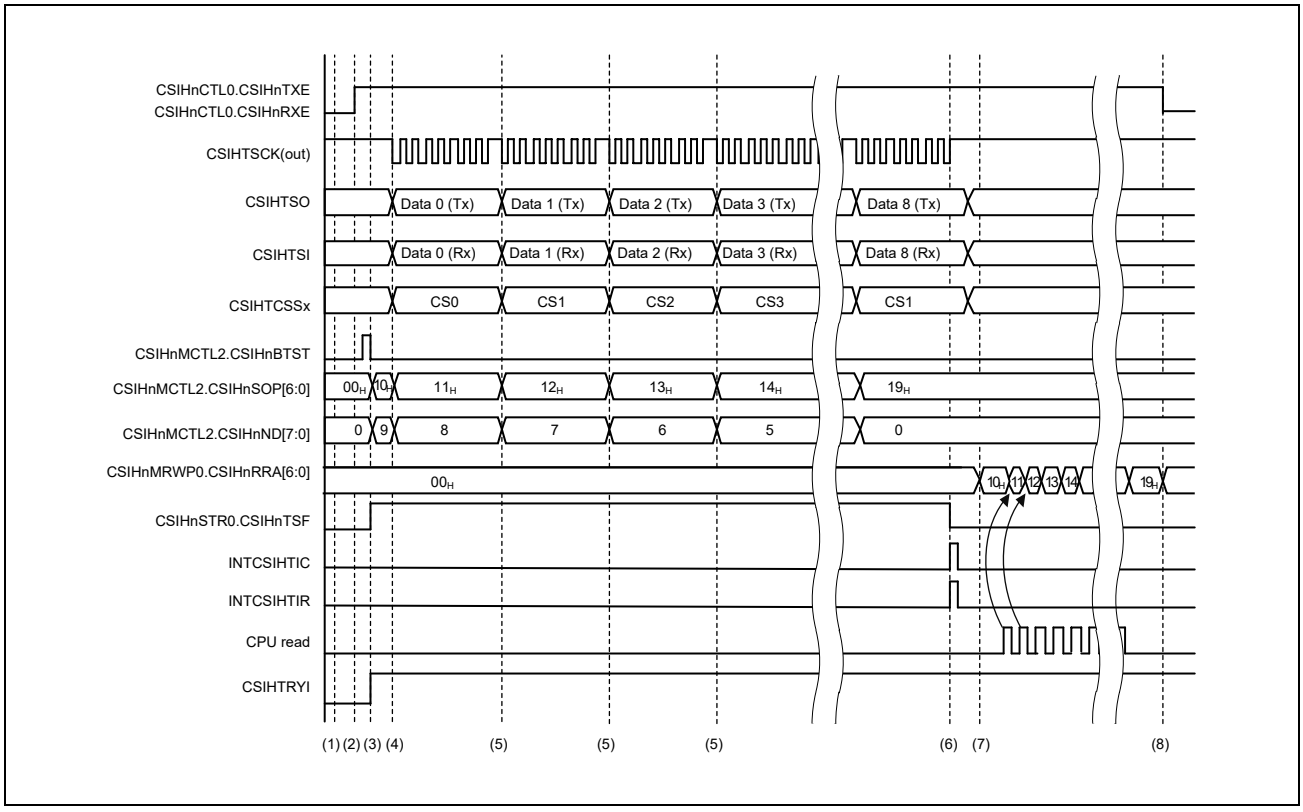


Figure 11.53 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0

NOTE

The process of writing the data into the buffer is not described here.

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CSIHTCSS0 to CSIHTCSS3.
   Specify the transfer and operating modes by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   Set CSIHnMCTL0.CSIHnMMS[1:0] = 01_B (dual buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits the reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Configure communication by setting CSIHnMCTL2.CSIHnSOP[6:0] and CSIHnMCTL2.CSIHnND[7:0]. Permit buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission is started. The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data transmission.
5. This is repeated until the last data is transmitted/received.
   The interrupt requests, INTCSIHTIC and INTCSIHTIR are not generated.
6. When the last data is transmitted/received, the interrupt requests, INTCSIHTIC and INTCSIHTIR are generated. The CPU starts to read the received data from the Rx buffer.
7. Set CSIHnMRWP0.CSIHnRRA[6:0] to specify the address where reading is to start (use software to set CSIHnRRA[6:0] to 10_H).
   The value of CSIHnRRA[6:0] is incremented every time a byte is read.
8. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.3.2 Transmission/reception in master mode when job mode is enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- The number of data is 8 (CSIHnMCTL2.CSIHnND[7:0] = 08_H).
- The transfer start address is 00_H (CSIHnMCTL2.CSIHnSOP[6:0] = 00_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Dual buffer mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 01_B)

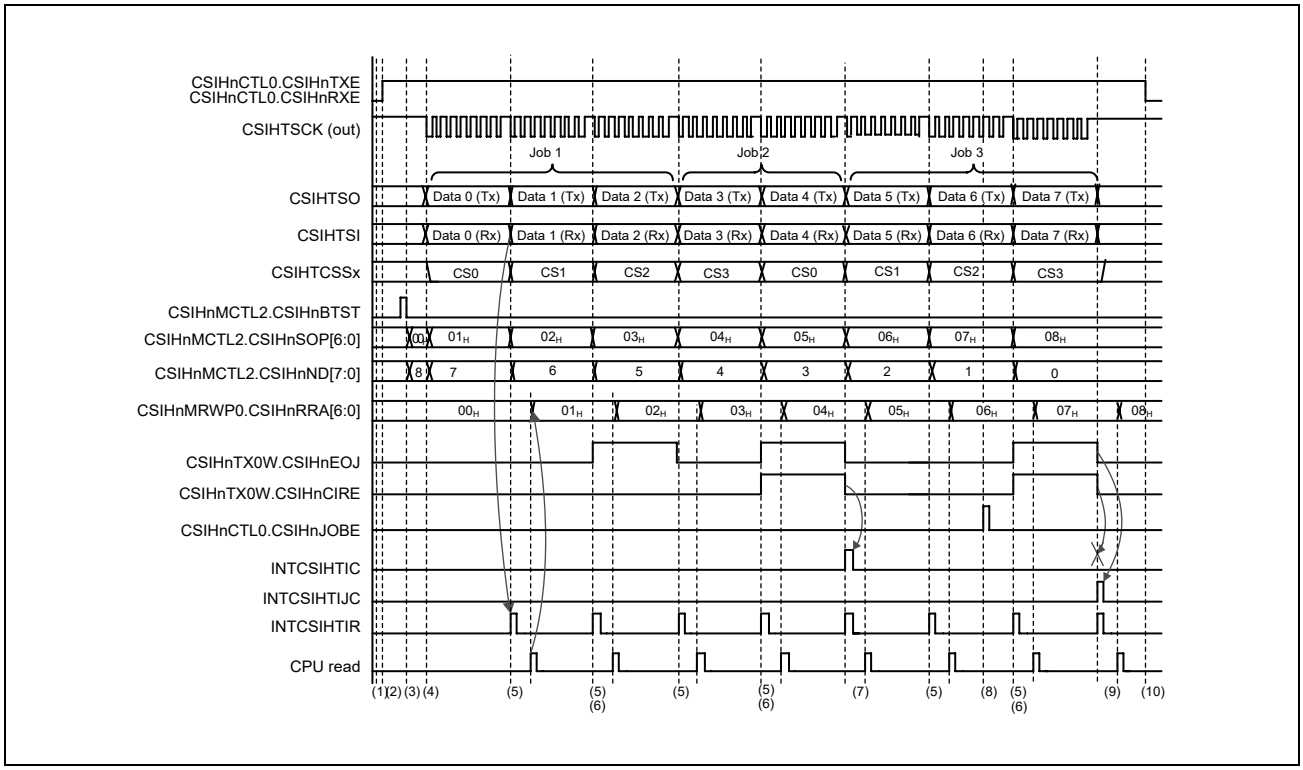


Figure 11.54 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 1

NOTE

The process of writing the data into the buffer is not described here.

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register.
   Specify the transfer and operating modes by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   Set CSIHnMCTL0.CSIHnMMS[1:0] = 01_B (dual buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Configure communication by setting the CSIHnMCTL2.CSIHnSOP[6:0] and CSIHnMCTL2.CSIHnND[7:0] bits. Start the buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission is started. The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented, and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data transmission. This is repeated until the last data is transmitted/received.
5. The INTCSIHTIR interrupt request is generated every time a data is received.
   The INTCSIHTIC interrupt request is not generated because the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 0.
6. CSIHnTX0W.CSIHnEOJ = 1 indicates that the last data of the current job is sent.
7. The INTCSIHTIC interrupt request is generated. INTCSIHTIC indicates that the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) is sent with CSIHnTX0W.CSIHnCIRE = 1.
8. CSIHnCTL0.CSIHnJOBE = 1 indicates that communication is forced to stop at the end of JOB3.
9. After the forced stop of communication, interrupt requests, INTCSIHTIJC and INTCSIHTIR are generated at the end of JOB3.
   The INTCSIHTIJC interrupt request indicates a forced stop of communication at the end of the current job.
   The INTCSIHTIC interrupt request is not generated because the INTCSIHTIJC interrupt request is generated instead of the INTCSIHTIC interrupt request. The transmit data available in CSIHnTX0W register is not transmitted this time.
10. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.3.3 Transmission/reception in slave mode when job mode is disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCTL1.CSIHnCKR = 0, CSIHnCFG0.CSIHnDAP0 = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- The number of data is 9 (CSIHnMCTL2.CSIHnND[7:0] = 09_H).
- The transfer start address is 10_H (CSIHnMCTL2.CSIHnSOP[6:0] = 10_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Dual buffer mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 01_B)
- Handshake function is enabled (CSIHnCTL1.CSIHnHSE = 1).

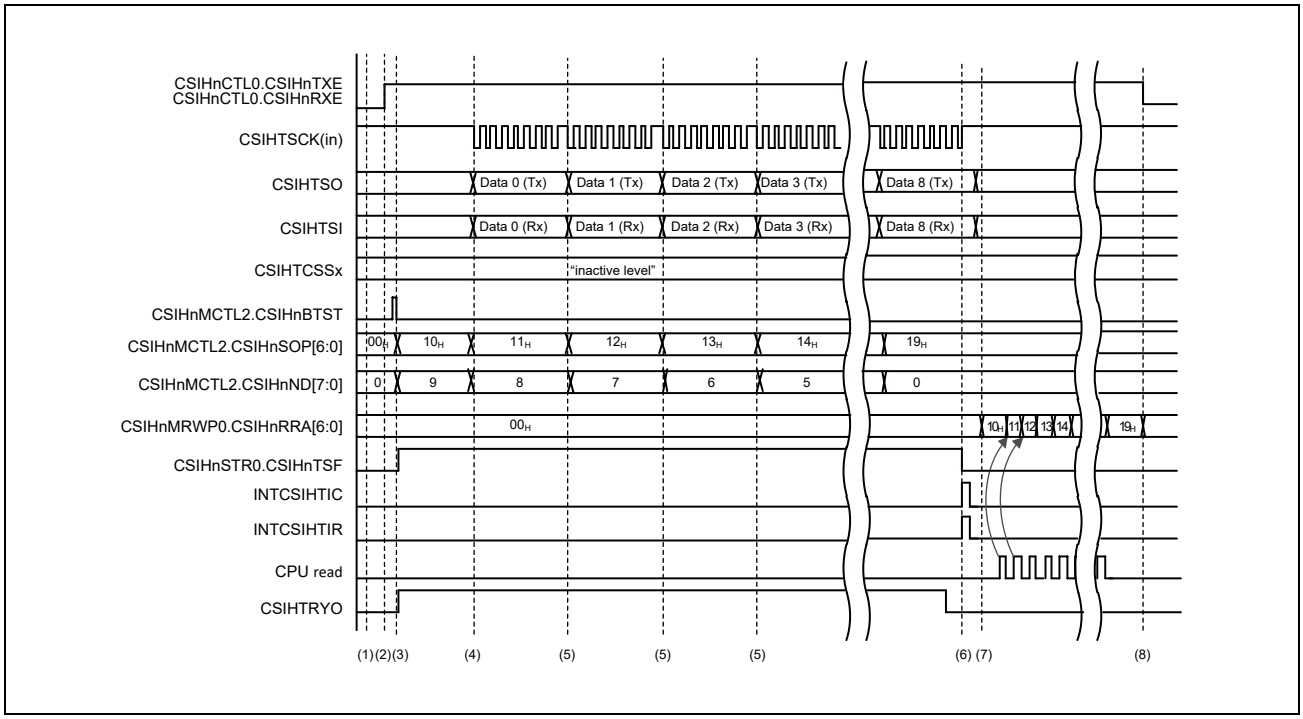


Figure 11.55 Slave in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0

NOTE

The process of writing the data into the buffer is not described here.

Procedure:

1. Configure the communication protocol in the CSIHnCFG0 register.
   Specify the transfer and operating modes by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   Set CSIHnMCTL0.CSIHnMMS[1:0] = 01_B (dual buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Specify the transfer start address by setting CSIHnMCTL2.CSIHnSOP[6:0] and the number of data by setting CSIHnMCTL2.CSIHnND[7:0]. Permit the buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission is started when the input clock from the master is received.
   The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data transmission.
5. This is repeated until the last data is transmitted/received.
   The interrupt requests, INTCSIHTIC and INTCSIHTIR are not generated because transmission data is sent from the buffer, and received data is stored in the buffer.
6. When the last data is transmitted/received, the interrupt requests, INTCSIHTIC and INTCSIHTIR are generated.
   The CPU starts to read the received data that is stored in the receive buffer.
7. Set CSIHnMRWP0.CSIHnRRA[6:0] to specify the address where reading is to start (use software to set CSIHnRRA[6:0] to 10_H).
   The value of CSIHnRRA[6:0] is incremented every time a byte is read.
8. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation.
   When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.4 Procedures in FIFO Mode

Two examples for a master is provided, one with job mode disabled, the other one with job mode enabled.

11.5.4.1 Transmission/reception in master mode when job mode is disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- FIFO mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 00_B)

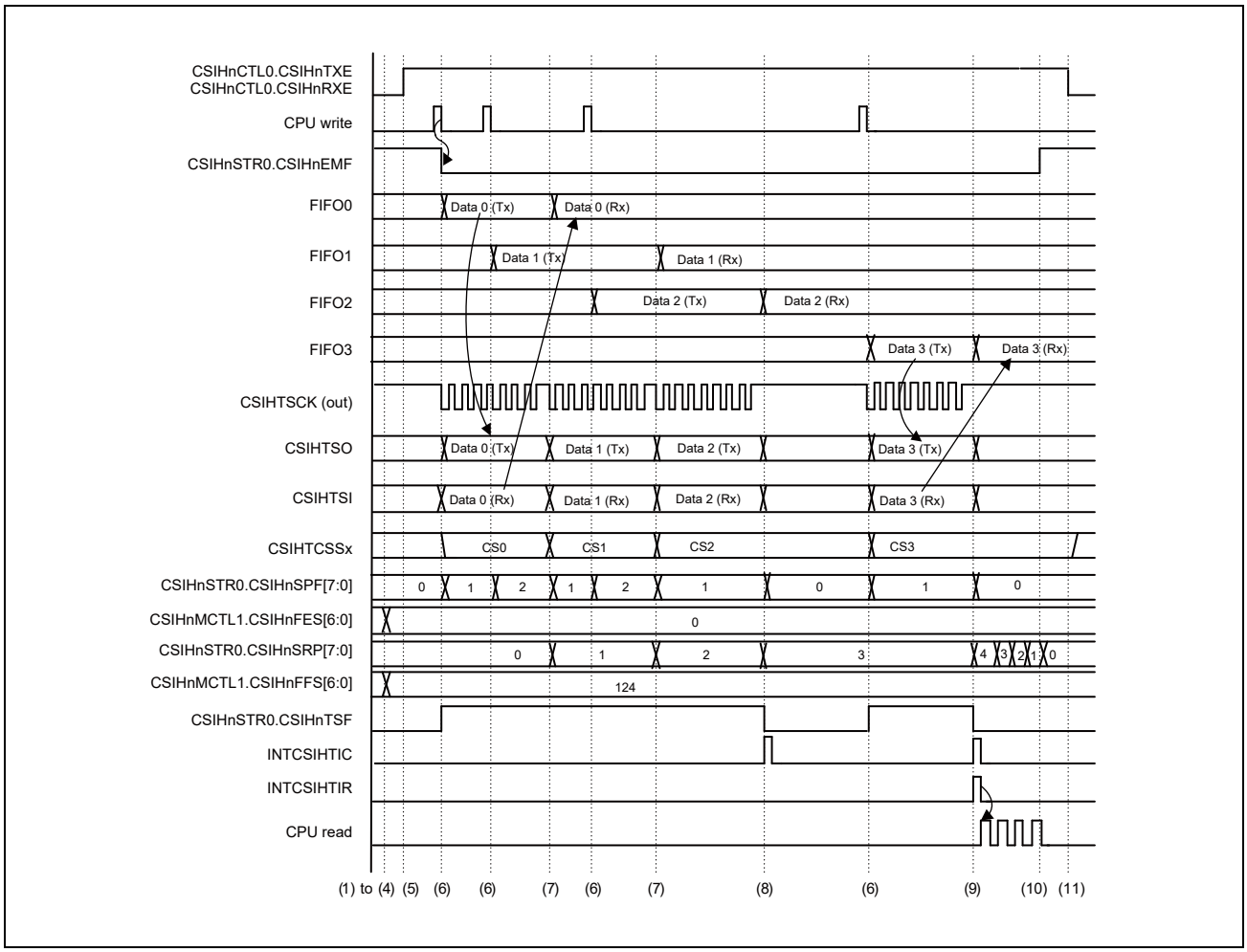


Figure 11.56 Master in FIFO Mode, CSIHnCTL1.CSIHnJE = 0

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. Specify the job mode disable and master mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Specify the FIFO mode by CSIHnMCTL0.CSIHnMMS[1:0] = 00_B. This example uses chip select signals CSIHTCSS0 to CSIHTCSS3.
2. Set CSIHnSTCR0.CSIHnPCT = 1 to clear all buffer pointers.
3. Check that CSIHnSTR0.CSIHnFLF = 0, CSIHnSTR0.CSIHnEMF = 1, and CSIHnSTR0.CSIHnSPF[7:0] = 00_H.
4. The CSIHnMCTL1.CSIHnFES[6:0] bits specify the condition for the INTCSIHTIC interrupt output.
   The CSIHnFFS[6:0] bits in the same register specify the condition for the INTCSIHTIR interrupt.
5. Set CSIHnCTL0.CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
6. Write the first transmit data packet to the transmit register CSIHnTX0W. Transmission starts automatically when the first data becomes available.
7. The current transmission is completed.
   As the CSIHnFES[6:0] bits are not the same as the CSIHnSPF[7:0] bits, the interrupt request INTCSIHTIC is not generated.
8. As the CSIHnFES[6:0] bits = CSIHnSPF[7:0] bits, the interrupt request INTCSIHTIC is generated.
9. When CSIHnFFS[6:0] = 128-CSIHnSRP[7:0], the interrupt request INTCSIHTIR is generated. As CSIHnFES[6:0] = CSIHnSRF[7:0], the interrupt request INTCSIHTIC is generated. After the generation of the interrupt, the CPU starts reading received data that is stored in the receive buffer.
10. When the CPU completes reading the received data that is stored in the receive buffer, CSIHnSTR0.CSIHnEMF is set to 1 and the FIFO buffer will be empty.
11. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.4.2 Transmission/reception in master mode when job mode is enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0)
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- JOB1 consists of four data, JOB2 consists of three data, and JOB3 consists of five data packets.
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- FIFO mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 00_B)

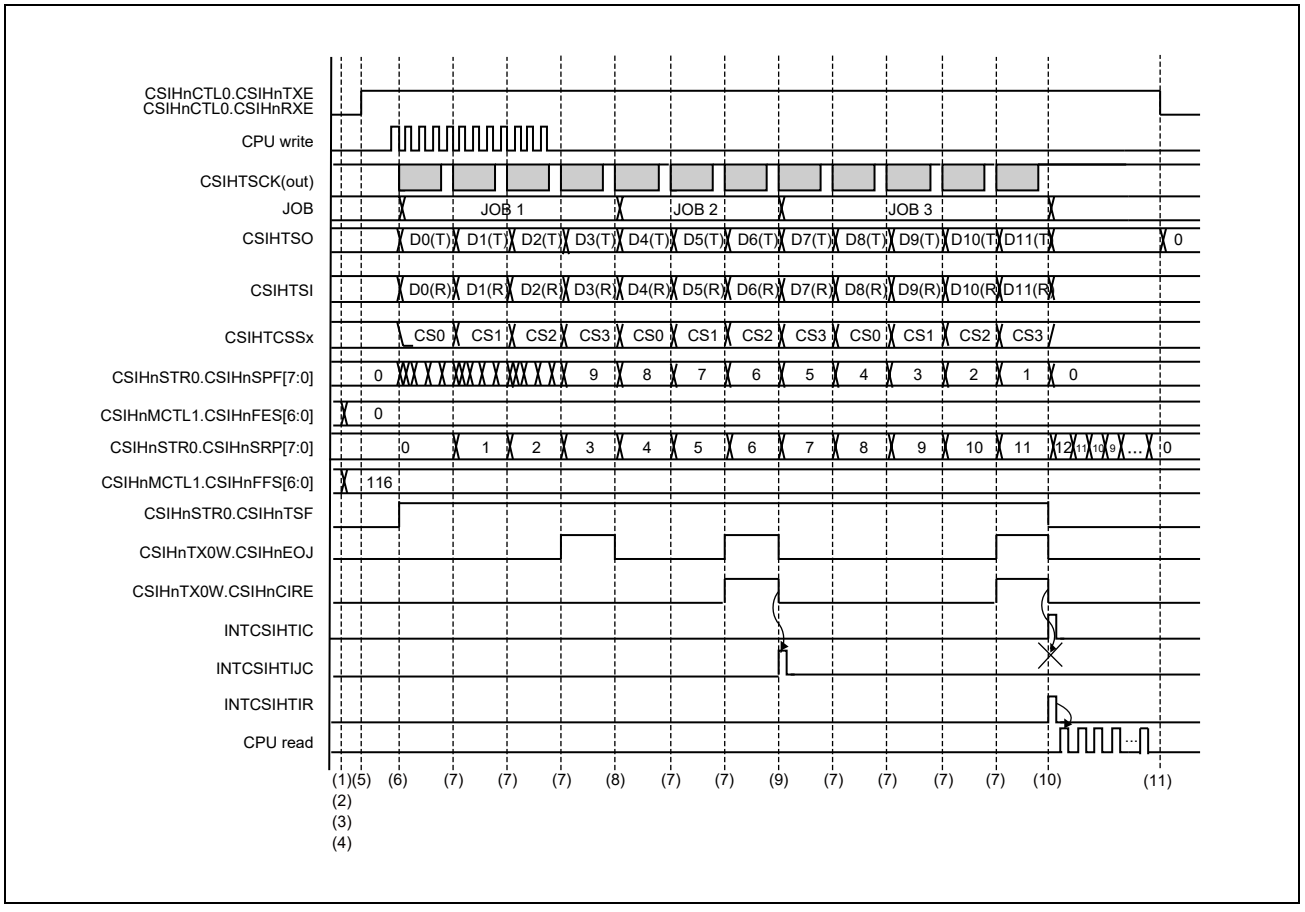


Figure 11.57 Master in FIFO Mode, CSIHnCTL1.CSIHnJE = 1

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register.
   Disable the job mode and select the memory mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Select the FIFO mode by setting CSIHnMCTL0.CSIHnMMS[1:0] = 00_B.
2. Set CSIHnSTCR0.CSIHnPCT to 1 to clear all buffer pointers.
3. Make sure CSIHnSTR0.CSIHnFLF is set to 0, CSIHnSTR0.CSIHnEMF is set to 1, and CSIHnSTR0.CSIHnSPF[7:0] is set to 00_H.
4. With CSIHnMCTL1.CSIHnFES[6:0], specify the conditions for generating the INTCSIHTIC interrupt request.
   With CSIHnMCTL1.CSIHnFFS[6:0], specify the conditions for generating the INTCSIHTIR interrupt request.
5. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
6. Write the first data to be sent to the CSIHnTX0W transmission register. Transmission starts automatically when the first data becomes available.
7. The current transmission is completed.
   As CSIHnFES[6:0] are not the same as CSIHnSPF[7:0], the interrupt request INTCSIHTIC is not generated.
8. The INTCSIHTIJC interrupt request is not generated because the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 0.
9. The INTCSIHTIJC interrupt request is generated because the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 1.
10. As the CSIHnFES[6:0] bits = the CSIHnSPF[7:0] bits, the interrupt request INTCSIHTIC is generated. As INTCSIHTIC is generated, INTCSIHTIJC is not generated.
    When CSIHnFFS[6:0] = 128-CSIHnSRP[7:0], the interrupt request INTCSIHTIR is generated. After the generation of the interrupt, the CPU starts reading received data that is stored in the receive buffer.
11. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation.
    When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.6 Notes

Table 11.43 Notes on Setting Registers (1/3)

Register	Bit	Content
CSIHnCTL0	CSIHnPWR	If this bit is cleared during communication, ongoing communication is suspended. After that, the communication must be restarted.
CSIHnCTL0	CSIHnTXE CSIHnRXE	Do not modify any of these bits while CSIHnCTL0.CSIHnPWR=0. (These bits can be modified at the same time as the CSIHnCTL0.CSIHnPWR bit.) Do not modify these bits while CSIHnSTR0.CSIHnTSF = 1, because the specified operation is not guaranteed if ongoing communication is suspended.
CSIHnCTL0	CSIHnJOBE	Do not modify this bit while CSIHnCTL0.CSIHnPWR = 0. This bit is only valid when CSIHnCTL1.CSIHnJE = 1. Setting this bit is prohibited in slave mode.
CSIHnCTL0	CSIHnMBS	Do not modify this bit while CSIHnCTL0.CSIHnPWR = 0. (This bit can be modified at the same time as the CSIHnCTL0.CSIHnPWR bit.) Modification of this bit is only allowed while CSIHnSTR0.CSIHnTSF = 0. Do not change the mode between FIFO mode and direct access mode while CSIHnCTL0.CSIHnPWR = 1. When the CPU-controlled high-priority communication function is enabled, the operation is the same as that in direct access mode regardless of the CSIHnMBS bit setting.
CSIHnCTL1	CSIHnCKR	Modification of this bit is only allowed while CSIHnCTL0.CSIHnPWR = 0. If CS is not used, use this bit instead of setting the CSIHnCFGx.CSIHnCKPx bit, and set the CSIHnCFGx.CSIHnCKPx bit to 0. This bit must be used in slave mode.
CSIHnCTL1	CSIHnSLIT CSIHnCSL[3:0] CSIHnEDLE CSIHnDCS CSIHnCSRI CSIHnHSE	Modification of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0.
CSIHnCTL1	CSIHnPHE CSIHnJE CSIHnLBM	Modification of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0. Setting of this bit is prohibited in slave mode.
CSIHnCTL1	CSIHnSSE	Modification of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. Setting of this bit is prohibited in master mode.
CSIHnCTL1	CSIHnSIT	Modification of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. This bit is only valid in master mode. In slave mode, no delay is generated.
CSIHnCTL2	CSIHnPRS[2:0]	Modification of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. Setting of the max baud rate is as follows. Master mode: PCLK/8 Slave mode: PCLK/16
CSIHnSTR0	CSIHnSRP[7:0] CSIHnSPF[7:0] CSIHnFLF CSIHnEMF CSIHnTSF	Writing these bits is prohibited, and only reading is permitted.
CSIHnSTR0	CSIHnTMOE CSIHnOFE CSIHnDCE CSIHnPE CSIHnOVE	Writing these bits is prohibited, and only reading is permitted. These bits are initialized when CSIHnCTL0.CSIHnPWR is changed from 0 to 1 or 1 to 0.
CSIHnSTCR0	CSIHnPCT	If this bit is set to1 during communication, ongoing communication is suspended.
CSIHnMCTL0	CSIHnMMS[1:0]	Modification of these bits is only allowed while CSIHnCTL0.CSIHnPWR = 0 and CSIHnCTL0.CSIHnMBS = 0.

Table 11.43 Notes on Setting Registers (2/3)

Register	Bit	Content
CSIHnMCTL0	CSIHnTO[4:0]	Modification of these bits is only allowed while CSIHnCTL0.CSIHnPWR = 0. Set these bits to 0 in master mode. Set these bits to 0 in direct access, dual buffer, and transmit-only buffer mode.
CSIHnMCTL1	CSIHnFES[6:0] CSIHnFFS[6:0]	Write to these bits during communication is permitted.
CSIHnMCTL2	CSIHnBTST CSIHnND[7:0] CSIHnSOP[6:0]	Writing these bits is prohibited when CSIHnCTL0.CSIHnPWR = 0. Writing these bits is prohibited when CSIHnCTL0.CSIHnTXE = CSIHnCTL0.CSIHnRXE = 0. Writing these bits is prohibited when CSIHnSTR0.CSIHnTSF = 1. Writing these bits is prohibited in direct access or FIFO mode.
CSIHnMRWP0	CSIHnRRA[6:0]	Write to these bits during communication is permitted. Writing these bits is prohibited in direct access or FIFO mode. When writing is required, set "0000_H" to these bits in transmit-only buffer mode.
CSIHnMRWP0	CSIHnTRWA[6:0]	Write to these bits during communication is permitted. Writing these bits is prohibited in direct access or FIFO mode.
CSIHnCFGx	CSIHnBRSSx[1:0] CSIHnRCBx CSIHnIDLx CSIHnIDx[2:0] CSIHnHDx[3:0] CSIHnINx[3:0] CSIHnSPx[3:0]	Modification of these bits is only allowed while CSIHnCTL0.CSIHnPWR = 0. These bits must be set to 0 in slave mode.
CSIHnCFGx	CSIHnPSx[1:0] CSIHnDLSx[3:0] CSIHnDIRx CSIHnDAPx	Modification of these bits is only allowed while CSIHnCTL0.CSIHnPWR = 0. In slave mode, the CSIHnCFG0 setting is used. Therefore, all the bits in CSIHnCFG1 to CSIHnCFG3 must be set to 0.
CSIHnCFGx	CSIHnCKPx	Modification of this bit is only allowed while CSIHnCTL0.CSIHnPWR = 0. As CSIHnCTL1.CSIHnCKR must be used in slave mode, set these bits to 0. If CS is not used, use the CSIHnCTL1.CSIHnCKR bit instead of these bits, and clear the bits to 0.
CSIHnTX0W	CSIHnEOJ CSIHnCIRE	This bit is only valid when CSIHnCTL1.CSIHnJE = 1. While CSIHnCTL1.CSIHnJE = 0, the value of this bit is ignored even if 1 is read. Set these bits to 0 in slave mode.
CSIHnTX0W	CSIHnEDL	This bit is only valid when CSIHnCTL1.CSIHnEDLE = 1. While CSIHnCTL1.CSIHnEDLE = 0, the value of this bit is ignored even if 1 is read.
CSIHnTX0W	CSIHnCS[3:0]	In master mode, setting these bits to F_H is prohibited. In slave mode, set these bits to E_H.
CSIHnTX0W CSIHnTX0H		Reading these bits during communication is prohibited in FIFO mode. While CSIHnCTL0.CSIHnPWR = 0 and in FIFO mode, read/write access to these bits is prohibited. While CSIHnCTL0.CSIHnTXE = CSIHnCTL0.CSIHnRXE = 0, write access to these bits are prohibited in direct access mode.
CSIHnRX0W		These bits are initialized when CSIHnCTL0.CSIHnPWR is changed from 0 to 1 or from 1 to 0. When CSIHnCTL0.CSIHnPWR = 0, reading these bits is prohibited in FIFO mode. When CSIHnCTL0.CSIHnPWR = 1, writing to these bits has no effect in FIFO mode. Reading is only effective. When CSIHnCTL0.CSIHnPWR = 0, writing to and reading from these bits is effective in modes other than FIFO mode (transmit-only buffer mode, dual buffer mode, or direct access mode). When CSIHnCTL0.CSIHnPWR = 1, writing to these bits has no effect in modes other than FIFO mode (transmit-only buffer mode, dual buffer mode, or direct access mode). Reading is only effective.

Table 11.43 Notes on Setting Registers (3/3)

Register	Bit	Content
CSIHnRX0H		These bits are initialized when CSIHnCTL0.CSIHnPWR is changed from 0 to 1 or from 1 to 0. When CSIHnCTL0.CSIHnPWR = 0, writing to and reading from these bits is prohibited in FIFO mode. When CSIHnCTL0.CSIHnPWR = 1, writing to these bits has no effect in FIFO mode. Reading is only effective. Writing to these bits has no effect in modes other than FIFO mode (transmit-only buffer mode, dual buffer mode, or direct access mode) regardless of the setting of CSIHnCTL0.CSIHnPWR. Reading is only effective.
CSIHnBRSy		Modification of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0.

Section 12 Serial Communication Interface 3 (SCI3)

This section contains a generic description of the serial communication interface (SCI3).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of SCI3.

12.1 Features of RH850/C1M-A SCI3

12.1.1 Units

This LSI has the following number of SCI3 units.

Table 12.1 Number of Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of Units	3	3
Name	SCI3n (n = 0 to 2)	SCI3n (n = 0 to 2)

Table 12.2 Index

Index	Meaning
n	Throughout this section, the individual SCI3 units are identified by the index “n” (n = 0 to 2); for example, SCI3nRSR is the receive shift register.

12.1.2 Register Base Address

SCI3 base addresses are listed in the following table.

SCI3 register addresses are given as offsets from the base addresses in general.

Table 12.3 Register Base Address

Base Address Name	Base Address
<SCI30_base>	FFD9 0000_H
<SCI31_base>	FF79 1000_H
<SCI32_base>	FFD9 2000_H

12.1.3 Clock Supply

SCI3 clock is listed in following table.

Table 12.4 Clock Supply

Unit Name	Clock for the Unit	Internal Clock Name
SCI3n	PCLK	CLKC_LSB (non-modulated low-speed peripheral clock)
	SCKn	SCInSCK pins (for operation with external clock signals)

12.1.4 Interrupts and DMA

SCI3 interrupt requests are listed in the following table.

Table 12.5 Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt Number	DMAC Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
SCI30						
ERI	Receive error	289	—	—	—	—
RXI	Receive data full	290	120	—	120	—
TXI	Transmit data empty	291	121	—	121	—
TEI	Transmit complete	292	—	—	—	—
SCI31						
ERI	Receive error	293	—	—	—	—
RXI	Receive data full	294	122	—	122	—
TXI	Transmit data empty	295	123	—	123	—
TEI	Transmit complete	296	—	—	—	—
SCI32						
ERI	Receive error	297	—	—	—	—
RXI	Receive data full	298	124	—	124	—
TXI	Transmit data empty	299	125	—	125	—
TEI	Transmit complete	300	—	—	—	—

Note: —: No number is assigned.

Note 1. 1st: Primary channel, 2nd: Secondary channel

12.1.5 Reset Sources

SCI3 reset sources are listed in the following table. SCI3 is initialized by these reset sources.

Table 12.6 Reset Source

Unit Name	Reset Source
SCI3n	Reset by any reset source

12.1.6 External Input/Output Signals

External input/output signals of SCI3 are listed in the following table.

Table 12.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
SCKn	Serial clock input/output signal	SCInSCK
RxDn	Receive data input signal	SCInRXD
TxDn	Transmit data output signal	SCInTXD

12.2 Overview

12.2.1 Functional Overview

The SCI3 can handle two methods of serial communications: asynchronous and clock synchronous. Asynchronous serial data transfer can be handled with standard LSI chips for asynchronous communication such as Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). Asynchronous mode also provides for serial communications between multiple processors (multi-processor communications).

In clock synchronous transfer, the interface serves as the master in communications through the output of its internal clock signal for synchronization, or as the slave in communications, where transfer is synchronized with the signal from an external clock.

Table 12.8 List of Functions

Function	Asynchronous Communications	Clock Synchronous Communications	
		Internal Clock (Master Mode)	External Clock (Slave Mode)
Full-duplex communication	Possible Independent transmitter unit and receiver unit allow simultaneous transmission and reception. Both the transmitter and receiver have a double-buffered structure, enabling continuous data transmission and reception.		
SCKn pin	Possible of clock output	Clock output	Clock input
Transmit/receive clock source	Internal clock by the on-chip baud rate generator		External clock
LSB/MSB first	Selectable (except for 7-bit length data)	Selectable	
Interrupt and DMA/DTS transfer	• Transmission complete • Transmission data empty • Reception data full • Reception error DMA and DTS can be activated by a transmit data empty interrupt signal or reception data full interrupt signal.		
Bit rate modulation	Possible	Possible (except for setting of the maximum speed)	
Data length	Selectable: 7- or 8-bit	8-bit	
Stop bit length	Selectable: 1- or 2-bit	—	
Parity	Selectable: with even/odd parity or none	—	
Detection of reception errors	• Parity error • Overrun error • Framing error	• Overrun error	
Break detection	Detectable by reading a register when a framing error occurs	—	
Serial input data pin level check	Possible		
Multi-processor communication	Possible	—	
Double-speed operation	Possible	—	
Maximum bit rate	5 Mbps (PCLK/8)	5 Mbps (PCLK/8)	3.3 Mbps (PCLK/12)

12.2.2 Block Diagram

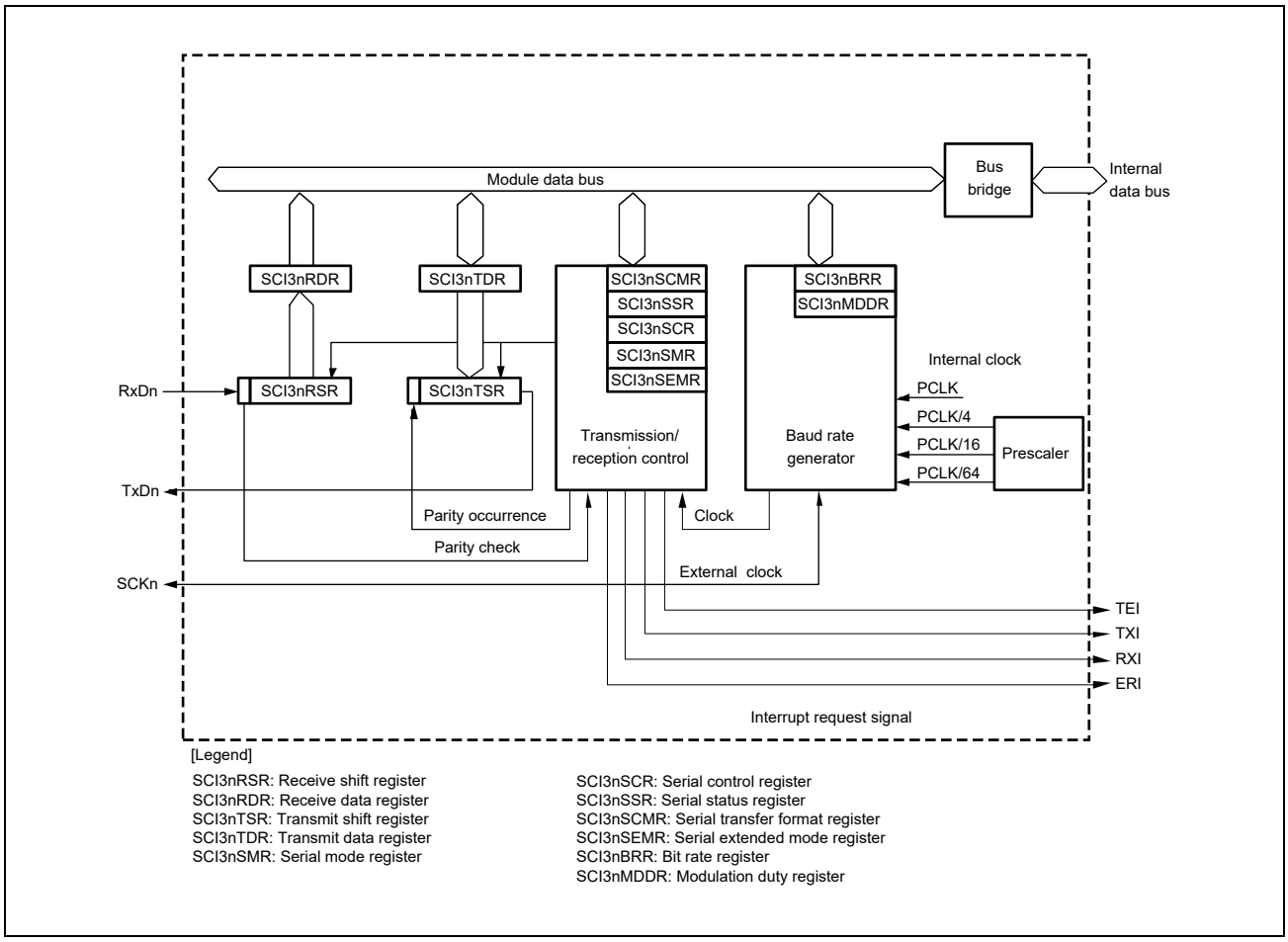


Figure 12.1 SCI3 Block Diagram

12.3 Registers

12.3.1 List of Registers

SCI3 registers are listed in the following table.

For information on <SCI3n_base>, see **Section 12.1.2, Register Base Address**.

Table 12.9 Register Configuration

Module Name	Register Name	Symbol	Address
SCI3n	Serial mode register	SCI3nSMR	<SCI3n_base>+ 00_H
SCI3n	Bit rate register / Modulation duty register*1	SCI3nBRR/ SCI3nMDDR	<SCI3n_base>+ 04_H
SCI3n	Serial control register	SCI3nSCR	<SCI3n_base>+ 08_H
SCI3n	Transmit data register	SCI3nTDR	<SCI3n_base>+ $0C_H$
SCI3n	Serial status register	SCI3nSSR	<SCI3n_base>+ 10_H
SCI3n	Receive data register	SCI3nRDR	<SCI3n_base>+ 14_H
SCI3n	Serial transfer format register	SCI3nSCMR	<SCI3n_base>+ 18_H
SCI3n	Serial extended mode register	SCI3nSEMR	<SCI3n_base>+ $1C_H$

Note 1. SCI3nBRR and SCI3nMDDR are assigned to the same address. These registers are switched by using the SCI3nSEMR.MDDRS bit.

12.3.2 SCI3nRDR — Receive Data Register

This register is used to store receive data. When one frame of data has been received, it is transferred from SCI3nRSR to this register allowing SCI3nRSR to receive the next data. SCI3nRSR and SCI3nRDR function as a double-buffer in this way, allowing continuous receive operations. Be sure to check that the SCI3nSSR.RDRF bit is set to 1 before reading SCI3nRDR. When the data length is 7 bits, receive data is stored in bit 0 to 6 and bit 7 is fixed to 0 regardless of the SCI3nSCMR.SINV bit.

Access: This register can be read in 8-bit units.

Address: <SCI3n_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	SCI3nRDR							
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 12.10 SCI3nRDR Register Contents

Bit Position	Bit Name	Function
7 to 0	SCI3nRDR	Receive Data Register These bits store receive data.

12.3.3 SCI3nTDR — Transmit Data Register

This register is used to store transmit data. When SCI3nTSR empty is detected, the transmit data written to SCI3nTDR is transferred to SCI3nTSR and transmission starts. The double-buffered structure of SCI3nTDR and SCI3nTSR allows continuous serial transmission. If the next transmit data has been written to SCI3nTDR when one frame of data is sent, the transmit data is transferred to SCI3nTSR to continue transmission. Be sure to check that the SCI3nSSR.TDRE bit is set to 1 before writing transmit data to SCI3nTDR.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + $0C_H$

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	SCI3nTDR							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12.11 SCI3nTDR Register Contents

Bit Position	Bit Name	Function
7 to 0	SCI3nTDR	Transmit Data Register These bits store transmit data.

12.3.4 SCI3nSMR — Serial Mode Register

SCI3nSMR is a register used to select the transfer format and the clock source for the on-chip baud rate generator.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 00_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CM	CHR	PE	PM	STOP	MP	CKS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. Writable only when TE = RE = 0.

Table 12.12 SCI3nSMR Register Contents (1/2)

Bit Position	Bit Name	Function
7	CM	Communications Mode 0: Asynchronous mode 1: Clock synchronous mode
6	CHR	Character Length (Valid only in asynchronous mode) 0: Selects 8 bits as the data length. 1: Selects 7 bits as the data length. LSB-first is fixed and the MSB (bit 7) in SCI3nTDR is not transmitted in transmission. In clock synchronous mode, the data length is fixed to 8 bits.
5	PE	Parity Enable (Valid only in asynchronous mode) When this bit is set to 1, a parity bit is added to transmit data and the parity bit is checked in reception. Regardless of the setting of this bit, no parity bit is added or checked in the multi-processor format.
4	PM	Parity Mode (Valid only when PE = 1 in asynchronous mode) 0: Even parity 1: Odd parity When even parity is set, the value of the appended parity bit causes the total number of 1-valued bits in a transmitted or received character plus the parity bit to be even. Similarly, when odd parity is set, the value of the appended parity bit causes the total number of 1-valued bits in a transmitted or received character plus the parity bit to be odd.
3	STOP	Stop Bit Length (Valid only in asynchronous mode) 0: 1 stop bit for transmission 1: 2 stop bits for transmission In reception, only the first stop bit is checked regardless of the setting of this bit. If the second stop bit is 0, it is treated as the start bit of the next transmission frame.
2	MP	Multi-Processor Mode (Valid only in asynchronous mode) When this bit is set to 1, the multi-processor communication function is enabled. In multi-processor mode, settings of the PE and PM bits are invalid.

Table 12.12 SCI3nSMR Register Contents (2/2)

Bit Position	Bit Name	Function
1, 0	CKS[1:0]	Clock Select 1, 0 These bits select the clock source for the on-chip baud rate generator. 00: PCLK clock (m = 0) 01: PCLK/4 clock (m = 1) 10: PCLK/16 clock (m = 2) 11: PCLK/64 clock (m = 3) For the relation between the setting of these bits and the baud rate, see **Section 12.3.9, SCI3nBRR — Bit Rate Register**. The character m is the decimal notation of the value of m in **Section 12.3.9, SCI3nBRR — Bit Rate Register**.

12.3.5 SCI3nSCR — Serial Control Register

SCI3nSCR is a register used for the transmission/reception control, interrupt control, and transmission/reception clock source selection listed below. For interrupt requests, see **Section 12.4.5, Interrupt Sources**.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W*1	R/W*1	R/W	R/W	R/W*2	R/W*2

Note 1. While the CM bit in SCI3nSMR is 1, a value of 1 can be written only when TE = 0 and RE = 0. After the TE or RE bit is set to 1, only 0 can be written to TE and RE. While the CM bit in SCI3nSMR is 0, writing is enabled at any timing.

Note 2. Writable only when TE = 0 and RE = 0. Also, writable at the same time when TE = 0 and RE = 0 are written.

Table 12.13 SCI3nSCR Register Contents (1/2)

Bit Position	Bit Name	Function
7	TIE	Transmit Interrupt Enable When this bit is set to 1, TXI interrupt request is enabled. TXI interrupt request can be cancelled by reading 1 from the TDRE flag and then clearing the flag to 0, or clearing the TIE bit.
6	RIE	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt requests are enabled. RXI and ERI interrupt requests can be cancelled by reading 1 from the RDRF, FER, PER, or ORER flag and then clearing the flag to 0, or clearing the RIE bit.
5	TE	Transmit Enable When this bit is set to 1, transmission is enabled. In this state, serial transmission is started by writing transmit data to SCI3nTDR and clearing the TDRE flag in SCI3nSSR to 0. Be sure to set SCI3nSMR before setting the TE bit to 1 to determine the transmission format. When this bit is set to 0 to disable transmission, the TDRE flag in SCI3nSSR is fixed to 1.

Table 12.13 SCI3nSCR Register Contents (2/2)

Bit Position	Bit Name	Function
4	RE	Receive Enable When this bit is set to 1, reception is enabled. In this state, serial reception is started by detecting a start bit in asynchronous mode or the input of synchronous clock in clock synchronous mode. Be sure to set SCI3nSMR before setting the RE bit to 1 to determine the reception format. Even if reception is disabled by clearing this bit, the RDRF, FER, PER, and ORER flags are not affected and the previous value is retained.
3	MPIE	Multiprocessor Interrupt Enable (Valid only when MP in SCI3nSMR = 1 in asynchronous mode) When this bit is set to 1 and the data with the multi-processor bit set to 0 is received, the data is not read and setting the RDRF, FER, and ORER flags in SCI3nSSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, this bit is automatically cleared to 0, and normal reception is resumed. For details, see **Section 12.4.2, Multi-Processor Communication Function**. When the receive data includes MPB = 0 in SCI3nSSR, the receive data is not transferred from SCI3nRSR to SCI3nRDR, a receive error is not detected, and setting the RDRF, FER, and ORER flags in SCI3nSSR to 1 is disabled. When the receive data includes MPB = 1 in SCI3nSSR, the MPB bit in SCI3nSSR is set to 1, the MPIE bit is automatically cleared to 0, the RXI and ERI interrupt requests are enabled (if the RIE bit in SCI3nSCR is set to 1), and setting the FER and ORER flags to 1 is enabled.
2	TEIE	Transmit End Interrupt Enable When this bit is set to 1, TEI interrupt request is enabled. TEI interrupt request can be cancelled by reading 1 from the TDRE flag and then clearing the flag to 0 to clear the TEND flag to 0, or clearing the TEIE bit.
1, 0	CKE[1:0]	Clock Enable 1, 0 These bits select the clock source and the SCKn pin function. For asynchronous mode 00: On-chip baud rate generator (The SCKn pin functions as an input/output port.) 01: On-chip baud rate generator (The clock with the same frequency as the bit rate is output from the SCKn pin.) 1X: Setting prohibited For clock synchronous mode 0X: Internal clock (The SCKn pin functions as a clock output pin.) 1X: External clock (The SCKn pin functions as a clock input pin.)

Note: X: Don’t care.

NOTE

When writing to any bit other than the MPIE bit of this register, use a store instruction such that the value of the MPIE bit becomes 0.

Note that when using a bit-manipulation instruction for writing to this register, the read-modify-write operation may inadvertently set the MPIE bit to 1.

12.3.6 SCI3nSSR — Serial Status Register

SCI3nSSR consists of SCI3 status flags and transfer multi-processor bits. The TDRE, RDRF, ORER, PER, and FER flags can only be cleared.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 10_H

Value after reset: 84_H

Bit	7	6	5	4	3	2	1	0
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Value after reset	1	0	0	0	0	1	0	0
R/W	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R/W

Note 1. Only 0 can be written to clear the flag.

Table 12.14 SCI3nSSR Register Contents (1/2)

Bit Position	Bit Name	Function
7	TDRE	Transmit Data Register Empty Indicates whether or not transmit data exists in SCI3nTDR. [Setting conditions] • When the TE bit in SCI3nSCR is 0 • When the data in SCI3nTDR has been transferred to SCI3nTSR so that new data can be written to SCI3nTDR [Clearing condition] • Writing 0 to TDRE after reading TDRE = 1 • When transmit data is written to SCI3nTDR while TE = 1
6	RDRF	Receive Data Register Full Indicates whether or not receive data exists in SCI3nRDR [Setting condition] • When reception finishes successfully and the receive data is transferred from SCI3nRSR to SCI3nRDR [Clearing conditions] • Writing 0 to RDRF after reading RDRF = 1 • When data is read from SCI3nRDR Even when the RE bit in SCI3nSCR is cleared, the RDRF flag is not affected and the previous value is retained. Note that completing the reception of the next data with the RDRF flag set to 1 will cause an overrun error, resulting in the loss of the receive data.
5	ORER	Overrun Error Indicates that an overrun error has occurred during reception and the reception abended. [Setting condition] • When the next data is received while RDRF = 1 In SCI3nRDR, receive data prior to an overrun error is retained, but data received after the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be continued. In clock synchronous mode, serial transmission also cannot be continued. [Clearing condition] • Writing 0 to ORER after reading ORER = 1 Even when the RE bit in SCI3nSCR is cleared, the ORER flag is not affected and the previous value is retained.

Table 12.14 SCI3nSSR Register Contents (2/2)

Bit Position	Bit Name	Function
4	FER	Framing Error Indicates that a framing error has occurred during reception in asynchronous mode and the reception abended. [Setting condition] • When the stop bit is 0 In 2-stop-bit mode, only whether the first stop bit is 1 is checked but the second stop bit is not checked. Although the receive data when a framing error has occurred is transferred to SCI3nRDR, the RDRF flag is not set. In addition, while the FER flag is set to 1, the subsequent receive data is not transferred to SCI3nRDR. [Clearing condition] • Writing 0 to FER after reading FER = 1 Even when the RE bit in SCI3nSCR is cleared to 0, the FER flag is not affected and the previous value is retained.
3	PER	Parity Error Indicates that a parity error has occurred during reception in asynchronous mode and the reception abended. [Setting condition] • Although the receive data when a parity error has occurred is transferred to SCI3nRDR, the RDRF flag is not set. In addition, while the PER flag is set to 1, the subsequent receive data is not transferred to SCI3nRDR. [Clearing condition] • Writing 0 to PER after reading PER = 1 Even when the RE bit in SCI3nSCR is cleared to 0, the PER flag is not affected and the previous value is retained.
2	TEND	Transmit End [Setting conditions] • When the TE bit in SCI3nSCR is 0 • When the TDRE flag is 1 while the last bit of a transmit character is being transmitted [Clearing condition] • Writing 0 to the TDRE flag after reading TDRE = 1 • When transmit data is written to SCI3nTDR while TE = 1
1	MPB	Multi-processor Bit Holds the value of the multi-processor bit in the received frame.
0	MPBT	Multi-processor Bit Transfer Sets the value of the multi-processor bit to be added to the transmit frame.

12.3.7 SCI3nSCMR — Serial Transfer Format Register

SCI3nSCMR is a register to select transfer format for both asynchronous mode and clock synchronous mode.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 18_H

Value after reset: $F2_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	SDIR	SINV	ASTLS	—
Value after reset	1	1	1	1	0	0	1	0
R/W	R	R	R	R	R/W*1	R/W*1	R/W*1	R

Note 1. Writable only when TE = 0 and RE = 0.

Table 12.15 SCI3nSCMR Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	SDIR	Serial Data Transfer Direction (Valid in asynchronous mode and clock synchronous mode) This bit is used to select the direction of serial/parallel conversion. 0: Transfer with LSB-first 1: Transfer with MSB-first This bit is valid only when the transfer format is 8-bit data. For 7-bit data, LSB first transfer is used.
2	SINV	Serial Data Invert (Valid in asynchronous mode and clock synchronous mode) This bit is used to invert the transfer data logic level. The SINV bit does not affect the logic level of the parity bit. To invert the parity bit, invert the PM bit in SCI3nSMR. 0: SCI3nTDR data is transmitted as it is, and receive data is stored in SCI3nRDR as it is. 1: SCI3nTDR data is inverted and transmitted, and receive data is inverted and stored in SCI3nRDR.
1	ASTLS	Asynchronous start bit level detection select (Valid only in asynchronous mode) 0: The beginning of a start bit is detected as a falling edge of the signal on the RxDn input pin. 1: The beginning of a start bit is detected as the low level of the signal on the RxDn input pin.
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

12.3.8 SCI3nSEMR — Serial Extended Mode Register

SCI3nSEMR is a register to select a 1-bit period.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + $1C_H$

Value after reset: 04_H

Bit	7	6	5	4	3	2	1	0
	BRME	MDDRS	—	BGDM	ABCS	RXDMON	—	—
Value after reset	0	0	0	0	0	1	0	0
R/W	R/W*1	R/W*1	R	R/W*1	R/W*1	R	R	R

Note 1. Writable only when TE = 0 and RE = 0.

Table 12.16 SCI3nSEMR Register Contents

Bit Position	Bit Name	Function
7	BRME	Bit Rate Modulation Enable When this bit is set to 1, the bit rate modulation function is enabled.
6	MDDRS	Modulation Duty Register Select This bit is used to select an accessible register. 0: SCI3nBRR is accessible. 1: SCI3nMDDR is accessible.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	BGDM	Baud Rate Generator Double-Speed Mode Select (only valid in asynchronous mode) This bit selects the frequency of the clock output from the baud rate generator. 0: The Baud rate generator outputs the clock with its normal frequency. 1: The Baud rate generator outputs the clock with its frequency doubled (double-speed operation).
3	ABCS	Asynchronous Reference Clock Select (only valid in asynchronous mode) This bit is used to select the reference clock for 1-bit period. 0: Operates on the reference clock with a frequency of 16 times the transfer rate. 1: Operates on the reference clock with a frequency of 8 times the transfer rate (double-speed operation).
2	RXDMON	Serial Input Data Monitor This bit indicates the RxDn pin state. 0: RxDn pin state is the low level. 1: RxDn pin state is the high level.
1, 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

12.3.9 SCI3nBRR — Bit Rate Register

SCI3nBRR is an 8-bit register to adjust the bit rate. Since each SCI3 channel has an independent baud rate generator, different bit rates can be set for each channel. **Table 12.18** shows the relationship between the setting (N) in SCI3nBRR and the bit rate (B) in normal asynchronous mode and clock synchronous mode. The value after reset of SCI3nBRR is FF_H. SCI3nBRR is allocated at the same address as SCI3nMDDR and is selected when MDDRS in SCI3nSEMR is 0. This register is writable only when TE = 0 and RE = 0.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 04_H

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	BRR							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12.17 SCI3nBRR Register Contents

Bit Position	Bit Name	Function
7 to 0	BRR	Baud rate generator setting (0 ≤ N ≤ 255)

Table 12.18 Relationship between Setting N in SCI3nBRR and Bit Rate B

Mode	SCI3nSEMR Setting		Bit Rate	Mean Error
	BGDM Bit	ABCS Bit		
Asynchronous	0	0	$B=\frac{PCLK\times10^6}{64\times2^{2m-1}\times(N+1)}$	$Error\ (\%)=\left\{\frac{PCLK\times10^6}{B\times64\times2^{2m-1}\times(N+1)}-1\right\}\times100$
	1	0	$B=\frac{PCLK\times10^6}{32\times2^{2m-1}\times(N+1)}$	$Error\ (\%)=\left\{\frac{PCLK\times10^6}{B\times32\times2^{2m-1}\times(N+1)}-1\right\}\times100$
	0	1		
	1	1	$B=\frac{PCLK\times10^6}{16\times2^{2m-1}\times(N+1)}$	$Error\ (\%)=\left\{\frac{PCLK\times10^6}{B\times16\times2^{2m-1}\times(N+1)}-1\right\}\times100$
Clock synchronous			$B=\frac{PCLK\times10^6}{8\times2^{2m-1}\times(N+1)}$	—

Note: B: Bit rate (bps)
N: SCI3nBRR setting for baud rate generator (0 ≤ N ≤ 255)
PCLK: Operating frequency (MHz)
m: Determined by the SCI3nSMR setting shown in the following table.

SCI3nSMR Setting		
CKS1	CKS0	m
0	0	0
0	1	1
1	0	2
1	1	3

Table 12.19 lists sample N settings of the SCI3nBRR register in asynchronous mode. **Table 12.20** lists the maximum settable bit rates.

Table 12.19 Examples of BRR Settings for Bit Rates (Asynchronous Mode)

	Operating Frequency PCLK = 40 (MHz)							
	SCI3nSEMR.ABCS = 0				SCI3nSEMR.ABCS = 1			
Bit Rate (bps)	m	N	Actual Bit Rate (bps)	Error (%)	m	N	Actual Bit Rate (bps)	Error (%)
110	3	177	109.73	– 0.25	—	—	—	—
150	3	129	150.24	0.16	3	255	152.59	1.73
300	3	64	300.48	0.16	3	129	300.48	0.16
600	2	129	600.96	0.16	3	64	600.96	0.16
1200	2	64	1201.92	0.16	2	129	1201.92	0.16
2400	1	129	2403.85	0.16	2	64	2403.85	0.16
4800	1	64	4807.69	0.16	1	129	4807.69	0.16
9600	0	129	9615.38	0.16	1	64	9615.38	0.16
19200	0	64	19230.77	0.16	0	129	19230.77	0.16
31250	0	39	31250.00	0.00	0	79	31250.00	0.00
38400	0	32	37878.79	– 1.36	0	64	38461.54	0.16

Note: This is an example where BGDM = 0.
When BGDM = 1, the bit rates are doubled.

Table 12.20 Maximum Bit Rate (Asynchronous Mode)

PCLK (MHz)	Setting				Maximum Bit Rate (bps)
	BGDM Setting	ABCS Setting	m	N	
40	0	0	0	0	1250000
	0	1	0	0	2500000
	1	0	0	0	2500000
	1	1	0	0	5000000

Table 12.21 lists sample N settings of the SCI3nBRR register in clock synchronous mode.

Table 12.21 Examples of Bit Rate Settings for Clock Synchronous Mode

Bit Rate (bps)	Operating Frequency PCLK = 40 (MHz)		
	m	N	Actual Bit Rate (bps)
1k	3	155	1001.60
2.5k	3	62	2480.16
5k	2	124	5000.00
10k	2	62	9920.63
25k	1	99	25000.00
50k	1	49	50000.00
100k	0	99	100000.00
250k	0	39	250000.00
500k	0	19	500000.00
1M	0	9	1000000.00
2M	0	4	2000000.00
2.5M	0	3	2500000.00
5M	0	1	5000000.00

Table 12.22 Maximum Bit Rate when Internal Clock Is Output (Clock Synchronous Mode)

PCLK (MHz)	m	N	Maximum Bit Rate (bps)
40	0	1	5000000.00

12.3.10 SCI3nMDDR — Modulation Duty Register

SCI3nMDDR is a register to correct the bit rate adjusted by SCI3nBRR. The value after reset of SCI3nMDDR is FF_H. When the BRME bit in SCI3nSEMR is set to 1, the bit rate generated by the on-chip baud rate generator is corrected to SCI3nMDDR/256 on average. **Table 12.24** shows the relationship between the SCI3nMDDR setting and the bit rate B. SCI3nMDDR is allocated at the same address as SCI3nBRR and is selected when MDDRS in SCI3nSEMR is 1. This register is writable only when TE = 0 and RE = 0. Bit 7 is fixed to 1.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 04_H

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	MDDR							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12.23 SCI3nMDDR Register Contents

Bit Position	Bit Name	Function
7 to 0	MDDR	Baud rate generator setting (128 ≤ MDDR ≤ 255)

Table 12.24 Relationship between SCI3nMDDR Setting and Bit Rate B when Bit Rate Modulation Function Is Used

Mode	SCI3nSEMR Setting		Bit Rate	Mean Error
	BGDM bit	ABCS bit		
Asynchronous	0	0	$B = \frac{PCLK \times 10^6}{64 \times 2^{2m-1} \times (256/MDDR) \times (N+1)}$	$Error\ (\%) = \left\{\frac{PCLK \times 10^6}{B \times 64 \times 2^{2m-1} \times (256/MDDR) \times (N+1)} - 1\right\} \times 100$
	1	0	$B = \frac{PCLK \times 10^6}{32 \times 2^{2m-1} \times (256/MDDR) \times (N+1)}$	$Error\ (\%) = \left\{\frac{PCLK \times 10^6}{B \times 32 \times 2^{2m-1} \times (256/MDDR) \times (N+1)} - 1\right\} \times 100$
	0	1		
	1	1	$B = \frac{PCLK \times 10^6}{16 \times 2^{2m-1} \times (256/MDDR) \times (N+1)}$	$Error\ (\%) = \left\{\frac{PCLK \times 10^6}{B \times 16 \times 2^{2m-1} \times (256/MDDR) \times (N+1)} - 1\right\} \times 100$
Clock synchronous			$B = \frac{PCLK \times 10^6}{8 \times 2^{2m-1} \times (256/MDDR) \times (N+1)}$	—

Note: B: Bit rate (bps)
N: SCI3nBRR setting of baud rate generator (0 ≤ N ≤ 255)
PCLK: Operating frequency (MHz)
m: See **Table 12.18, Relationship between Setting N in SCI3nBRR and Bit Rate B**.
MDDR: SCI3nMDDR setting (128 ≤ SCI3nMDDR ≤ 255)

12.4 Functions

12.4.1 Operation in Asynchronous Mode

Figure 12.2 shows the general format for asynchronous serial communications. One frame consists of a start bit (low level), transmit/receive data, a parity bit, and stop bits (high level). In asynchronous serial communications, the communication line is usually held in the mark state (high level). The SCI3 monitors the communication line, and when the SCI3 detects the space state (low level), it recognizes a start bit and starts serial communications. Inside the SCI3, the transmitter and the receiver are independent, enabling full-duplex communications. Both the transmitter and the receiver have a double-buffered structure so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.

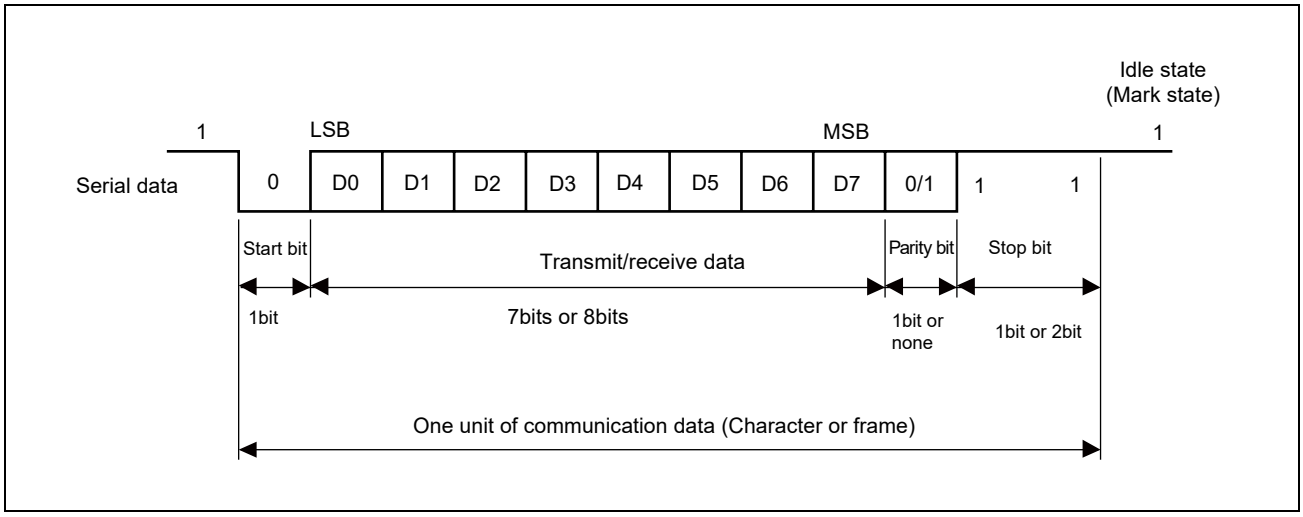


Figure 12.2 Data Format in Asynchronous Serial Communications (Example with 8- Bit Data, Parity, Two Stop Bits)

12.4.1.1 Transmission/Reception Format

Table 12.25 lists settable transmission/reception formats in asynchronous mode. Any of 12 transmission/reception formats can be selected according to the SCI3nSMR setting. For details of the multi-processor bit, see **Section 12.4.2, Multi-Processor Communication Function**.

Table 12.25 Serial Transmission/Reception Formats (Asynchronous Mode)

SMR Setting				Serial Transmission/Reception Format and Frame Length
CHR	PE	MP	STOP	1 2 3 4 5 6 7 8 9 10 11 12
0	0	0	0	S 8-bit data STOP
0	0	0	1	S 8-bit data STOP STOP
0	1	0	0	S 8-bit data P STOP
0	1	0	1	S 8-bit data P STOP STOP
1	0	0	0	S 7-bit data STOP
1	0	0	1	S 7-bit data STOP STOP
1	1	0	0	S 7-bit data P STOP
1	1	0	1	S 7-bit data P STOP STOP
0	–	1	0	S 8-bit data MPB STOP
0	–	1	1	S 8-bit data MPB STOP STOP
1	–	1	0	S 7-bit data MPB STOP
1	–	1	1	S 7-bit data MPB STOP STOP

Note: S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multi-processor bit

12.4.1.2 Receive Data Sampling Timing and Reception Margin

In asynchronous mode, the SCI3 operates on a reference clock with a frequency of 16 times (8 times when SCI3nSEMR.ABCS = 1) the bit rate. In reception, the SCI3 samples the falling edge of the beginning of the start bit (low level) using the reference clock and performs internal synchronization. As shown in **Figure 12.3**, data is latched at the middle of each bit by sampling receive data at the rising edge of the eighth pulse (fourth pulse when SCI3nSEMR.ABCS = 1) of the reference clock.

Thus the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1+F) \right| \times 100\ [\%] \quad \cdots \text{ Formula (1)}$$

M: Reception margin
N: Ratio of bit rate to clock (N = 16 when ABCS in SCI3nSEMR = 0, N = 8 when ABCS = 1)
D: Duty cycle of clock (D = 0.5 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute value of clock frequency deviation

Assuming that F (absolute value of clock frequency deviation) = 0, D (duty cycle of clock) = 0.5, and N = 16 in formula (1), the reception margin is obtained by the formula below.

$M = \{0.5 - 1 / (2 \times 16)\} \times 100\ [\%] = 46.875\%$

However, this is only the calculated value, and a margin of 20% to 30% should be allowed in system design.

When the bit rate modulation function is used, the reference clock frequency is corrected on average.

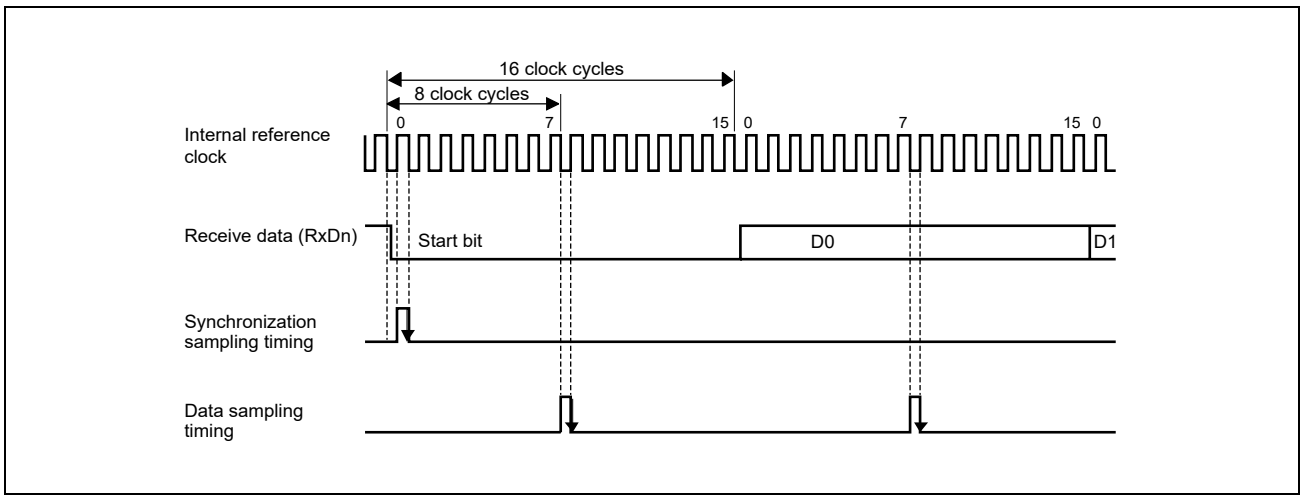


Figure 12.3 Receive Data Sampling Timing in Asynchronous Mode

12.4.1.3 Clock

An internal clock generated by the on-chip baud rate generator can be selected as the SCI3's transmission/reception clock according to the settings of the CM bit in SCI3nSMR and the CKE1 and CKE0 bits in SCI3nSCR. When the SCI3 is operated on an internal clock, the clock can be output from the SCKn pin.

For details of clock synchronous mode, see **Section 12.4.3, Operation in Clock Synchronous Mode**.

In asynchronous mode, the frequency of the clock output is equal to the bit rate and the phase is such that the rising edge of the clock comes at the center of the transmit data, as shown in **Figure 12.4**.

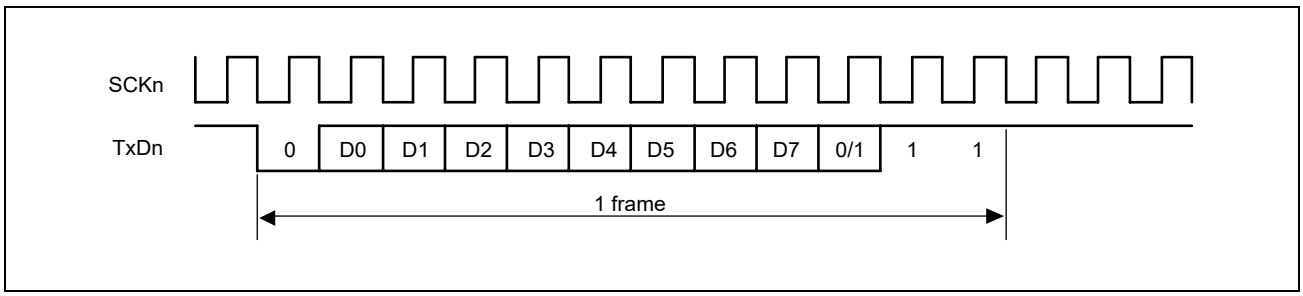


Figure 12.4 Phase Relationship between Output Clock and Transmit Data (Asynchronous Mode)

12.4.1.4 Double-Speed Operation

In addition to the operation described in **Section 12.4.1.3, Clock**, double-speed operation is enabled by the setting of the ABCS bit in SCI3nSEMR.

In double-speed operation, the same operation on a clock with a frequency of 16 times the bit rate in normal operation can be conducted on a clock with a frequency of 8 times the bit rate, meaning that the SCI3 operates on a double transfer rate using the same reference clock.

Setting the BGDM bit of the SCI3nSEMR register to 1 produces a signal derived by halving the frequency period of the reference clock, meaning that the SCI3 operates at double the bit rate when the setting of the BGDM bit is 0.

Selection of double-speed operation by the ABCS and BGDM bits has a multiplicative effect when the on-chip baud rate generator is selected while SCI3nSCR.CKE1 = 0. That is, setting the ABCS and BGDM bits to 1 enables operation at four times the bit rate when the ABCS and BGDM bits are both set to 0.

12.4.1.5 SCI3 Initialization (Asynchronous Mode)

Before transmitting and receiving data, clear the TE and RE bits in SCI3nSCR and then initialize the SCI3 according to the sample flowchart in **Figure 12.5**. Before changing the operating mode or transfer format, be sure to clear the TE and RE bits to 0. Note that clearing the TE bit to 0 sets the TDRE flag to 1, but clearing the RE bit to 0 initializes neither the RDRF, PER, FER, and ORER flags nor SCI3nRDR.

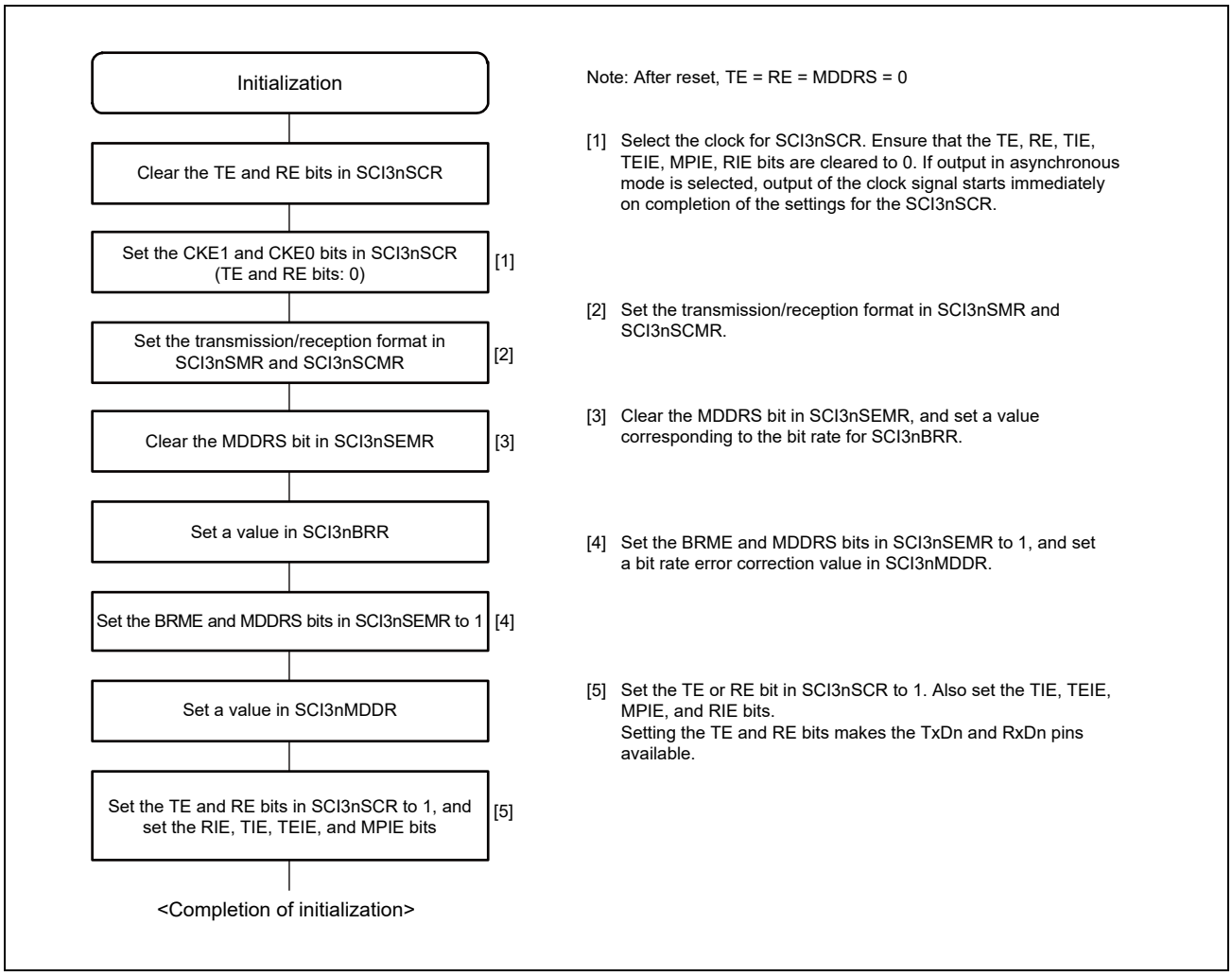


Figure 12.5 Sample Flowchart for SCI3 Initialization

12.4.1.6 Serial data transmission (asynchronous mode)

Figure 12.6 shows an example of operation for data transmission in asynchronous mode. In data transmission, the SCI3 operates as described below.

1. When transmit data is written to SCI3nTDR, the TDRE flag is automatically cleared to 0. The SCI3 monitors the TDRE flag in SCI3nSSR. When the flag is cleared, the SCI3 recognizes that data has been written to SCI3nTDR and transfers data from SCI3nTDR to SCI3nTSR. When writing transmit data to SCI3nTDR at a trigger of TXI interrupt request, set the TIE bit to 1 and then set the TE bit to 1 or set both TIE and TE bits simultaneously with one instruction to generate a TXI interrupt request for starting data transfer.
2. Transmission starts after data is transferred from SCI3nTDR to SCI3nTSR and the TDRE flag is set to 1. If the TIE bit in SCI3nSCR is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to SCI3nTDR in this TXI interrupt processing routine before transmission of the previously transferred data is completed. When using a TEI interrupt request, write the last transmit data to SCI3nTDR and then clear the TIE bit to 0 and set the TEIE bit to 1.
3. Data is sent from the TxDn pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.
4. The TDRE flag is checked when the stop bit is output.
5. When the TDRE flag is 0, the next transmit data is transferred from SCI3nTDR to SCI3nTSR. After the stop bit has been sent, transmission of the next frame starts.
6. When the TDRE flag is 1, the TEND flag in SCI3nSSR is set to 1, the stop bit is sent, and then 1 is output to enter the mark state. If the TEIE bit in SCI3nSCR is set to 1 at this time, a TEI interrupt request is generated.

Figure 12.7 shows a sample flowchart for data transmission. **Figure 12.8** shows a sample flowchart for stopping the SCI3 after data transmission.

Supplementary note on operation when data transmission in asynchronous mode is enabled

When the TE bit is set to 1, the high level (preamble) is output for a frame. When transmit data is written to SCI3nTDR while the preamble is output, the data is transferred from SCI3nTDR to SCI3nTSR following the completion of the preamble output.

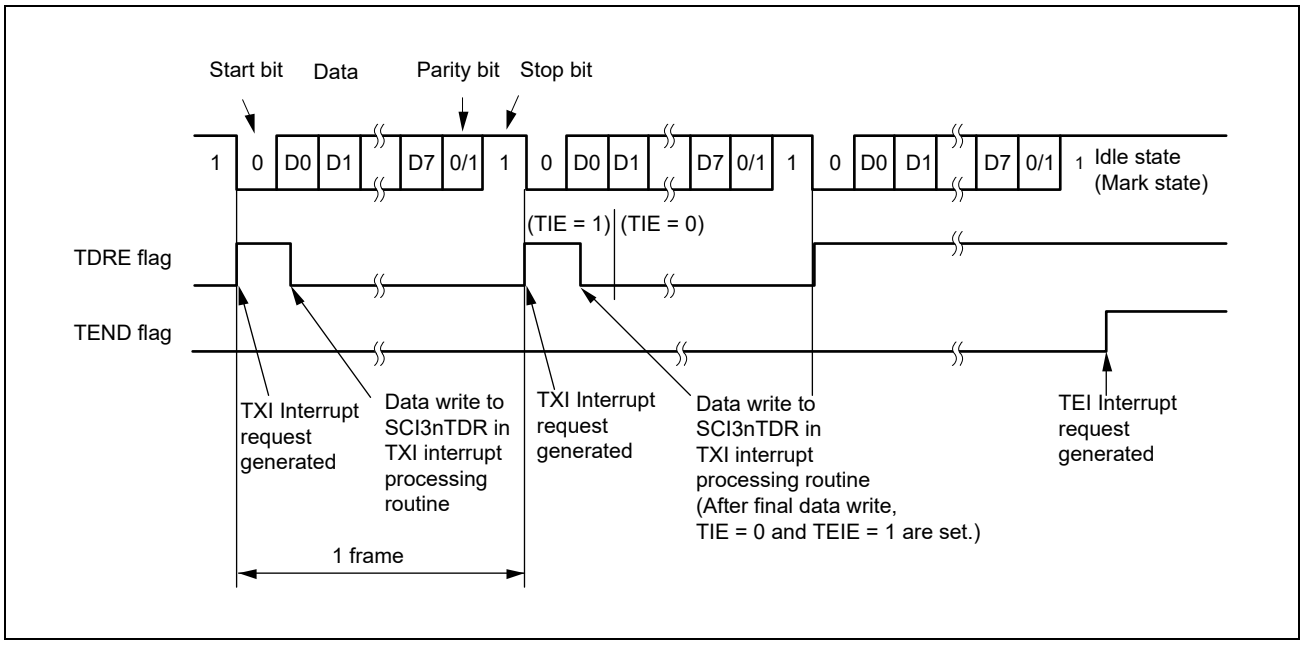


Figure 12.6 Example of Operation for Transmission in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit)

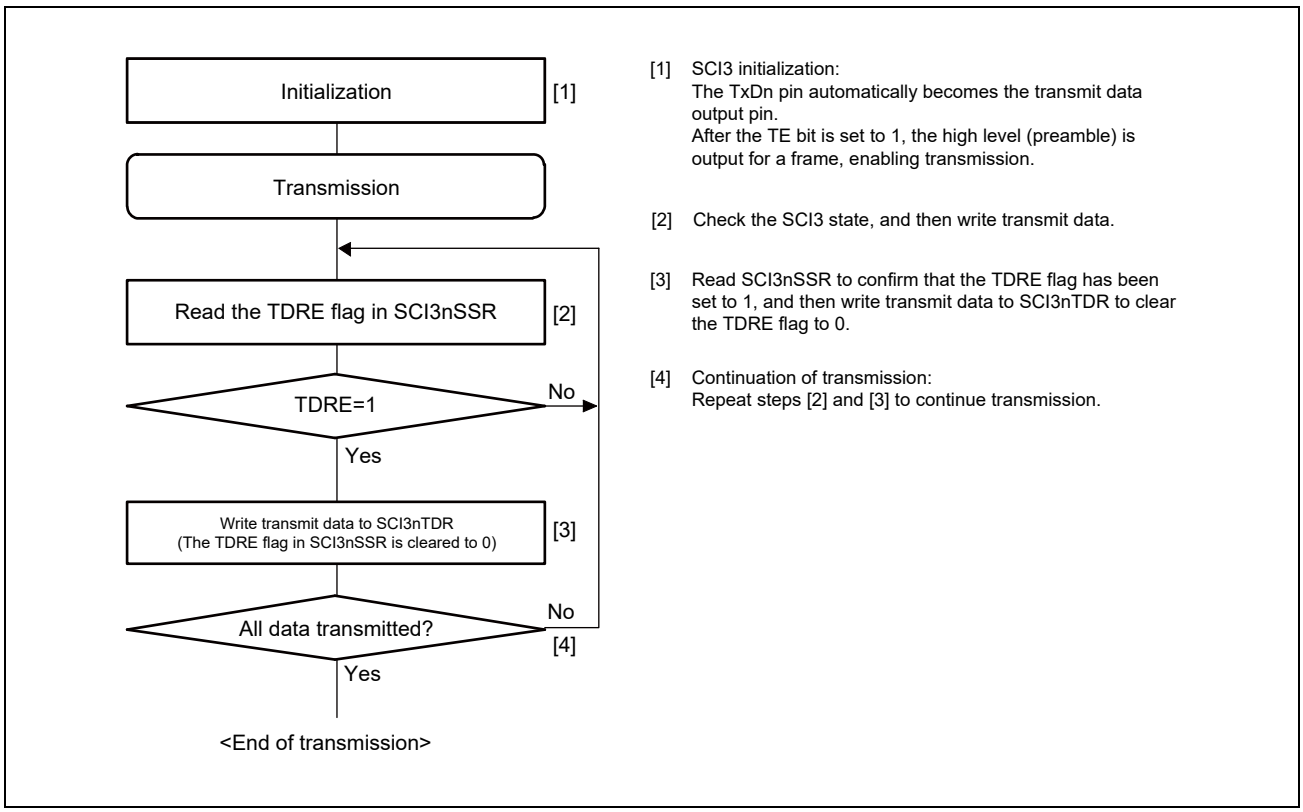


Figure 12.7 Example of Serial Transmission Flowchart

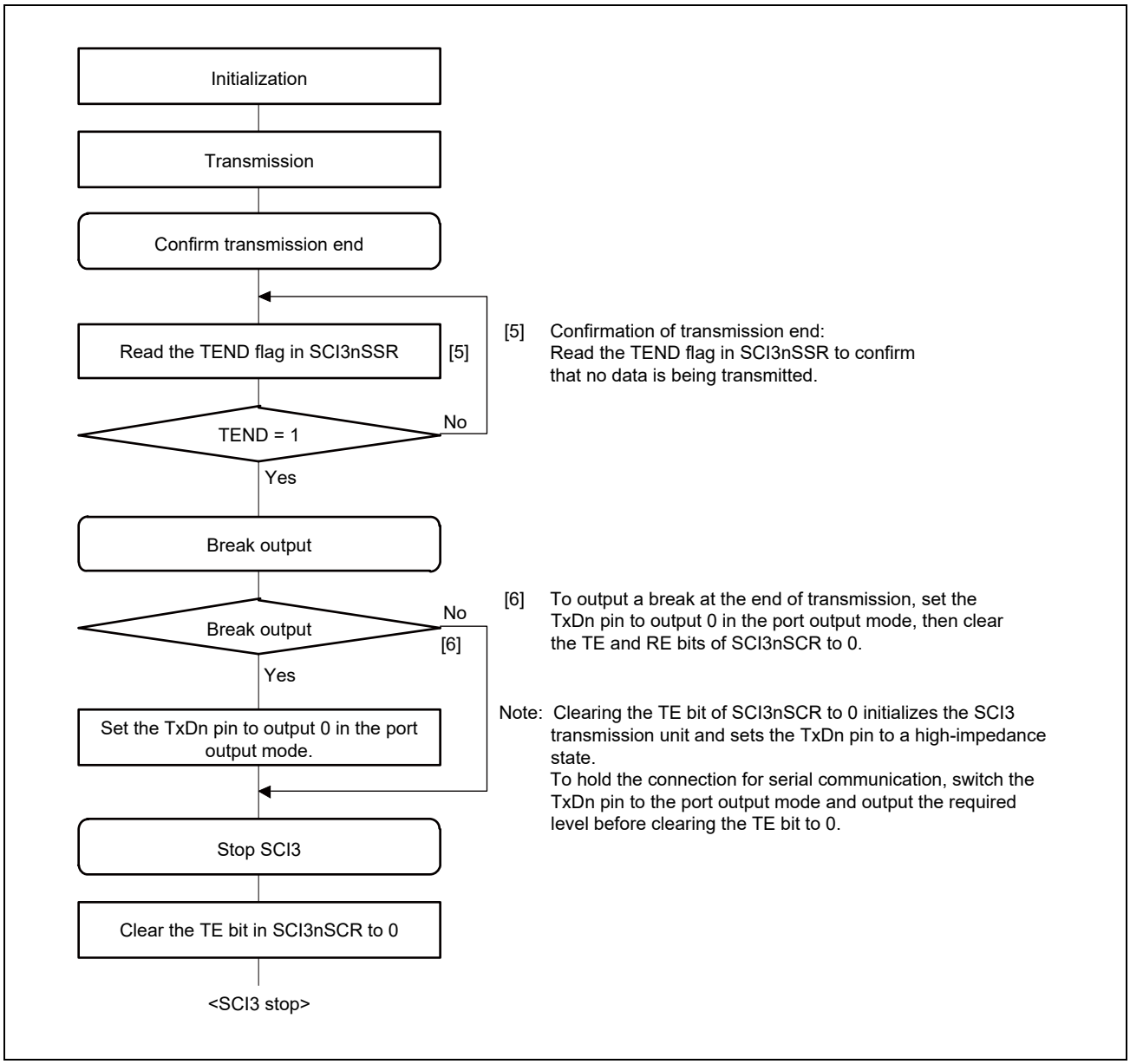


Figure 12.8 Example Flowchart for Stopping the SCI3 after Serial Transmission

12.4.1.7 Serial Data Reception (Asynchronous Mode)

Figure 12.9 shows an example of the operation for data reception in asynchronous mode. In data reception, the SCI3 operates as described below.

1. When the SCI3 monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in SCI3nRSR, and checks the parity bit and the stop bit.
2. When an overrun error occurs (the next data has been received with the RDRF flag in SCI3nSSR set to 1), the ORER flag in SCI3nSSR is set to 1. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to SCI3nRDR. The RDRF flag retains the state of being set to 1.
3. When a parity error is detected, the PER flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated.
4. When a framing error (when the stop bit is 0) is detected, the FER flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated.
5. When reception finishes successfully, the RDRF flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to SCI3nRDR in this RXI interrupt processing routine before reception of the next receive data is completed. Reading SCI3nRDR automatically clears the RDRF flag to 0.

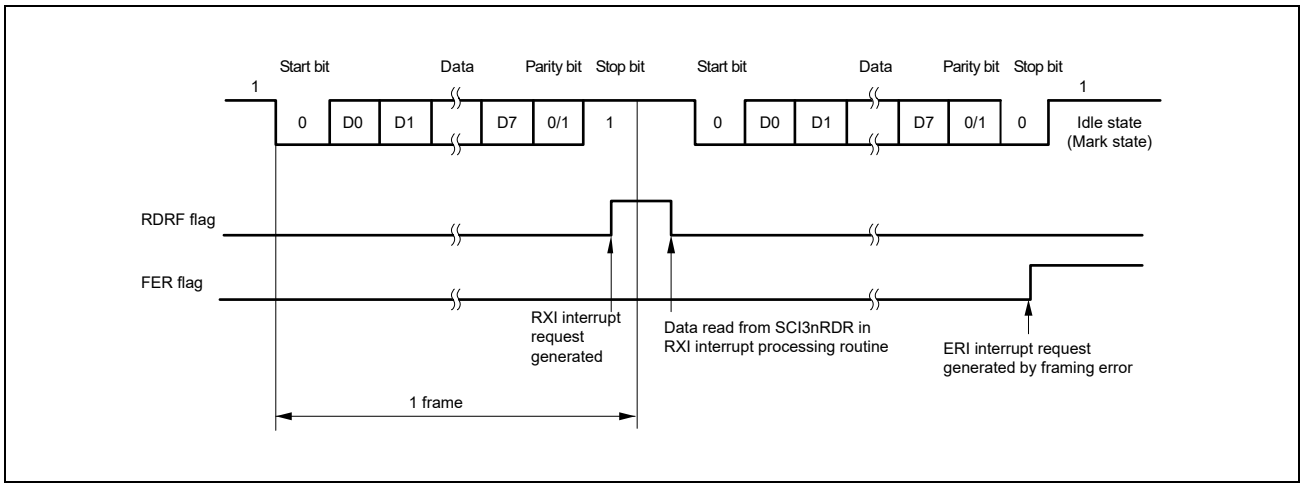


Figure 12.9 Example of SCI3 Reception (Example with 8-Bit Data, Parity, One Stop Bit)

Table 12.26 lists the states of the SCI3nSSR status flags and receive data handling when a receive error is detected. When a receive error is detected, the RDRF flag retains the status before receiving the data. Subsequent data reception is disabled while a receive error flag is set to 1. Accordingly, clear the ORER, FER, PER, and RDRF flags before continuing data reception. **Figure 12.10** shows a sample flowchart for data reception.

Table 12.26 SCI3nSSR Status Flags and Receive Data Handling

SCI3nSSR Status Flags					
RDRF*1	ORER	FER	PER	Receive Data	Receive Status
1	0	0	0	Transferred to SCI3nRDR	Successful reception
0	0	1	0	Transferred to SCI3nRDR	Framing error
0	0	0	1	Transferred to SCI3nRDR	Parity error
0	0	1	1	Transferred to SCI3nRDR	Framing error + parity error
1*	1	0	0	Lost	Overrun error
1*	1	1	0	Lost	Overrun error + framing error
1*	1	0	1	Lost	Overrun error + parity error
1*	1	1	1	Lost	Overrun error + framing error + parity error

Note 1. In the case of an overrun error, the RDRF flag retains the state before the data reception.

Note: A sign “+” indicates that two or more receive states occur simultaneously in a single reception operation.

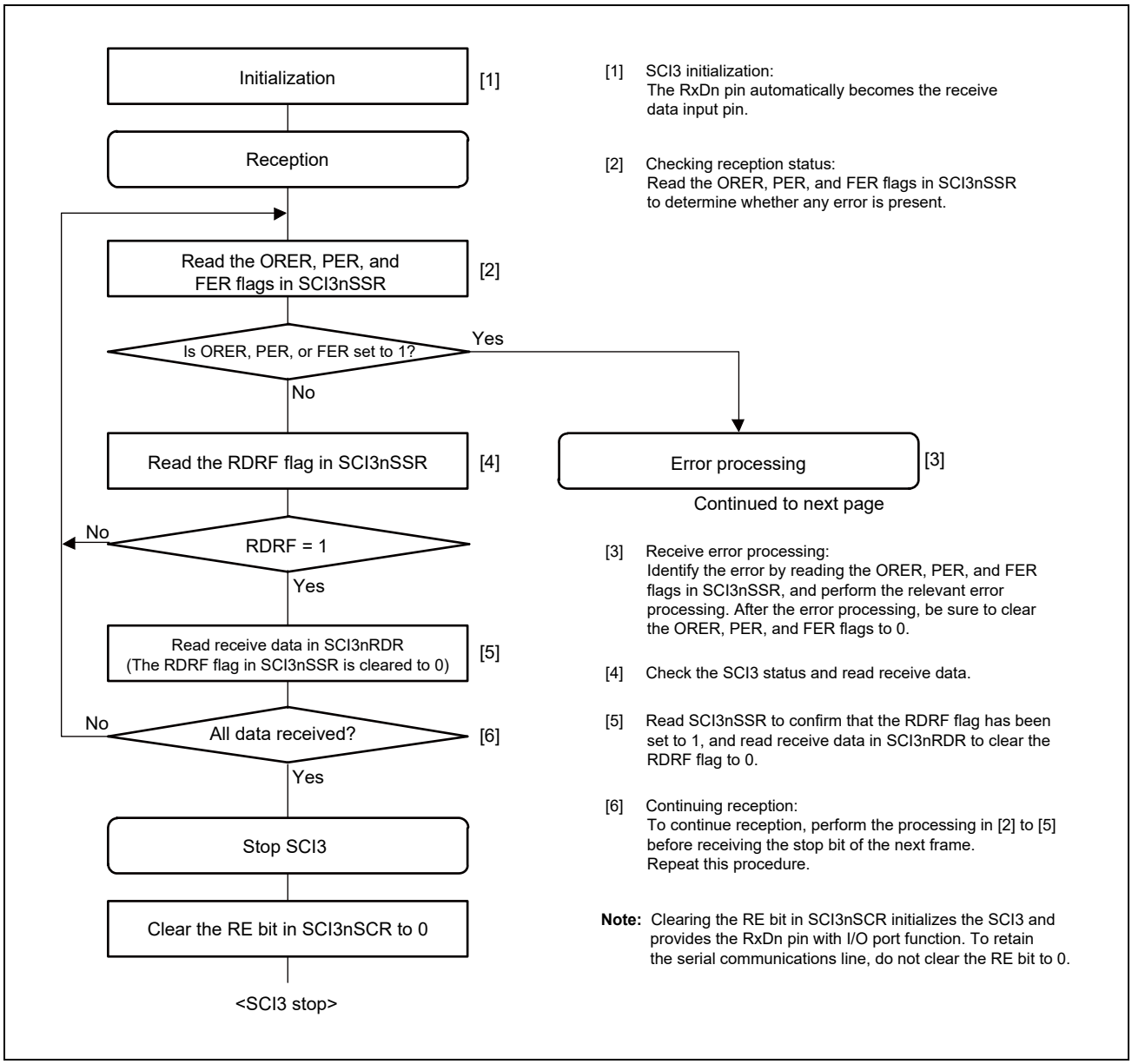


Figure 12.10 Example of Serial Reception Flowchart (1)

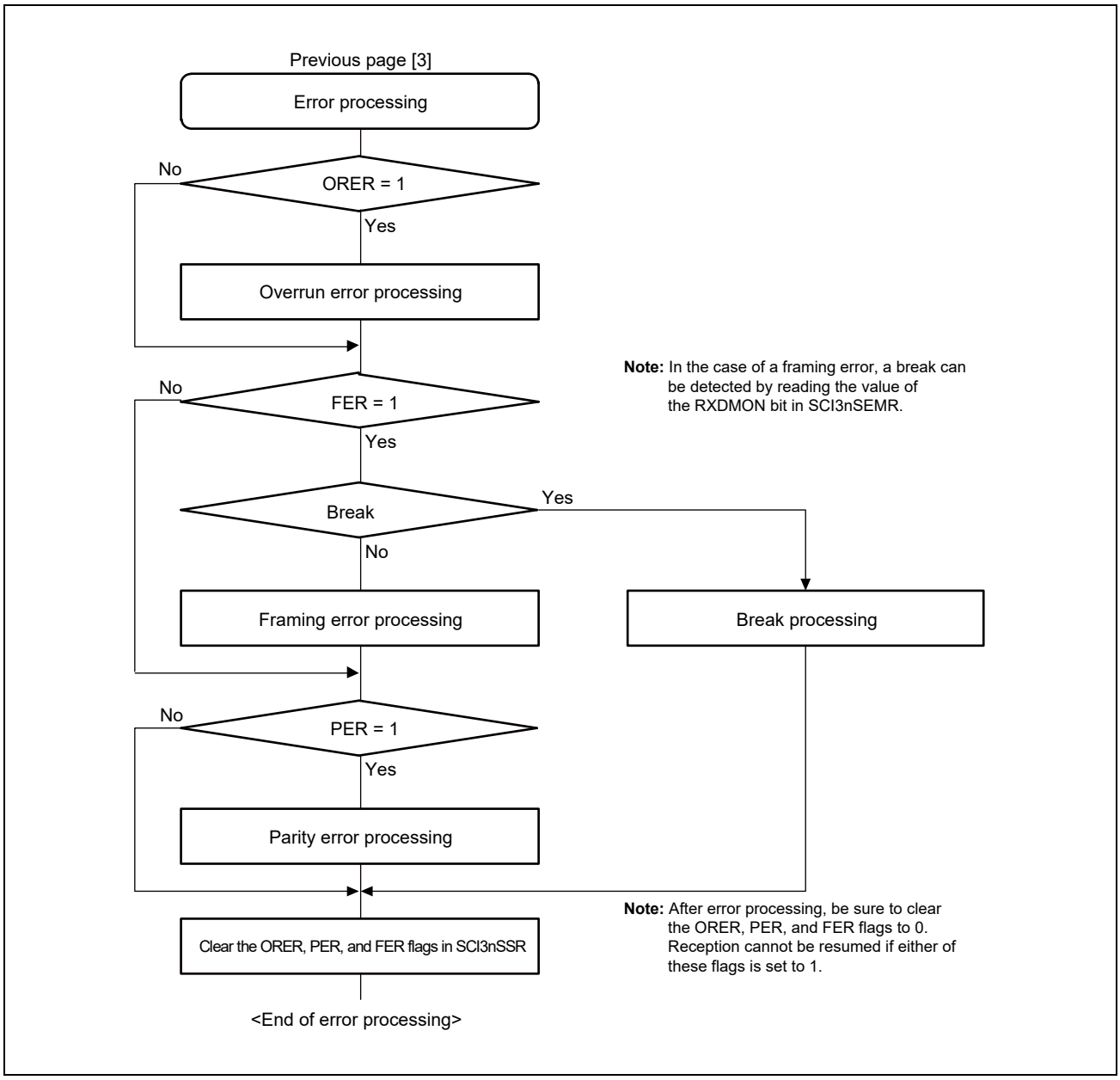


Figure 12.11 Example of Serial Reception Flowchart (2)

12.4.2 Multi-Processor Communication Function

12.4.2.1 Overview and Sample Connection

Using the multi-processor communication function allows data transmission and reception by sharing a communication line between multiple processors by using asynchronous serial communication in which the multi-processor bit is added. In multi-processor communication, a unique ID code is allocated to each receiving station. Serial communication cycles consist of an ID transmission cycle to specify the receiving station and a data transmission cycle to transmit data to the specified receiving station. The multi-processor bit is used to distinguish the ID transmission cycle from the data transmission cycle. When the multi-processor bit is set to 1, it indicates the ID transmission cycle. When the multi-processor bit is set to 0, it indicates the data transmission cycle. **Figure 12.12** shows an example of communication between processors by using the multi-processor format. First, a transmitting station sends communication data in which the multi-processor bit (= 1) is added to the ID code of the receiving station. Next, the transmitting station sends communication data in which the multi-processor bit (= 0) is added to the transmit data. Upon receiving the communication data in which the multi-processor bit is set to 1, the receiving station compares the received ID with the ID of the receiving station itself. When these IDs match, the receiving station receives communication data that is subsequently transmitted. If these IDs do not match, the receiving station skips communication data until it receives communication data in which the multi-processor bit is set to 1.

To support this function, the SCI3 provides the MPIE bit in SCI3nSCR. When the MPIE bit is set to 1, transfer of receive data from SCI3nRSR to SCI3nRDR, detection of a receive error, and setting the RDRF, FER, and ORER flags in SCI3nSSR are disabled until data in which the multi-processor bit is set to 1 is received. Upon receiving a character in which the multi-processor bit is set to 1, the MPB bit in SCI3nSSR is set to 1 and the MPIE bit is automatically cleared to 0, thus returning to a normal reception operation. When the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. While the MPIE bit is cleared to 0, reception operation is conducted regardless of the multi-processor bit value. The multi-processor bit is stored in the MPB bit in SCI3nSSR.

When the multi-processor format is specified, specification of the parity bit is disabled. Apart from this, there is no difference from the operation in the normal asynchronous mode. A clock that is used for multi-processor communications is also the same as the clock used in the normal asynchronous mode.

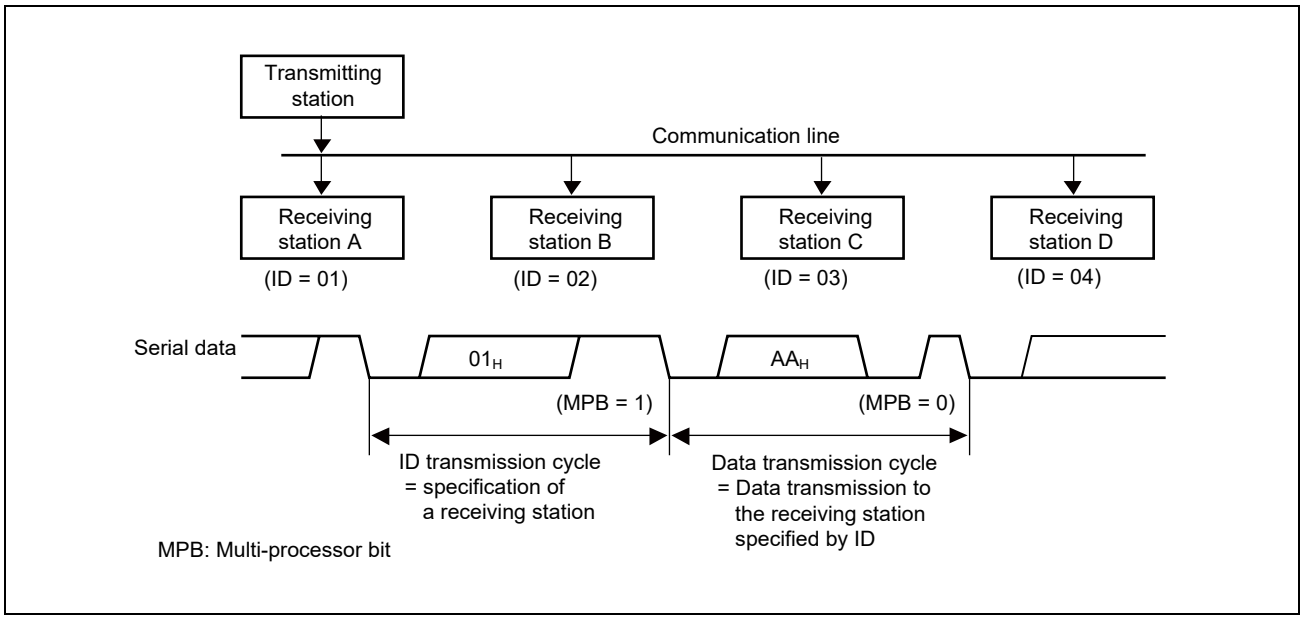


Figure 12.12 Example of Communication Using the Multi-Processor Format (Example of Transmission of Data AA_H to Receiving Station A)

12.4.2.2 Multi-processor serial data transmission

Figure 12.13 shows a sample flowchart of multi-processor data processing. In the ID transmission cycle, send the ID with the MPBT bit in SCI3nSSR set to 1. In the data transmission cycle, send data with the MPBT bit in SCI3nSSR cleared to 0. Other operations are the same as operations in asynchronous mode.

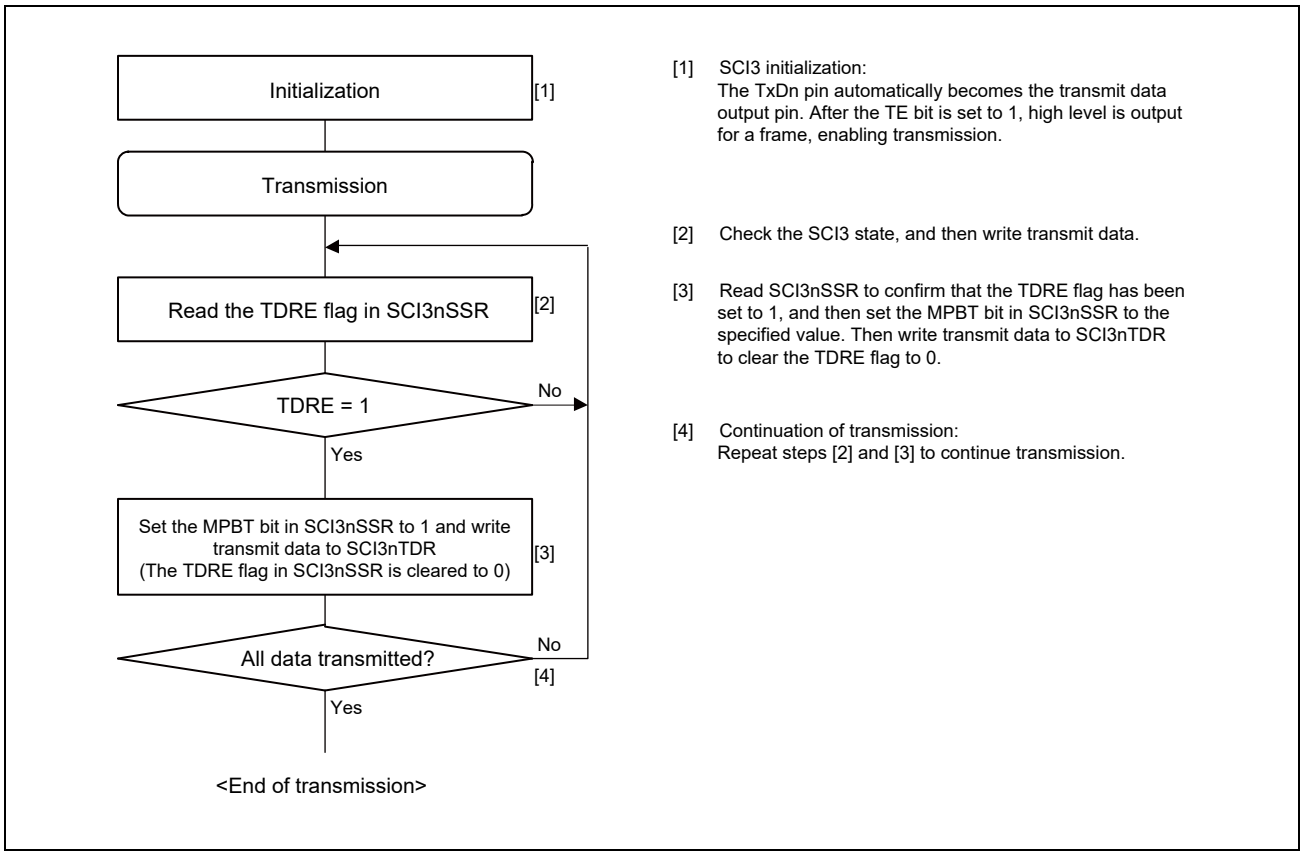


Figure 12.13 Example of Multi-Processor Serial Transmission Flowchart

12.4.2.3 Multi-processor serial data reception

Figure 12.15 to **Figure 12.17** shows sample flowcharts of multi-processor data reception. When the MPIE bit in SCI3nSCR is set to 1, reading the communication data is skipped until communication data in which the multi-processor bit is set to 1 is received. When communication data in which the multi-processor bit is set to 1 is received, the receive data is transferred to SCI3nRDR. At this time, an RXI interrupt request is generated. Other operations are the same as operations in asynchronous mode. **Figure 12.14** shows an example of operation for reception.

CAUTION

Do not write data to SCI3nSCR when communication data in which the multi-processor bit is set to 1 is received. The MPIE bit may not become the desired state.

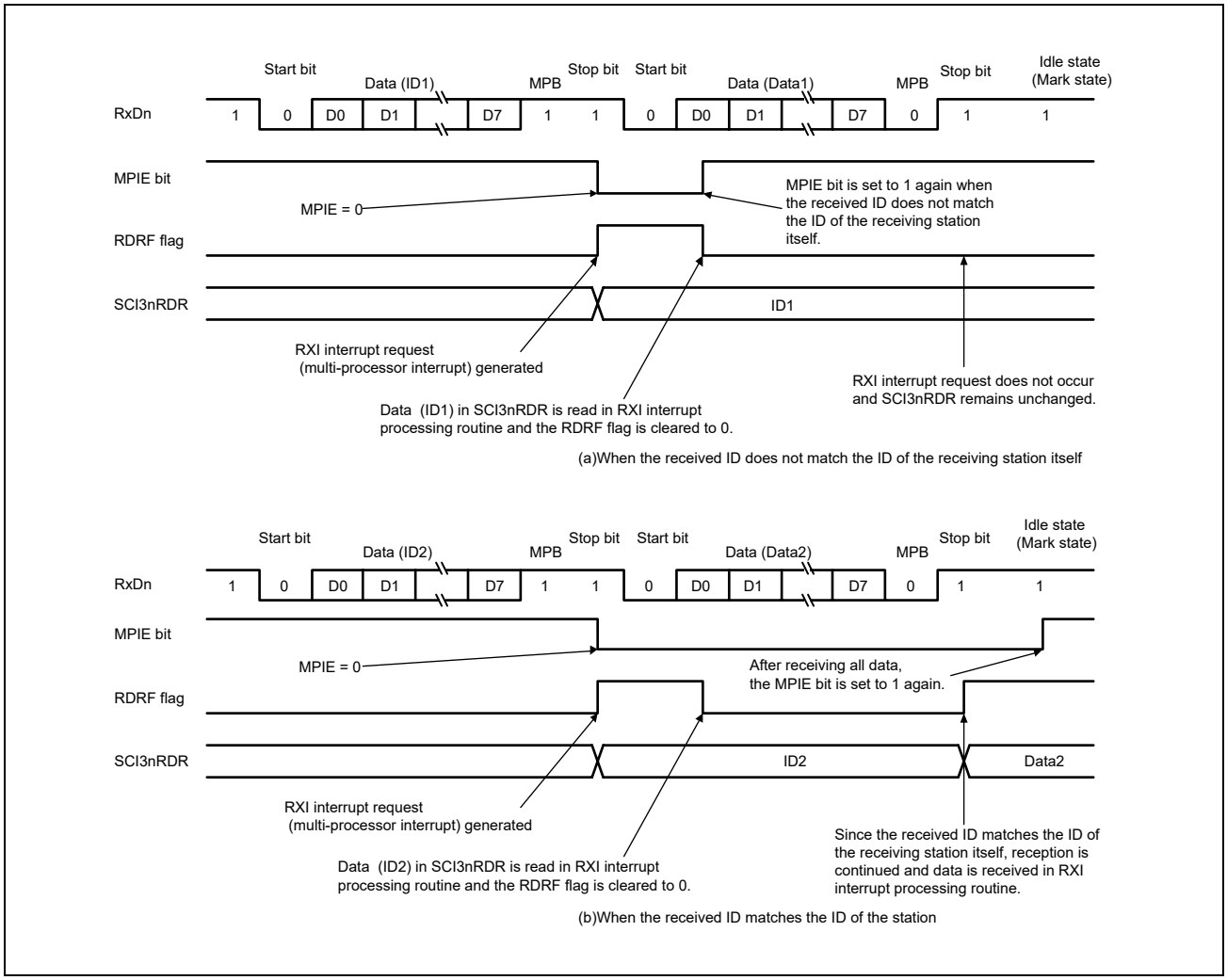


Figure 12.14 Example of SCI3 Reception (8-Bit Data, Multi-Processor Bit, One Stop Bit)

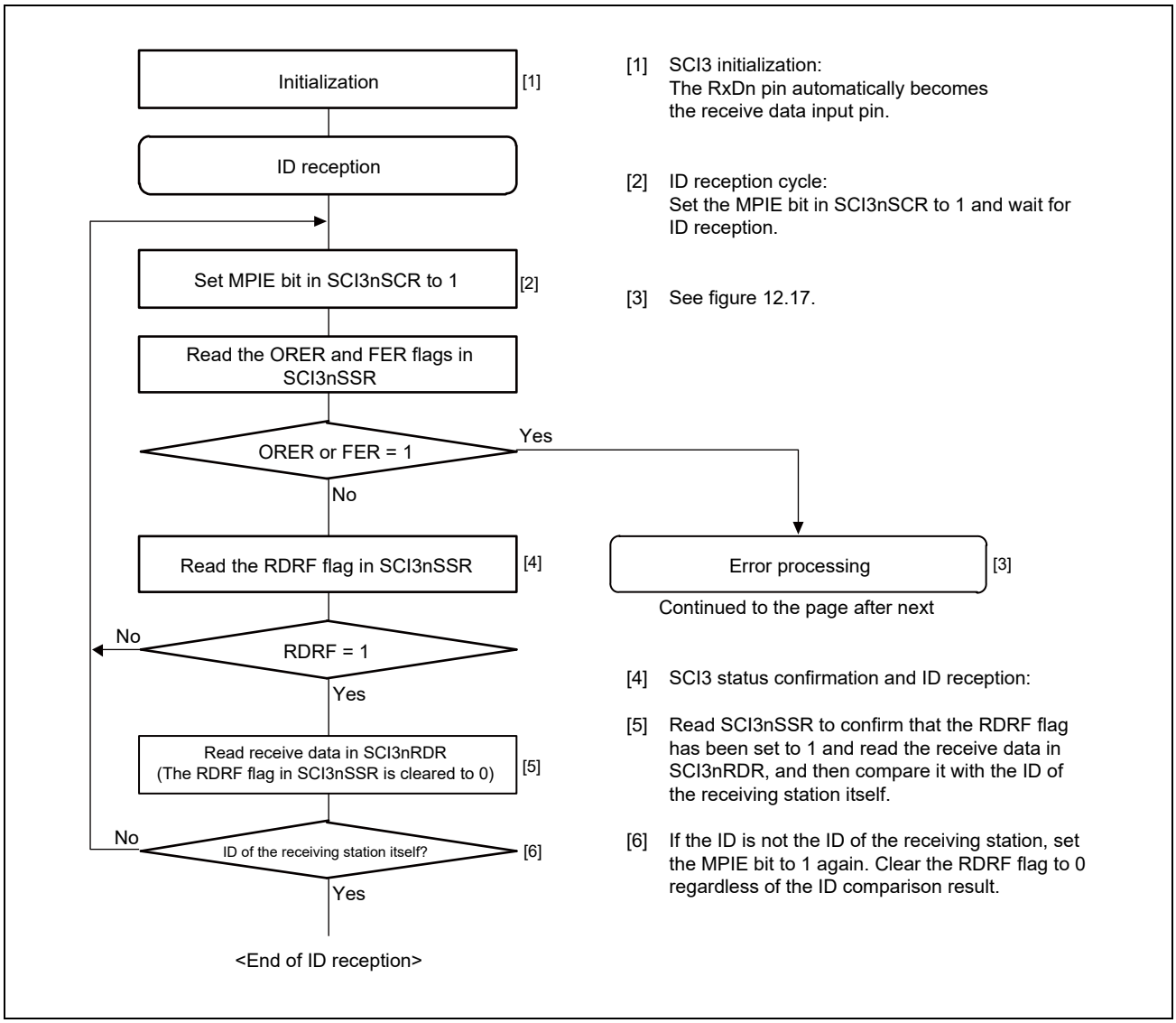


Figure 12.15 Example of Multi-Processor Serial Reception Flowchart (1)

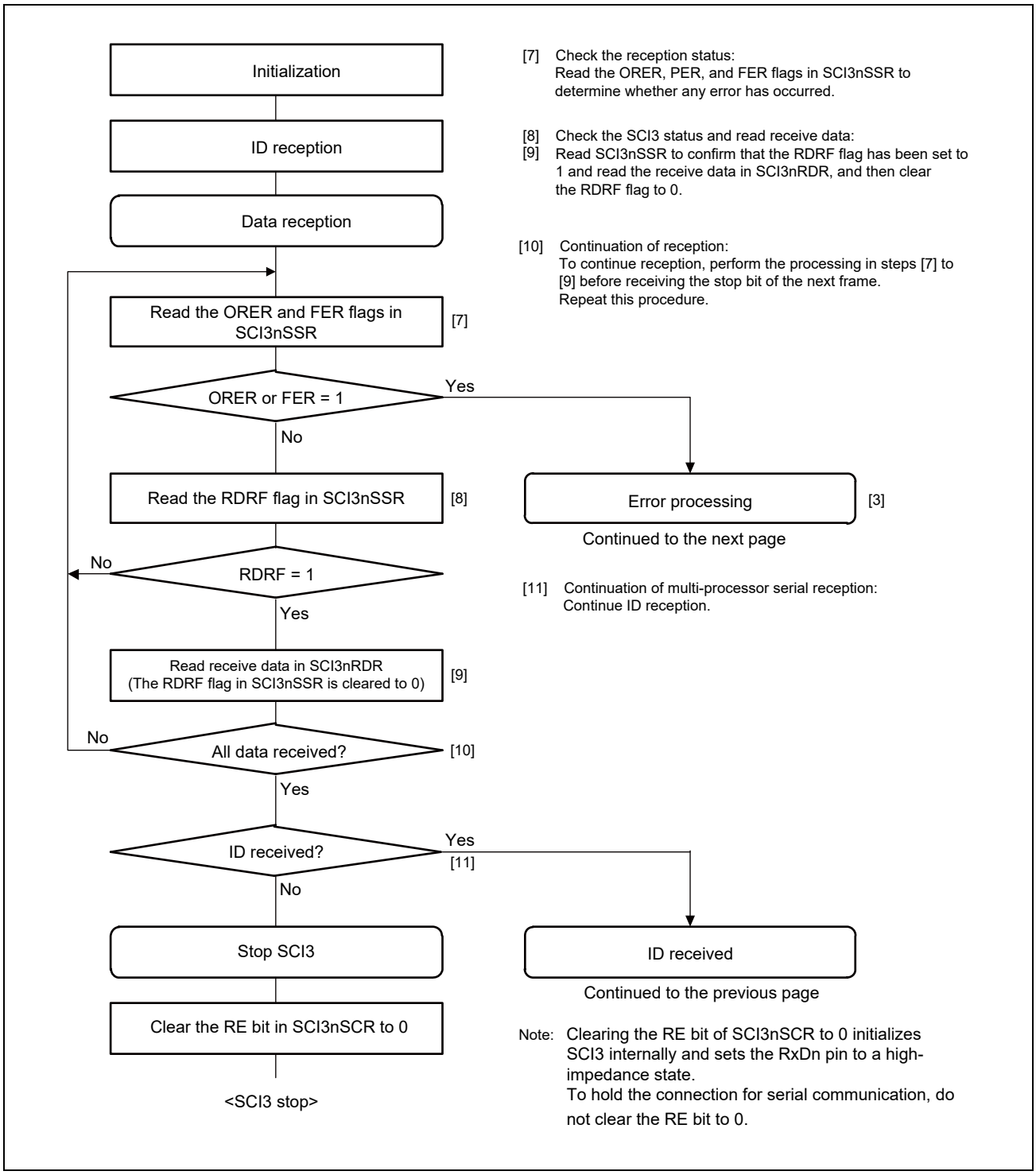


Figure 12.16 Example of Multi-Processor Serial Reception Flowchart (2)

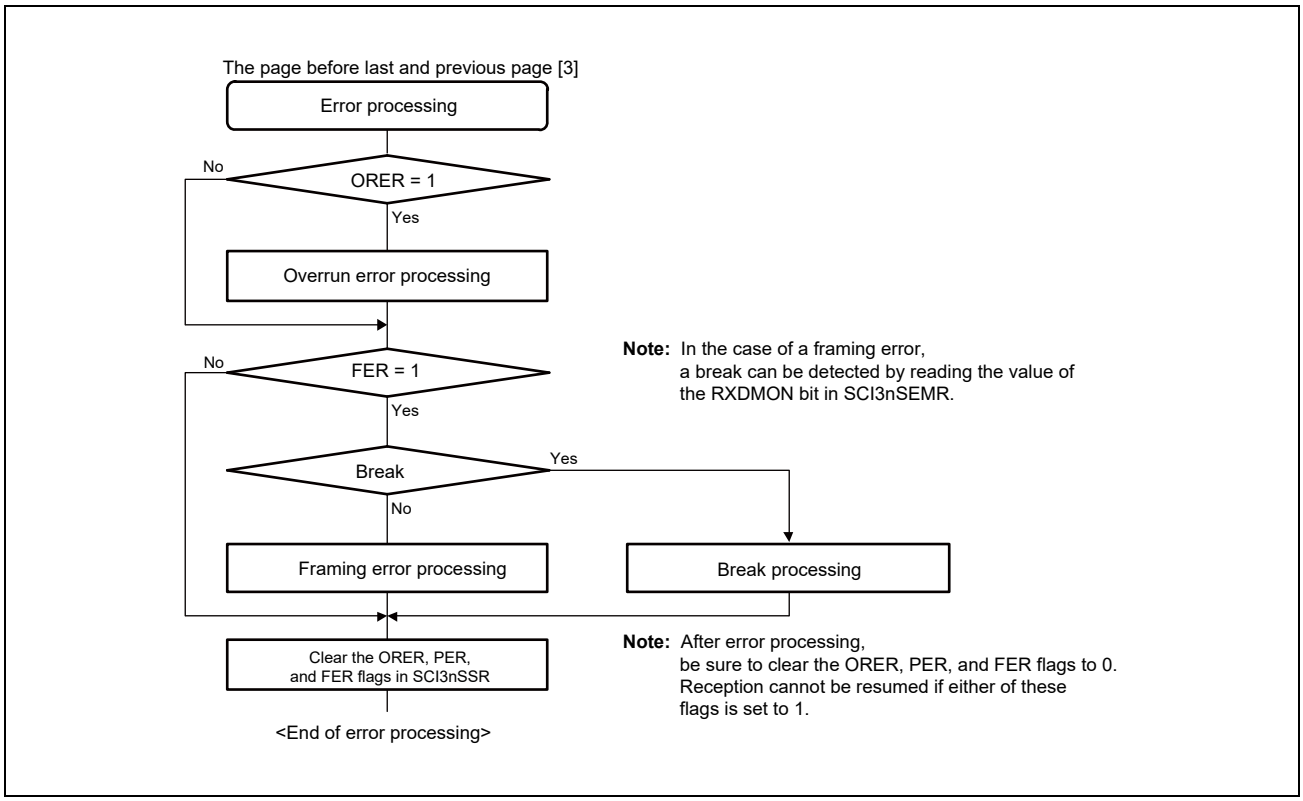
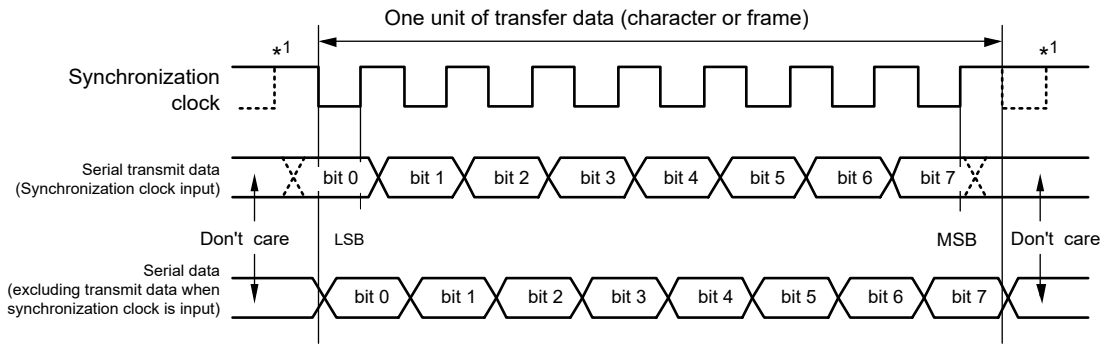


Figure 12.17 Example of Multi-Processor Serial Reception Flowchart (3)

12.4.3 Operation in Clock Synchronous Mode

Figure 12.18 shows the data format for clock synchronous serial data communication. In clock synchronous mode, data is transmitted or received in synchronization with clock pulses. One character in transfer data consists of 8-bit data. In data transmission with the synchronization clock output, the SCI3 outputs data from one falling edge to the next falling edge of the synchronization clock. In data transmission with the synchronization clock input, the SCI3 outputs the first data (bit 0) after starting transfer immediately after clearing the TDRE bit in SCI3nSSR to 0, and then outputs the next bit data after 2 to 3 PCLK clock cycles from the falling edge of the synchronization clock. In data reception, the SCI3 receives data in synchronization with the rising edge of the synchronization clock. After 8-bit data is output, the communication line holds the last-bit output state. In clock synchronous mode, neither parity bit nor multi-processor bit can be added. The transmitter and the receiver are independent in the SCI3, enabling full-duplex communication by using a common clock. Both the transmitter and the receiver have a double-buffered structure so that the next transmit data can be written during transmission or the previous receive data can be read during reception, enabling continuous data transfer.



Note 1. Holds a high level except during continuous transmission/reception.

Figure 12.18 Data Format in Clock Synchronous Mode (LSB-First)

12.4.3.1 Clock

An internal clock generated by the on-chip baud rate generator or an external synchronous clock that is input from the SCKn pin can be selected by the setting of the CKE1 and CKE0 bits in SCI3nSCR. When operating the SCI3 on the internal clock, a synchronous clock is output from the SCKn pin. Eight pulses of the synchronous clock are output during transfer of one character, and the clock is held high while no data is transferred.

12.4.3.2 SCI3 initialization (clock synchronous mode)

Before transmitting and receiving data, clear the TE and RE bits in SCI3nSCR to 0 and then initialize the SCI3 according to the sample flowchart in **Figure 12.19**. To switch the operation between transmission, reception, and transmission/reception, clear the TE and RE bits to 0 and then set these bits to the desired value. Before changing the transfer format, be sure to clear the TE and RE bits to 0. Note that clearing the TE bit to 0 sets the TDRE flag to 1, but clearing the RE bit to 0 initializes neither the RDRF, PER, FER, and ORER flags nor SCI3nRDR.

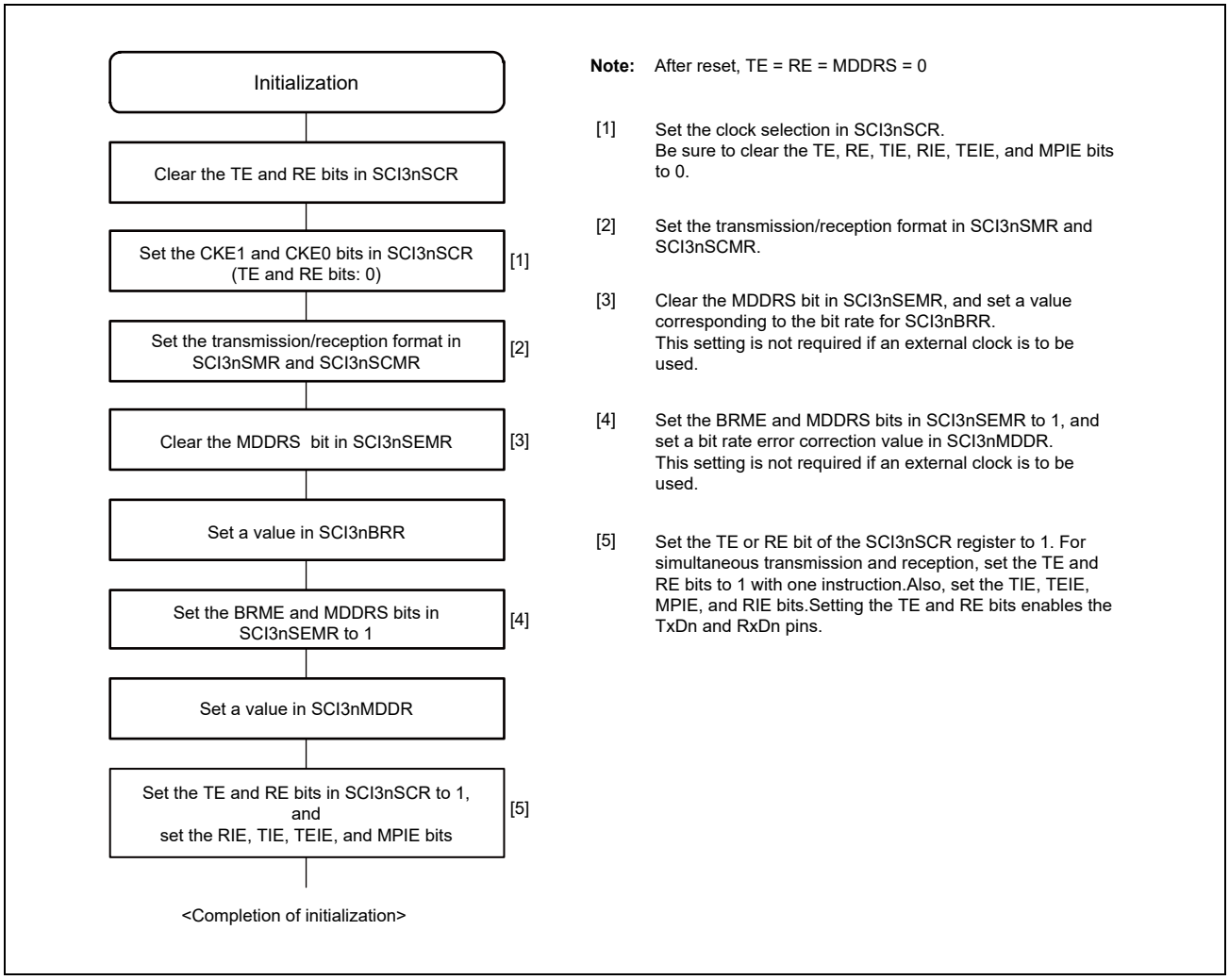


Figure 12.19 Example of SCI3 Initialization Flowchart

12.4.3.3 Serial data transmission (clock synchronous mode)

Figure 12.20 shows an example of the operation for serial transmission in clock synchronous mode. In serial data transmission, the SCI3 operates as described below.

1. When transmit data is written to SCI3nTDR, the TDRE flag is automatically cleared to 0. The SCI3 monitors the TDRE flag in SCI3nSSR. When the flag is cleared, the SCI3 recognizes that data has been written to SCI3nTDR and transfers data from SCI3nTDR to SCI3nTSR, starting output of the first bit when the synchronization clock is input. To write transmit data to SCI3nTDR at a trigger of TXI interrupt request, set the TIE bit to 1 and then set the TE bit to 1 or set both TIE and TE bits simultaneously with one instruction to generate a TXI interrupt request for starting data transfer.
2. Transmission starts after data is transferred from SCI3nTDR to SCI3nTSR, and the TDRE flag is set to 1. When the TIE bit in SCI3nSCR is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to SCI3nTDR in this TXI interrupt processing routine before the previously transferred data has been transmitted. When a TEI interrupt request is used, the TIE bit is cleared to 0 after the last transmit data has been written to SCI3nTDR, and the TEIE bit is set to 1.
3. 8-bit data is output from the TxDn pin in synchronization with the output clock (when clock output mode has been specified) or in synchronization with the input clock (when external clock has been specified).
4. The SCI3 checks for the TDRE flag at the time of the last bit output.
5. When the TDRE flag is 0, the next transmit data is transferred from SCI3nTDR to SCI3nTSR, and serial transmission of the next frame starts.
6. When the TDRE flag is 1, the TEND flag in SCI3nSSR is set to 1 and the TxDn pin retains the output state of the last bit. If the TEIE bit in SCI3nSCR is set to 1 at this time, a TEI interrupt request is generated. The SCKn pin is held high.

Figure 12.21 shows a sample flowchart of serial data transmission. Also, **Figure 12.22** shows a sample flowchart for stopping the SCI3 after data transmission. Transmission will not start even if the TDRE flag is cleared while a receive error flag (ORER) is set to 1. Be sure to clear the receive error flags to 0 before starting transmission. Note that clearing the RE bit to 0 does not clear the receive error flags.

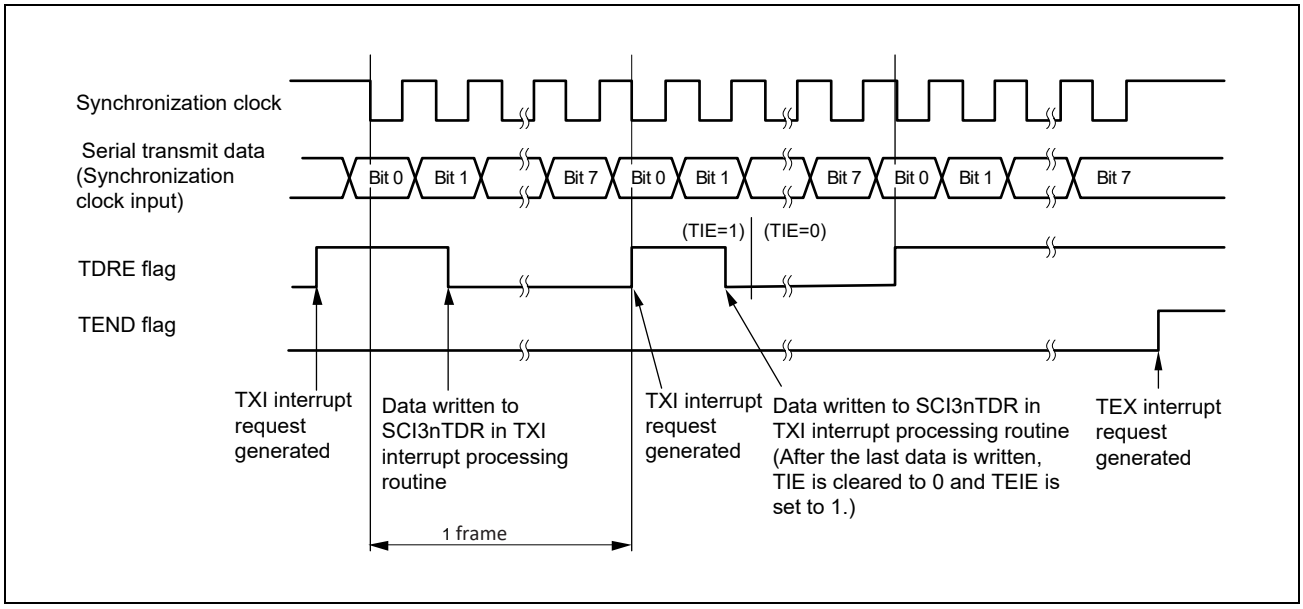


Figure 12.20 Example of Operation for Transmission in Clock Synchronous Mode

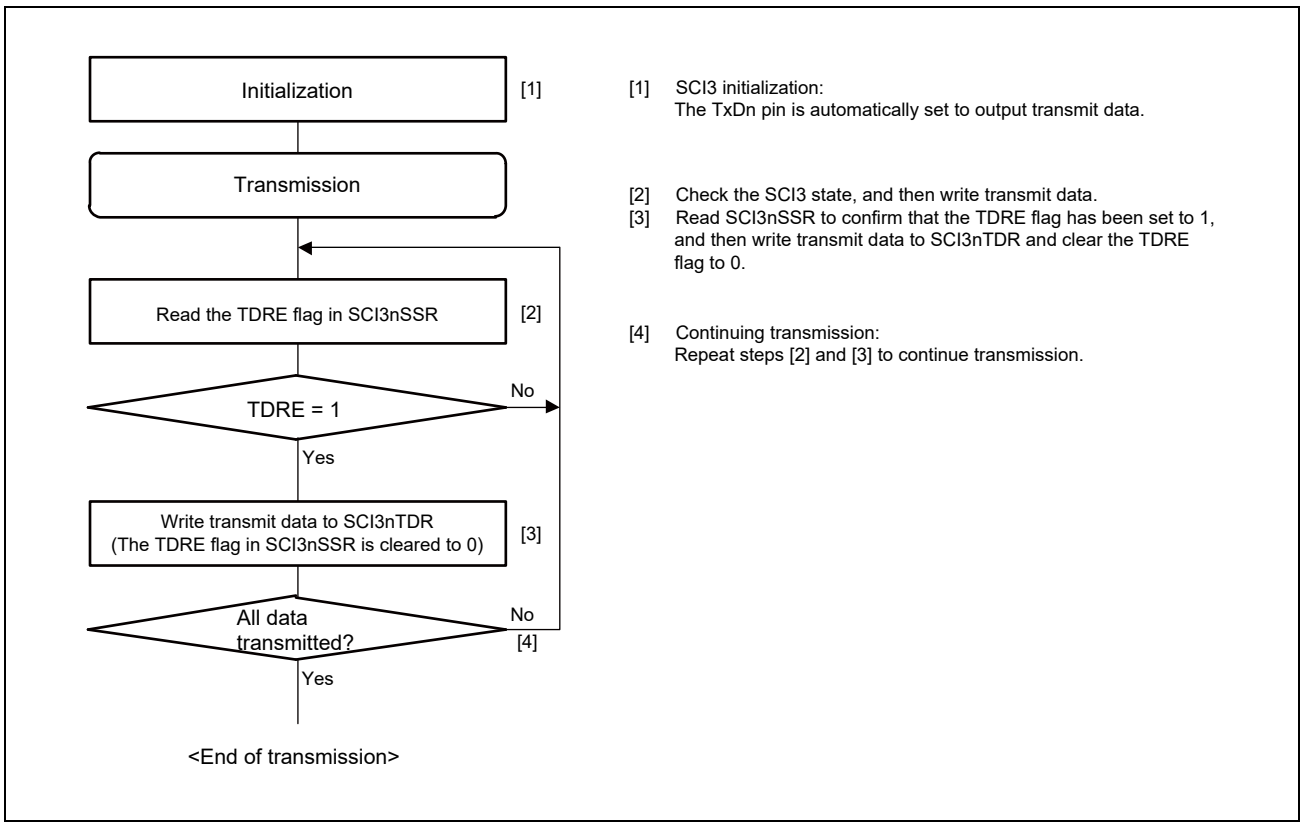


Figure 12.21 Example of Serial Transmission Flowchart

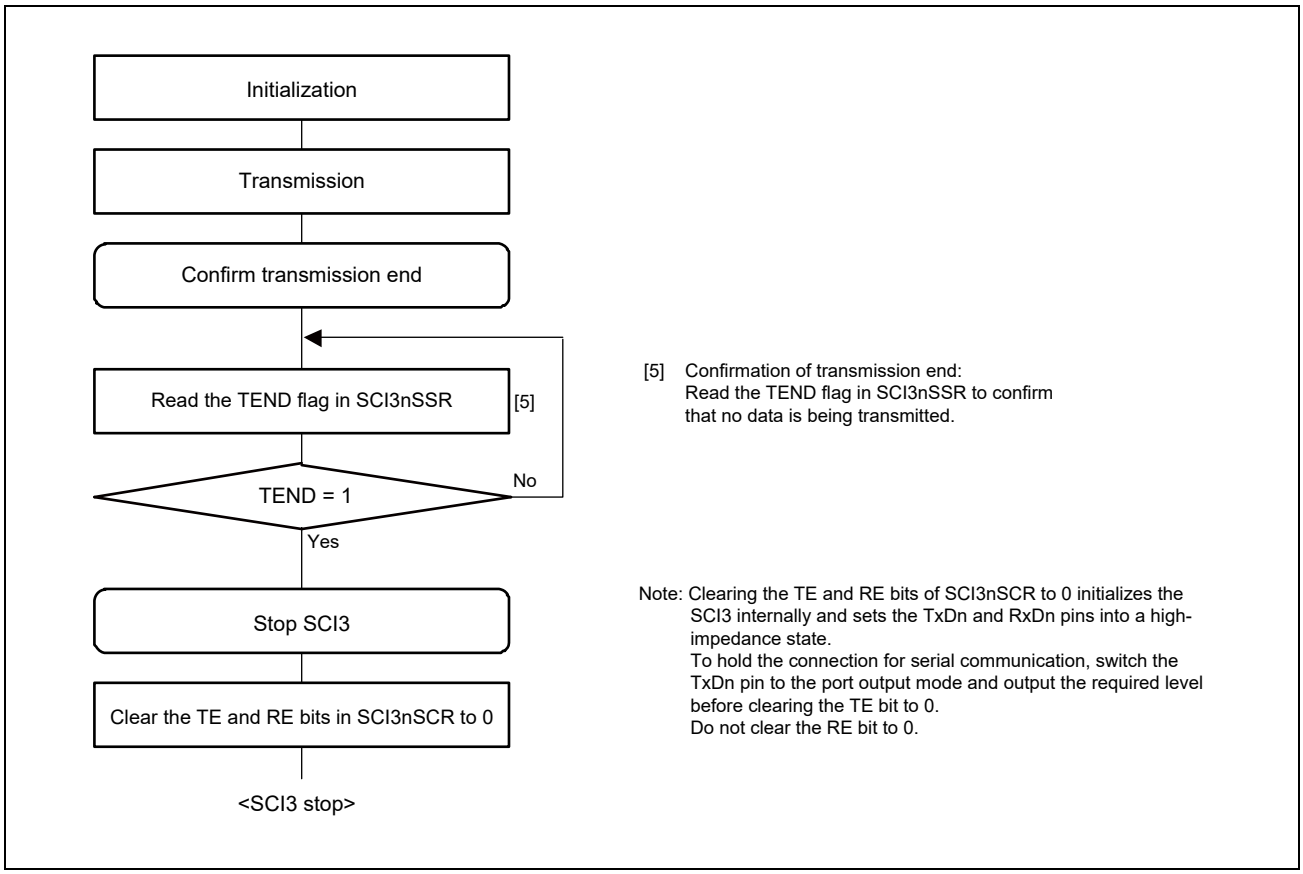


Figure 12.22 Example Flowchart for Stopping the SCI3 after Serial Transmission

12.4.3.4 Serial data reception (clock synchronous mode)

Figure 12.23 shows an example of SCI3 operation for serial reception in clock synchronous mode. In serial data reception, the SCI3 operates as described below.

1. The SCI3 performs internal initialization and starts receiving data in synchronization with a synchronization clock input or output, and stores receive data in SCI3nRSR.
2. When an overrun error occurs (the reception of the next data is completed with the RDRF flag in SCI3nSSR set to 1), the ORER flag in SCI3nSSR is set to 1. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to SCI3nRDR. The RDRF flag retains the state of being set to 1.
3. When data has been successfully received, the RDRF flag in SCI3nSSR is set to 1 and the receive data is transferred to SCI3nRDR. When the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to SCI3nRDR in this RXI interrupt processing routine before reception of the next data is completed. Reading SCI3nRDR automatically clears the RDRF flag to 0.

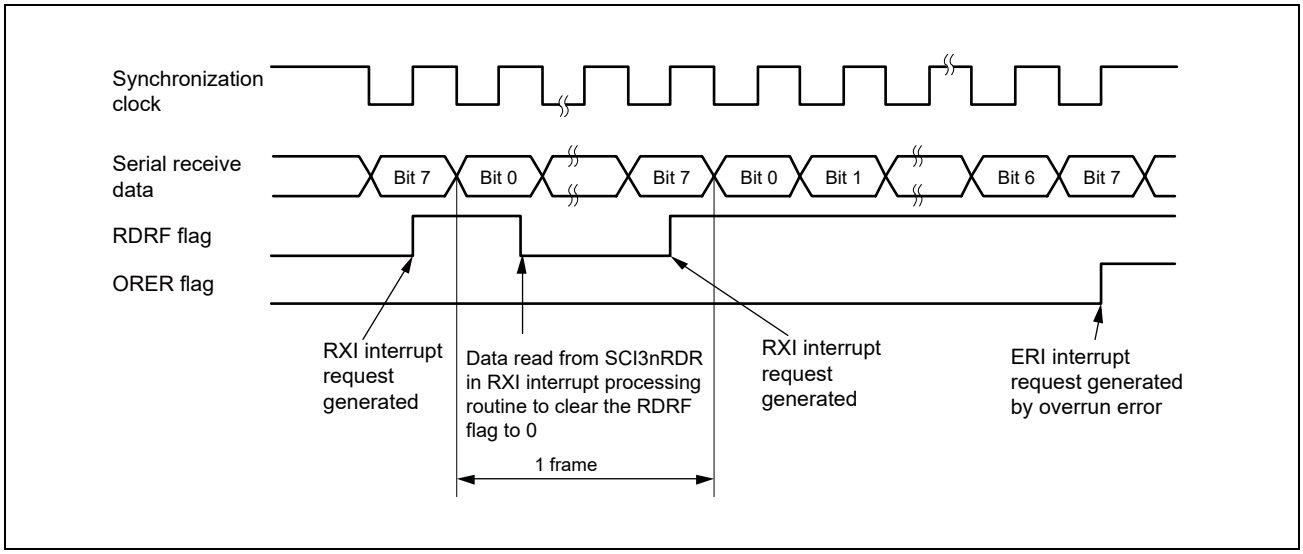


Figure 12.23 Example of SCI3 Reception

Subsequent transmission and reception are disabled with a receive error flag set to 1. Therefore, be sure to clear the ORER, FER, PER, and RDRF flags before continuing reception. **Figure 12.24** shows an example of flowchart for data reception.

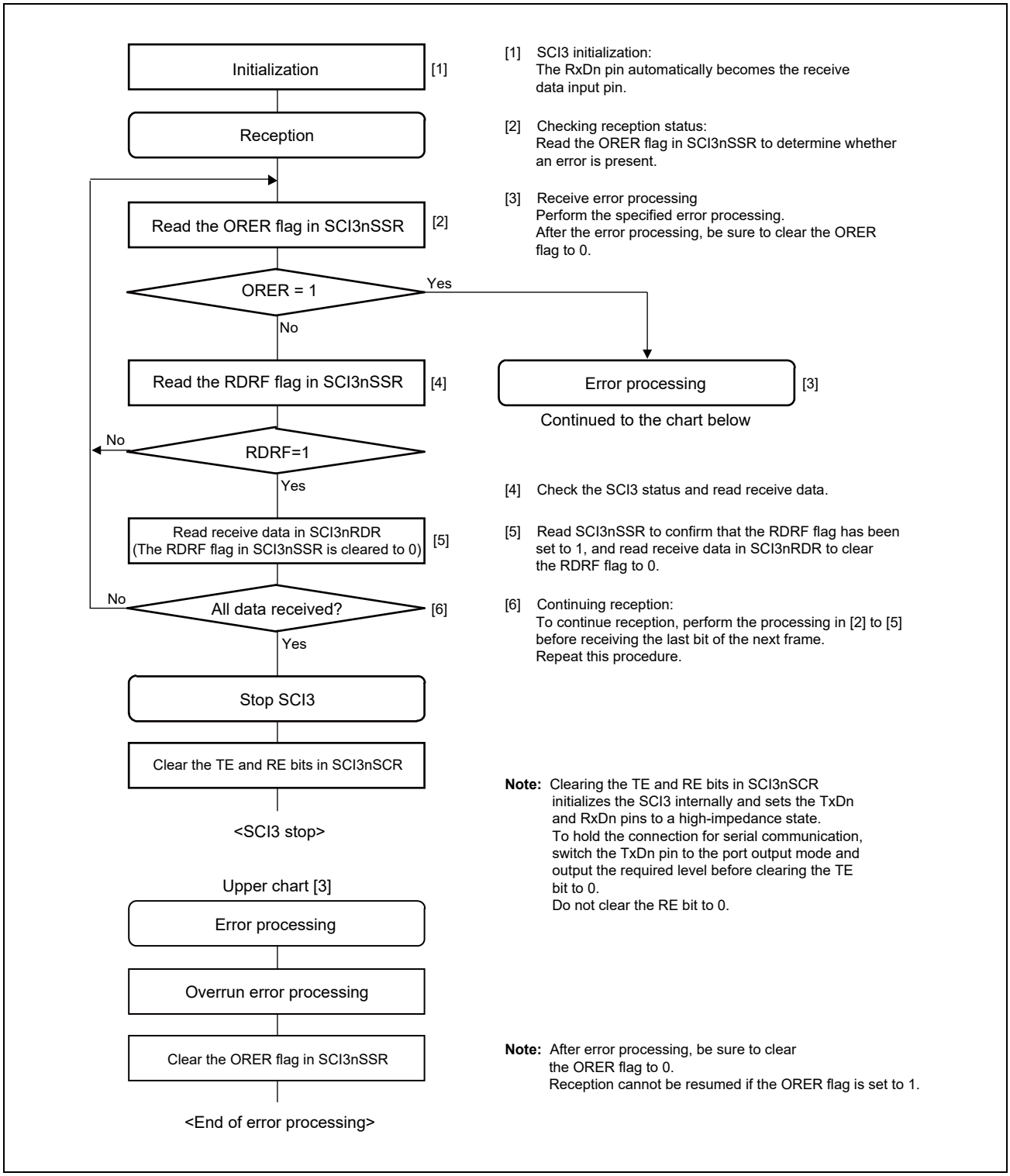


Figure 12.24 Example of Serial Reception Flowchart

12.4.3.5 Simultaneous serial data transmission and reception (clock synchronous mode)

Figure 12.25 shows a sample flowchart for simultaneous data transmit and receive operations. After the SCI3 is initialized, perform the following procedure for simultaneous data transmit and receive operations.

1. To change transmit mode to simultaneous transmit and receive mode, check that the SCI3 has finished transmission and the TDRE and TEND flags are set to 1. Then clear the TE bit to 0 and then set the TE and RE bits to 1 with a single instruction.
2. To change receive mode to simultaneous transmit and receive mode, check that the SCI3 has finished reception and clear the RE bit to 0. Then check that the RDRF and error flags (ORER, FER, and PER) are cleared to 0 and then set the TE and RE bits to 1 with a single instruction.

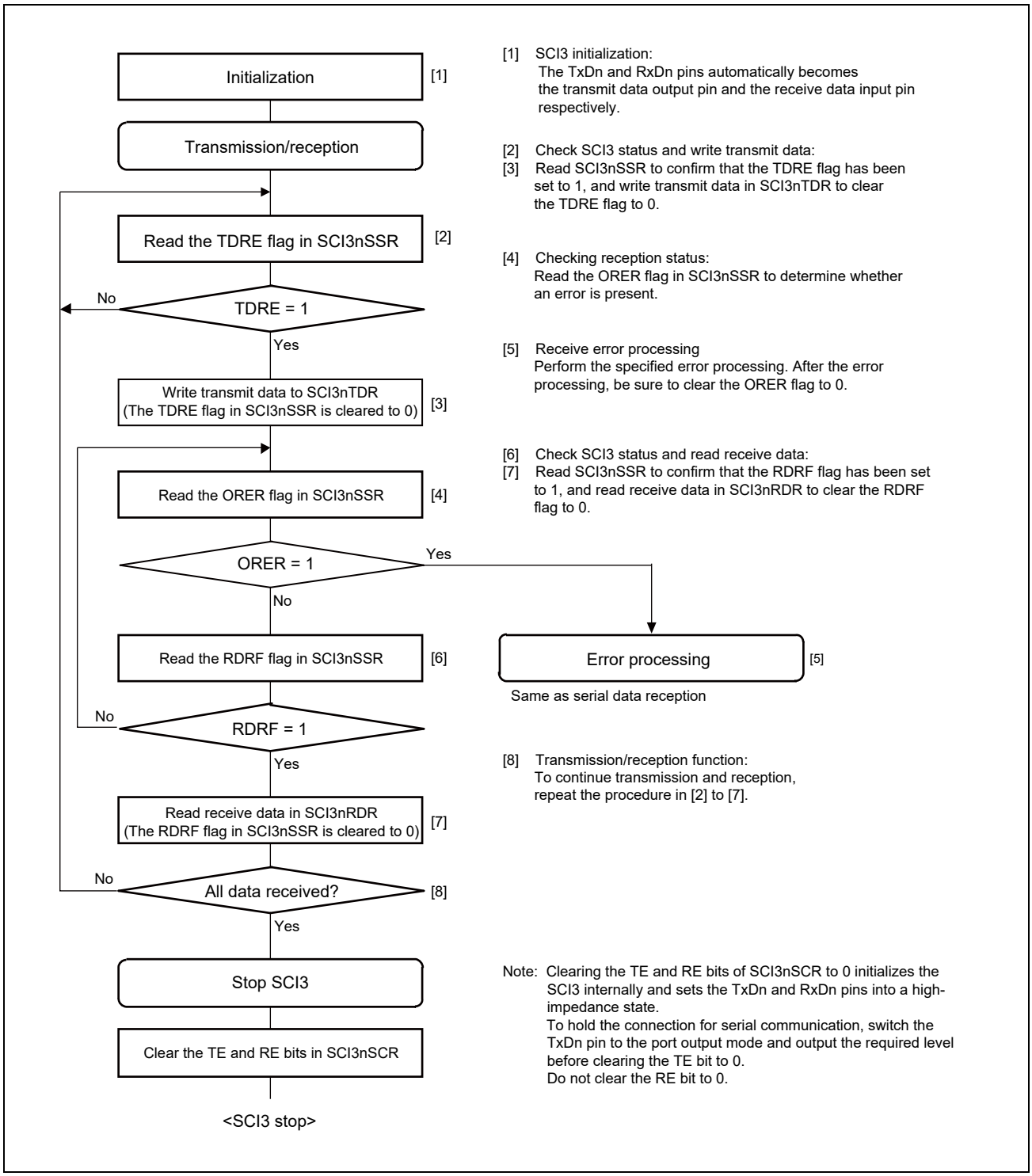


Figure 12.25 Example of Simultaneous Serial Transmission and Reception Flowchart

12.4.4 Bit Rate Modulation Function

The bit rate modulation function corrects a bit rate by enabling the internal clocks for the number specified by SCI3nMDDR, in a manner that they are enabled approximately even out of the 256 internal clocks specified by the CKS1 and CKS0 bits in SCI3nSMR.

Figure 12.26 shows an example of asynchronous mode in which PCLK clock is selected with the CKS1 and CKS0 bits, SCI3nBRR is set to 0, and SCI3nMDDR is set to 160. In this example, the reference clock cycle is corrected to 256/160 approximately evenly and the bit rate is corrected to 160/256. Note that the pulse widths of the internal reference clock expand or contract for the amount of the selected internal clocks because there is a deviation in the enabling of the internal clocks.

Do not use this function with the maximum speed settings (CKS1 and CKS0 bits = 0 in SCI3nSMR, CKE1 bit = 0 in SCI3nSCR, and SCI3nBRR = 0) for clock synchronous mode.

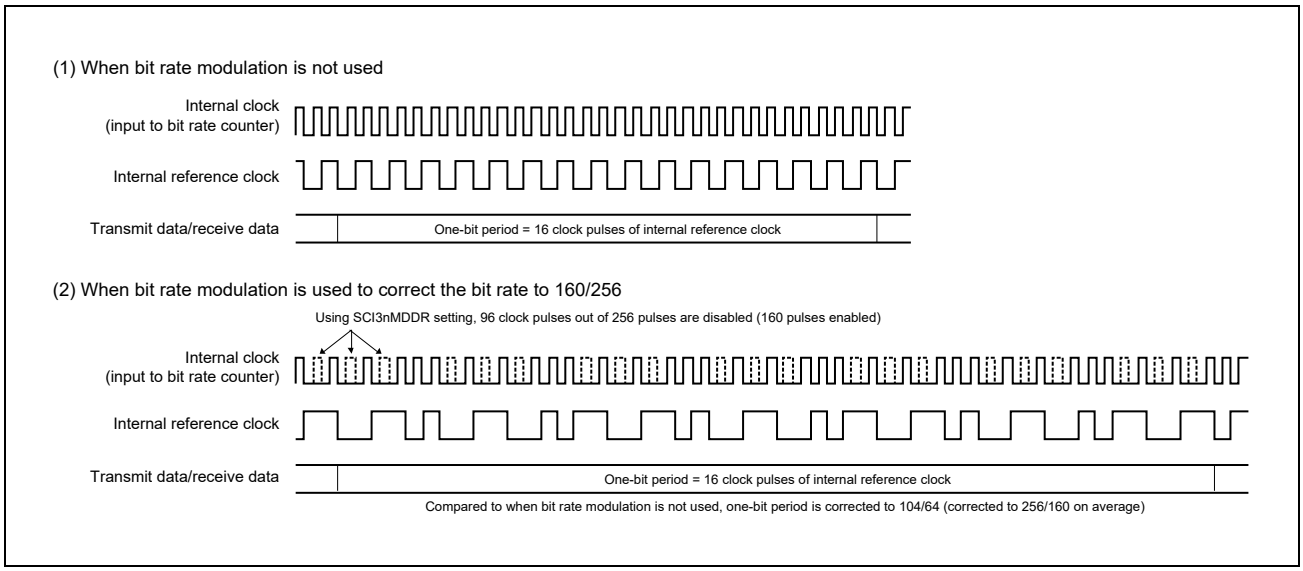


Figure 12.26 Example of Internal Reference Clock when the Bit Rate Modulation Function is Used

12.4.5 Interrupt Sources

Table 12.27 lists interrupt sources. Each interrupt source outputs an individual interrupt request signal. These interrupt sources can be enabled independently with the enable bit in SCI3nSCR.

A TXI interrupt request is generated when the TDRE flag in SCI3nSSR is set to 1. A TEI interrupt request is generated when the TEND flag in SCI3nSSR is set to 1. A TXI interrupt request can activate the DMAC to handle data transfer. The TDRE flag is automatically cleared to 0 when data is transferred using the DMAC.

CAUTION

The TDRE and TEND flags cannot be cleared to 0 while the TE bit in SCI3nSCR is 0. Since the TEND flag is the level interrupt request flag for a TEI interrupt, do not set the TEIE bit in SCI3nSCR to 1 while the TE bit is 0.

An RXI interrupt request is generated when the RDRF flag in SCI3nSSR is set to 1. An ERI interrupt is generated when either of the ORER, PER, and FER flags in SCI3nSSR is set to 1. An RXI interrupt request can activate the DMAC to handle data transfer. The RDRF flag is automatically cleared to 0 when data is transferred using the DMAC.

A TEI interrupt request is generated when the TEND flag is set to 1 with the TEIE bit set to 1.

CAUTION

If a TEI interrupt request and a TXI interrupt request are generated at the same time, the TXI interrupt request is accepted first. Note that clearing the TDRE flag to 0 at this time in the TXI interrupt processing routine also clears the TEND flag to 0 automatically, disabling the branch to the TEI interrupt processing routine.

Table 12.27 SCI3 Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC/DTS Activation
ERI	Receive error	ORER, FER, PER	Not possible
RXI	Receive data full	RDRF	Possible
TXI	Transmit data empty	TDRE	Possible
TEI	Transmit end	TEND	Not possible

12.5 Notes

12.5.1 Break Detection and Processing

A break can be detected by reading the RXDMON bit in SCI3nSEMR at the time of detection of a framing error. Since all inputs from the RxDn pin are 0 in the break state, the FER flag is set to 1 and the PER flag may also be set to 1. When the ASTLS bit in SCI3nSCMR is 1, the SCI3 continues data reception even after it receives a break. For this reason, note that the FER flag is set to 1 again even after having been cleared to 0.
After the end of the break state, retain the input from the RxDn pin at the high level over one frame to avoid a disparity in bit timing between the transmission and reception sides. Meanwhile, when the ASTLS bit in SCI3nSCMR is 0, the SCI3 stops reception in the state of waiting to detect the start bit of the next frame after the FER flag has been set to 1. If the FER flag is cleared to 0 at this time, the FER flag retains the value 0 in the break state. If the input from the RxDn pin becomes 1 and the break state ends, the beginning of the start bit is detected on a falling edges of the input on the RxDn pin and reception starts.

12.5.2 Mark State and Break Output

While the TE bit is 0 (transmission/reception disabled), the TxDn pin can output any level by switching the TxDn pin to a general output port. This allows the TxDn pin to be the mark state or break output during data transmission.

12.5.3 Receive Error Flags and Transmit Operations in Clock Synchronous Mode

During the clock synchronous simultaneous data transmit/receive operation, transmission cannot be started when a receive error flag (ORER) is set to 1, even if the TDRE flag is cleared to 0. Be sure to clear the receive error flags to 0 before starting transmission. Note that the receive error flags cannot be cleared to 0 even by clearing the RE bit to 0.

12.5.4 Relationship between Writing to SCI3nTDR and the TDRE Flag

The TDRE flag in SCI3nSSR is a status flag indicating that transmit data in SCI3nTDR has been transferred to SCI3nTSR. The TDRE flag is set to 1 when the SCI3 transfers data from SCI3nTDR to SCI3nTSR.

Data can be written to SCI3nTDR regardless of the status of the TDRE flag. However, writing new data to SCI3nTDR while the TDRE flag is 0 will make the data stored in SCI3nTDR lost because it has not been transferred to SCI3nTSR. Therefore, make sure to check that the TDRE flag is set to 1 before writing transmit data to SCI3nTDR.

12.5.5 Restrictions on Using an External Clock for Transmission in Clock Synchronous Mode

When using an external synchronization clock, clear the TDRE flag to 0 and then input a transmission clock (**Figure 12.27**). Also in continuous transmission mode, clear the TDRE flag to 0 and then input a transmission clock for the next frame.

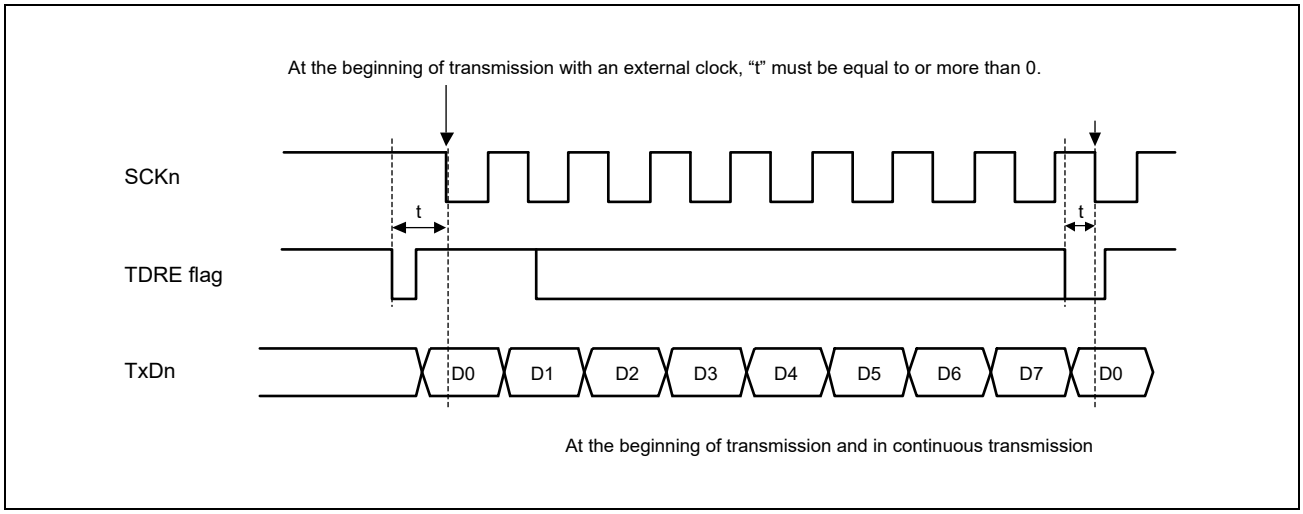


Figure 12.27 Restrictions on Using an External Clock for Transmission in Clock Synchronous Mode

12.5.6 External Clock Input in Clock Synchronous Mode

For input of the external clock SCInSCK in clock synchronous mode, see **Section 39.3.6, SCI/FLSCI Timing**.

Section 13 LIN/UART Interface (RLIN3)

This section contains a generic description of the LIN/UART interface (RLIN3).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of RLIN3.

13.1 Features of RH850/C1M-A RLIN3

13.1.1 Number of Units and Channels

This microcontroller has the following number of RLIN3 units.

RLIN3 unit has one channel interface. “Number of channels” is used with the same meaning as “number of units” in this section

Table 13.1 Number of Units

Product Name	RH850/C1M-A2	RH850/C1M-A1
Number of units	3	3
Name	RLIN3n (n = 0 to 2)	RLIN3n (n = 0 to 2)

Table 13.2 Unit Configurations and Channels

Unit Name (Channel Name) RLIN3n	Channels per Unit	RH850/C1M-A2	RH850/C1M-A1
RLIN30	1	√	√
RLIN31	1	√	√
RLIN32	1	√	√

Note: The channel names are same as those of the corresponding units.

Table 13.3 Indices

Index	Description
n	Throughout this section, the individual RLIN3 units are identified by the index “n” (n = 0 to 2): for example, RLIN3nLCUC is the LIN control register.
b	Throughout this section, the individual transmit/receive data buffers of RLIN3n are identified by the index “b” (b = 1 to 8): for example, RLN3nLDBRb is the data buffer register.

13.1.2 Register Base Address

RLIN3 base addresses are listed in the following table.

RLIN3 register addresses are given as offsets from the base addresses in general.

Table 13.4 Register Base Addresses

Base Address Name	Base Address
<RLIN30_base>	FFCE 0000$_H$
<RLIN31_base>	FF6E 1000$_H$
<RLIN32_base>	FFCE 2000$_H$

13.1.3 Clock Supply

The RLIN3 clock supply is shown in the following table.

Table 13.5 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
RLIN3n	clkc*1	1/2 x CLKC_LSB (unmodulated low-speed peripheral clock divided by 2)
	pclk	CLK_LSB (low-speed peripheral clock)

Note 1. LIN communication clock source

13.1.4 Interrupt Request

RLIN3 interrupt requests are listed in the following table.

Table 13.6 Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
RLIN30						
INTRLIN3n (n = 0)	RLIN30 interrupt	277	—	—	—	—
INTRLIN3nUR0 (n = 0)	RLIN30 transmit interrupt	278	—	121	—	121
INTRLIN3nUR1 (n = 0)	RLIN30 receive completion interrupt	279	—	120	—	120
INTRLIN3nUR2 (n = 0)	RLIN30 status interrupt	280	—	—	—	—
RLIN31						
INTRLIN3n (n = 1)	RLIN31 interrupt	281	—	—	—	—
INTRLIN3nUR0 (n = 1)	RLIN31 transmit interrupt	282	—	123	—	123
INTRLIN3nUR1 (n = 1)	RLIN31 receive completion interrupt	283	—	122	—	122
INTRLIN3nUR2 (n = 1)	RLIN31 status interrupt	284	—	—	—	—
RLIN32						
INTRLIN3n (n = 2)	RLIN32 interrupt	285	—	—	—	—
INTRLIN3nUR0 (n = 2)	RLIN32 transmit interrupt	286	—	125	—	125
INTRLIN3nUR1 (n = 2)	RLIN32 receive completion interrupt	287	—	124	—	124
INTRLIN3nUR2 (n = 2)	RLIN32 status interrupt	288	—	—	—	—

—: Not assigned.

Note 1. 1st: Primary channel , 2nd: Secondary channel

13.1.5 Reset Sources

RLIN3 reset sources are listed in the following table. RLIN3 is initialized by these reset sources.

Table 13.7 Reset Sources

Unit Name	Reset Source
RLIN3n	All reset sources

13.1.6 External Input/Output Signals

External input/output signals of RLIN3 are listed below.

Table 13.8 I/O signals of RLIN3

Unit Signal Name	Outline	Alternative Port Pin Signal
RLIN30		
RLIN3nRX (n = 0)	RLIN30 receive data input	RLIN30RX
RLIN3nTX (n = 0)	RLIN30 transmit data output	RLIN30TX
RLIN31		
RLIN3nRX (n = 1)	RLIN31 receive data input	RLIN31RX
RLIN3nTX (n = 1)	RLIN31 transmit data output	RLIN31TX
RLIN32		
RLIN3nRX (n = 2)	RLIN32 receive data input	RLIN32RX
RLIN3nTX (n = 2)	RLIN32 transmit data output	RLIN32TX

13.2 Overview

13.2.1 Functional Overview

The LIN/UART interface is a hardware LIN communication controller that supports LIN Specification Package Revision 1.3, 2.0, 2.1, 2.2, and SAE J2602, and automatically performs frame communication and error determination.

The LIN/UART interface is provided with UART mode and can also be used as a UART.

The appropriate mode should be used for the LIN/UART interface according to the application: LIN master, LIN slave, or UART.

LIN master

- LIN reset mode
- LIN mode (LIN master mode)
  - LIN wake-up mode
  - LIN operation mode
- LIN self-test mode

LIN slave

- LIN reset mode
- LIN mode (LIN slave mode [auto baud rate] or LIN slave mode [fixed baud rate])
  - LIN wake-up mode
  - LIN operation mode
- LIN self-test mode

UART

- LIN reset mode
- UART mode

Table 13.9 gives the LIN/UART interface specifications.

Table 13.9 LIN/UART Interface Specifications (1/3)

Item			Specifications
	Channel count		3 channels
LIN communication function	Protocol		LIN Specification Package Revision 1.3, 2.0, 2.1, 2.2, SAE J2602
	Variable frame structure	Master	• Break transmission width: 13 to 28 Tbit • Break delimiter transmission width: 1 to 4 Tbit • Transmission inter-byte space width (header): 0 to 7 Tbits (space between Sync field and ID field)*1 • Transmission response space width: 0 to 7 Tbit*1 • Transmission inter-byte space width: 0 to 3 Tbits (space between data bytes in response area) • Transmission wake-up width: 1 to 16 Tbits
		Slave	• Break reception width : 9.5 or 10.5 Tbits [for fixed baud rate] : 10 or 11 Tbits [for auto baud rate] • Transmission response space width: 0 to 7 Tbits • Transmission inter-byte space width: 0 to 3 Tbits (space between data bytes in response area) • Transmission wake-up width: 1 to 16 Tbits
	Checksum		• Automatic operation for both transmission and reception • Classic or enhanced selectable (for each frame)
	Response field data byte count		Variable from 0 to 8 bytes Multi-byte (9 or more bytes) response transmission and reception also possible
	Frame communication modes	Master	• Mode in which header transmission and response transmission/reception is started with a single transmission start request • Mode in which header transmission and response transmission are started with separate transmission start requests (frame separate mode)
		Slave	• Mode in which header is automatically received with fixed baud rate • Mode in which header is automatically received with the baud rate set according to the sync field measurement result of the sync field and break field detected
	Wake-up transmission and reception		LIN wake-up mode provided • Wake-up transmission (1 to 16 Tbits) • Wake-up reception Low-level width of input signals measured
	Status	Master	• Successful frame/wake-up transmission • Successful header transmission • Successful frame/wake-up reception* *2 • Successful data 1 reception • Error detection • Operation mode (LIN reset mode, LIN wake-up mode, LIN operation mode, LIN self-test mode)
		Slave	• Successful response/wake-up transmission • Successful response/wake-up reception*2 • Successful header reception • Successful data 1 reception • Error detection • Operation mode (LIN reset mode, LIN wake-up mode, LIN operation mode, LIN self-test mode)

Table 13.9 LIN/UART Interface Specifications (2/3)

Item		Specifications	
LIN communication function	Error status	Master	• Bit error • Checksum error • Frame timeout error/response timeout error • Physical bus error • Framing error • Response preparation error
		Slave	• Bit error • Checksum error • Frame timeout error/response timeout error • Sync field error • ID parity error • Framing error • Response preparation error
	Baud rate selection	Baud rate conforming to the LIN specifications generated using baud rate generator	
	Test mode	Self-test mode for user evaluation	
	Interrupt function	Master	• Successful header/frame/wake-up transmission • Successful frame/wake-up reception*2*[2] • Error detection
		Slave	• Successful response/wake-up transmission • Header/response/wake-up reception*2*[2] • Error detection
UART communication function	Data buffer	• Transmission data buffer/transmission data buffer for wait • UART buffer (exclusively for transmission; variable data length from 1 to 9 bits; character length of 7 and 8 bits supported) • Reception data buffer (exclusively for reception; data length of 1. Character length of 7, 8, and 9 bits supported)	
	Data format	Character length: 7 or 8 bits 9 bits including the expansion bit supported.	
		Transmission stop bit: 1 or 2 bits	
		Parity function: odd, even, 0, or none	
		LSB- or MSB-first transfer selectable	
		Reverse input/output of transmission/reception data	
	Status	• Transmission status • Reception status • Successful UART buffer transmission • Error detection • Expansion bit detection • ID match • Reset mode status	

Table 13.9 LIN/UART Interface Specifications (3/3)

Item		Specifications
UART communication function	Error status	• Bit error • Framing error • Parity error • Overrun error
	Baud rate selection	With the baud rate generator incorporated, any baud rate can be set.
	When a certain expansion bit is at the expected level, the data received can be compared to the 8-bit data preset in the register.	
	The stop bit received is guaranteed (start of transmission can be delayed when start of transmission is attempted during reception of the stop bit).	
	Interrupt function	• Transmission start/complete • Reception complete • Status/error detection

Note 1. Since the same register is used for setting, the inter-byte space (header) = response space.

Note 2. For wake-up reception, the low level width of the input signal is indicated.

13.2.2 Block Diagram

Figure 13.1 shows a block diagram of the LIN/UART interface.

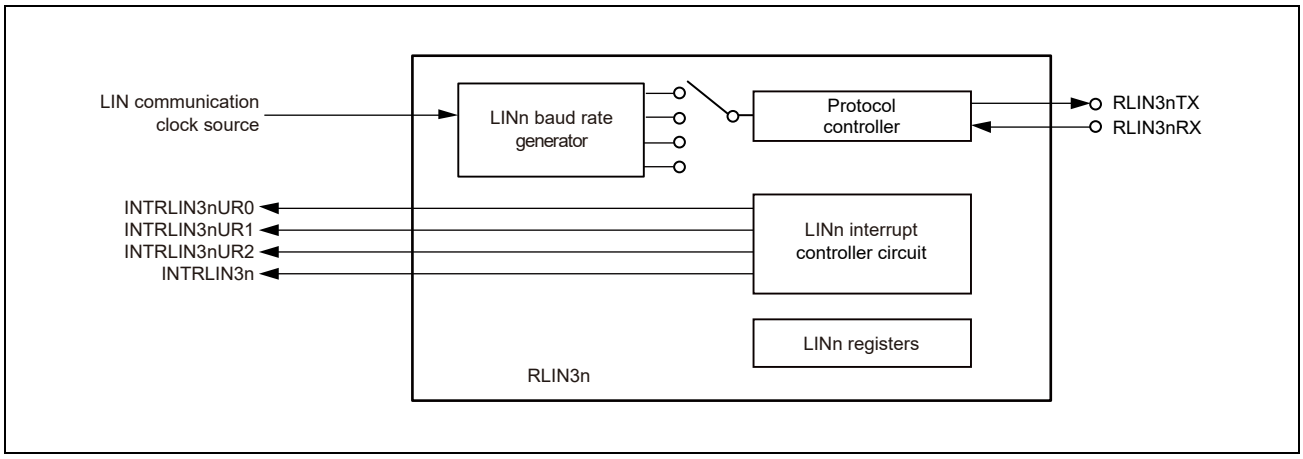


Figure 13.1 LIN/UART Interface Block Diagram

13.2.3 Description of Blocks

- RLIN3nTX, RLIN3nRX: LIN/UART interface I/O pins
- LINn baud rate generator: Generates the LIN/UART interface communication clock signal.
- LINn registers: LIN/UART interface registers
- LINn interrupt controller circuit: Controls interrupt requests generated by the LIN/UART interface

13.3 Registers

13.3.1 List of Registers

RLIN3 registers are listed in the following table.

For details about <RLIN3n_base>, see **Section 13.1.2, Register Base Address**.

Table 13.10 Registers

Module	Register	Symbol	Address	LIN Master	LIN Slave	UART
RLN3n	LIN wake-up baud rate selector register	RLN3nLWBR	<RLIN3n_base> + 01_H	√	√	√
RLN3n	LIN/UART baud rate prescaler 01 register	RLN3nLBRP01	<RLIN3n_base> + 02_H	—	√	√
RLN3n	LIN/UART baud rate prescaler 0 register	RLN3nLBRP0	<RLIN3n_base> + 02_H	√	√	√
RLN3n	LIN/UART baud rate prescaler 1 register	RLN3nLBRP1	<RLIN3n_base> + 03_H	√	√	√
RLN3n	LIN self-test control register	RLN3nLSTC	<RLIN3n_base> + 04_H	√	√	—
RLN3n	LIN/UART mode register	RLN3nLMD	<RLIN3n_base> + 08_H	√	√	√
RLN3n	LIN break field configuration register/ UART configuration register	RLN3nLBFC	<RLIN3n_base> + 09_H	√	√	√
RLN3n	LIN/UART space configuration register	RLN3nLSC	<RLIN3n_base> + $0A_H$	√	√	√
RLN3n	LIN wake-up configuration register	RLN3nLWUP	<RLIN3n_base> + $0B_H$	√	√	—
RLN3n	LIN interrupt enable register	RLN3nLIE	<RLIN3n_base> + $0C_H$	√	√	—
RLN3n	LIN / UART error detection enable register	RLN3nLEDE	<RLIN3n_base> + $0D_H$	√	√	√
RLN3n	LIN/UART control register	RLN3nLCUC	<RLIN3n_base> + $0E_H$	√	√	√
RLN3n	LIN/UART transmission control register	RLN3nLTRC	<RLIN3n_base> + 10_H	√	√	√
RLN3n	LIN/UART mode status register	RLN3nLMST	<RLIN3n_base> + 11_H	√	√	√
RLN3n	LIN/UART status register	RLN3nLST	<RLIN3n_base> + 12_H	√	√	√
RLN3n	LIN/UART error status register	RLN3nLEST	<RLIN3n_base> + 13_H	√	√	√
RLN3n	LIN/UART data field configuration register	RLN3nLDFC	<RLIN3n_base> + 14_H	√	√	√
RLN3n	LIN/UART ID buffer register	RLN3nLIDB	<RLIN3n_base> + 15_H	√	√	√
RLN3n	LIN checksum buffer register	RLN3nLCBR	<RLIN3n_base> + 16_H	√	√	—
RLN3n	UART data buffer 0 register	RLN3nLUDB0	<RLIN3n_base> + 17_H	—	—	√
RLN3n	LIN/UART data buffer 1 register	RLN3nLDBR1	<RLIN3n_base> + 18_H	√	√	√
RLN3n	LIN/UART data buffer 2 register	RLN3nLDBR2	<RLIN3n_base> + 19_H	√	√	√
RLN3n	LIN/UART data buffer 3 register	RLN3nLDBR3	<RLIN3n_base> + $1A_H$	√	√	√
RLN3n	LIN/UART data buffer 4 register	RLN3nLDBR4	<RLIN3n_base> + $1B_H$	√	√	√
RLN3n	LIN/UART data buffer 5 register	RLN3nLDBR5	<RLIN3n_base> + $1C_H$	√	√	√
RLN3n	LIN/UART data buffer 6 register	RLN3nLDBR6	<RLIN3n_base> + $1D_H$	√	√	√
RLN3n	LIN/UART data buffer 7 register	RLN3nLDBR7	<RLIN3n_base> + $1E_H$	√	√	√
RLN3n	LIN/UART data buffer 8 register	RLN3nLDBR8	<RLIN3n_base> + $1F_H$	√	√	√
RLN3n	UART operation enable register	RLN3nLUOER	<RLIN3n_base> + 20_H	—	—	√
RLN3n	UART option register 1	RLN3nLUOR1	<RLIN3n_base> + 21_H	—	—	√
RLN3n	UART transmission data register	RLN3nLUTDR	<RLIN3n_base> + 24_H	—	—	√
RLN3n	UART transmission data register L	RLN3nLUTDRL	<RLIN3n_base> + 24_H	—	—	√
RLN3n	UART transmission data register H	RLN3nLUTDRH	<RLIN3n_base> + 25_H	—	—	√
RLN3n	UART reception data register	RLN3nLURDR	<RLIN3n_base> + 26_H	—	—	√
RLN3n	UART reception data register L	RLN3nLURDRL	<RLIN3n_base> + 26_H	—	—	√
RLN3n	UART reception data register H	RLN3nLURDRH	<RLIN3n_base> + 27_H	—	—	√
RLN3n	UART wait transmission data register	RLN3nLUWTDR	<RLIN3n_base> + 28_H	—	—	√
RLN3n	UART wait transmission data register L	RLN3nLUWTDRL	<RLIN3n_base> + 28_H	—	—	√
RLN3n	UART wait transmission data register H	RLN3nLUWTDRH	<RLIN3n_base> + 29_H	—	—	√

Note: √: Used, —: Not used

When writing to a register not used, write the value after reset.

13.3.2 LIN Master Related Registers

13.3.2.1 RLN3nLWBR ─ LIN Wake-Up Baud Rate Select Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 01_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NSPB[3:0]				LPRS[2:0]			LWBR0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.11 RLN3nLWBR Register Contents

Bit Position	Bit Name	Function
7 to 4	NSPB[3:0]	Bit Sampling Count Select b7 b4 0 0 0 0: 16 sampling 1 1 1 1: 16 sampling Settings other than the above are prohibited.
3 to 1	LPRS[2:0]	Prescaler Clock Select b3 b1 0 0 0: 1/1 0 0 1: 1/2 0 1 0: 1/4 0 1 1: 1/8 1 0 0: 1/16 1 0 1: 1/32 1 1 0: 1/64 1 1 1: 1/128
0	LWBR0	Wake-up Baud Rate Select 0: In LIN wake-up mode, the clock specified in the LCKS bit of the RLN3nLMD register is used. (for LIN1.3) 1: In LIN wake-up mode, the clock fa is used regardless of the setting in the LCKS bit of the RLN3nLMD register. (for LIN2.x)

Set the RLN3nLWBR register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

NSPB[3:0] Bits (Bit Sampling Count Select)

These bits select the number of sampling in one Tbit (reciprocal of the baud rate).
In LIN master mode (LIN/UART mode select bits in LIN mode register = 00_B), set these bits to 0000_B or 1111_B (16 sampling).

LPRS[2:0] Bits (Prescaler Clock Select)

These bits select the frequency division ratio for the prescaler.
The LIN communication clock source is divided by this prescaler.

LWBR0 Bit (Wake-up Baud Rate Select)

When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the RLN3nLWBR register to 0. This allows the 2.5-Tbit or longer low-level width of the input signal to be measured.

When LIN Specification Package Revision 2.x is used, set the LWBR0 bit to 1. Setting the LWBR0 bit to 1 selects the LIN system clock (fLIN) as fa regardless of the setting of the RLN3nLMD.LCKS bit (the LCKS bit is not changed). This allows the 2.5-Tbit or longer low-level width of the input signal to be measured.

Setting the baud rate to 19200 bps while fa is selected allows the 130 µs or longer low-level width of the input signal to be detected during LIN wake-up mode regardless of the setting of the RLN3nLMD.LCKS bit.

13.3.2.2 RLN3nLBRP0 — LIN Baud Rate Prescaler 0 Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 02_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LBRP0[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.12 RLN3nLBRP0 Register Contents

Bit Position	Bit Name	Function
7 to 0	LBRP0[7:0]	Assuming that the value set in this register is N (0 to 255), the baud rate prescaler divides the frequency of the prescaler clock by N + 1. Setting range: 00_H to FF_H

Set the RLN3nLBRP0 register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

The value set in this register is used to control the frequency of baud rate clock sources fa, fb, and fc.

Assuming that the value set in this register is N, baud rate prescaler 0 divides the frequency of the clock that is selected by the LPRS bits by N + 1.

13.3.2.3 RLN3nLBRP1 — LIN Baud Rate Prescaler 1 Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 03_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LBRP1[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.13 RLN3nLBRP1 Register Contents

Bit Position	Bit Name	Function
7 to 0	LBRP1[7:0]	Assuming that the value set in this register is M (0 to 255), the baud rate prescaler divides the frequency of the prescaler clock by M + 1. Setting range: 00_H to FF_H

Set the RLN3nLBRP1 register when the OMM0 bit in the RLN3LMST register is 0_B (in LIN reset mode).

The value set in this register is used to control the frequency of baud rate clock source fd.

Assuming that the value set in this register is M, baud rate prescaler 1 divides the frequency of the clock that is selected by the LPRS bits (prescaler clock select bits) by M + 1.

13.3.2.4 RLN3nLSTC — LIN Self-Test Control Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LSTM[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.14 RLN3nLSTC Register Contents

Bit Position	Bit Name	Function
7 to 0	LSTM[7:0]	LIN Self-Test Mode 00_H: The module is not in LIN self-test mode. 01_H: The module is in LIN self-test mode. Writing $A7_H$, 58_H, and 01_H successively to the RLN3nLSTC register places the module into LIN self-test mode.

The RLN3nLSTC register cancels protection of LIN self-test mode.

Set the RLN3nLSTC register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

Writing $A7_H$, 58_H, and 01_H successively to the RLN3nLSTC register places the module into LIN self-test mode.

When successive writing is completed thus placing LIN self-test mode to be entered, the LSTM bit is set to 1.

Do not write any other value during successive writing.

For making transition to LIN self-test mode, see **Section 13.9, LIN Self-Test Mode**.

When read, bits 6 to 1 return "000000_B", and bit 7 returns an undefined value.

LSTM Bit (LIN Self-Test Mode)

When transition to LIN self-test mode is completed, the LSTM bit is set to 1.
For leaving LIN self-test mode, see **Section 13.9, LIN Self-Test Mode**
Writing 1 to this bit does not affect the value of the RLN3nLSTC register if it is not a part of successive writing of $A7_H$, 58_H, and 01_H.

13.3.2.5 RLN3nLMD — LIN Mode Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	LRDNFS	LIOS	LCKS[1:0]		LMD[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.15 RLN3nLMD Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	LRDNFS	LIN Reception Data Noise Filtering Disable 0: The noise filter is enabled. 1: The noise filter is disabled.
4	LIOS	LIN Interrupt Output Select 0: RLIN3n interrupt is used. 1: RLIN3n transmission interrupt, RLIN3n successful reception interrupt, and RLIN3n status interrupt are used.
3, 2	LCKS[1:0]	LIN System Clock Select b3 b2 0 0: fa (Clock generated by baud rate prescaler 0) 0 1: fb (1/2 clock generated by baud rate prescaler 0) 1 0: fc (1/8 clock generated by baud rate prescaler 0) 1 1: fd (1/2 clock generated by baud rate prescaler 1)
1, 0	LMD[1:0]	LIN/UART Mode Select b1 b0 0 0: LIN master mode

Set the RLN3nLMD register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode)

LRDNFS Bit (LIN Reception Data Noise Filtering Disable)

The LRDNFS bit enables or disables the noise filter when receiving data.
With 0 set, the noise filter is enabled when receiving data.
With 1 set, the noise filter is disabled when receiving data.

LIOS Bit (LIN Interrupt Output Select)

The LIOS bit selects the number of interrupt outputs from the LIN/UART interface.
With 0 set, the RLIN3 interrupt is generated from the LIN/UART interface.
With 1 set, the RLIN3n transmission interrupt, RLIN3n successful reception interrupt, and RLIN3n status interrupt are generated from the LIN/UART interface.
For each interrupt source, see **Section 13.4, Interrupt Sources**.

LCKS[1:0] Bits (LIN System Clock Select)

The LCKS bits select the clock to be input to the protocol controller.

With 00_B set, the protocol controller is provided with fa (clock generated by baud rate prescaler 0).
With 01_B set, the protocol controller is provided with fb (1/2 clock generated by baud rate prescaler 0).
With 10_B set, the protocol controller is provided with fc (1/8 clock generated by baud rate prescaler 0).
With 11_B set, the protocol controller is provided with fd (1/2 clock generated by baud rate prescaler 1).
With 1_B is set in the LWBR0 bit in the RLN3nLWBR register (LIN 2.x is used), and the RLN3nLMST register is 01_H (LIN wake-up mode), the protocol controller is provided with fa regardless of the setting of the bit (the LCKS bit is not changed)

LMD[1:0] Bits (LIN/UART Mode Select)

The LMD bits select the LIN/UART interface mode.
To use the LIN/UART interface as an LIN master, set these bits to 00_B.

13.3.2.6 RLN3nLBFC — LIN Break Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 09_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	BDT[1:0]		BLT[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.16 RLN3nLBFC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5, 4	BDT[1:0]	Transmission Break Delimiter (High Level) Width Select b5 b4 0 0: 1 Tbit 0 1: 2 Tbits 1 0: 3 Tbits 1 1: 4 Tbits
3 to 0	BLT[3:0]	Transmission Break (Low Level) Width Select b3 b0 0 0 0 0: 13 Tbits 0 0 0 1: 14 Tbits 0 0 1 0: 15 Tbits : 1 1 1 0: 27 Tbits 1 1 1 1: 28 Tbits

Set the RLN3nLBFC register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

Some combinations of the set values result in the length of a frame exceeding the timeout time. Set the appropriate values in this register.

BDT[1:0] Bits (Transmission Break Delimiter (High Level) Width Select)

This bit is used to set the break high level width of transmission frame header. 1 Tbit to 4 Tbits can be set.

BLT[3:0] Bits (Transmission Break (Low Level) Width Select)

This BLT bits set the break low level width of transmission frame header.
13 Tbits to 28 Tbits can be set.

13.3.2.7 RLN3nLSC — LIN Space Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0A_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	IBS[1:0]		—	IBHS[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Table 13.17 RLN3nLSC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5, 4	IBS[1:0]	Inter-Byte Space Select b5 b4 0 0: 0 Tbit 0 1: 1 Tbit 1 0: 2 Tbits 1 1: 3 Tbits
3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2 to 0	IBHS[2:0]	Inter-Byte Space (Header)/Response Space Select b2 b0 0 0 0: 0 Tbit 0 0 1: 1 Tbit 0 1 0: 2 Tbits 0 1 1: 3 Tbits 1 0 0: 4 Tbits 1 0 1: 5 Tbits 1 1 0: 6 Tbits 1 1 1: 7 Tbits

Set the RLN3nLSC register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

Some combinations of the set values result in the length of a frame or a response exceeding the timeout time. Set the appropriate values in this register.

IBS[1:0] Bits (Inter-Byte Space Select)

The IBS bits set the width of the inter-byte space of the transmission frame response field.
0 Tbit to 3 Tbits can be set.
These bits are enabled only during response transmission; these are disabled during response reception.

IBHS[2:0] Bits (Inter-Byte Space (Header)/Response Space Select)

The IBHS bits set the width of the inter-byte space (header) of the transmission frame header field and the response space.
0 Tbit to 7 Tbits can be set.
The response space setting is enabled only during response transmission; setting is disabled during response reception.
The inter-byte space (header) is equal to the response space.

13.3.2.8 RLN3nLWUP — LIN Wake-Up Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0B_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	WUTL[3:0]				—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R

Table 13.18 RLN3nLWUP Register Contents

Bit Position	Bit Name	Function
7 to 4	WUTL[3:0]	Wake-up Transmission Low Level Width Select b7 b4 0 0 0 0: 1 Tbit 0 0 0 1: 2 Tbits 0 0 1 0: 3 Tbits 0 0 1 1: 4 Tbits : 1 1 0 0: 13 Tbits 1 1 0 1: 14 Tbits 1 1 1 0: 15 Tbits 1 1 1 1: 16 Tbits
3 to 0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Set the RLN3nLWUP register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

WUTL[3:0] Bits (Wake-up Transmission Low Level Width Select)

The WUTL bits set the low level width of the wake-up signal transmission.
1 Tbit to 16 Tbits can be set.

With 1 is set in the LWBR0 bit in the RLN3nLWBR register (LIN 2.x is used), fa is selected as the LIN system clock (fLIN) regardless of the setting of the RLN3nLMD.LCKS bit (the LCKS bit is not changed).

13.3.2.9 RLN3nLIE — LIN Interrupt Enable Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	SHIE	ERRIE	FRCIE	FTCIE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 13.19 RLN3nLIE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	SHIE	Successful Header Transmission Interrupt Request Enable 0: Disables successful header transmission interrupt request. 1: Enables successful header transmission interrupt request.
2	ERRIE	Error Detection Interrupt Request Enable 0: Disables error detection interrupt request. 1: Enables error detection interrupt request.
1	FRCIE	Successful Frame/Wake-up Reception Interrupt Request Enable 0: Disables successful frame/wake-up reception interrupt request. 1: Enables successful frame/wake-up reception interrupt request.
0	FTCIE	Successful Frame/Wake-up Transmission Interrupt Request Enable 0: Disables successful frame/wake-up transmission interrupt request. 1: Enables successful frame/wake-up transmission interrupt request.

Set the RLN3nLIE register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

SHIE Bit (Successful Header Transmission Interrupt Request Enable)

The SHIE bit enables or disables interrupt request upon successful transmission of a header.
With 0 set, the interrupt request for RLIN3n transmission is not generated when the HTRC flag in the RLN3nLST register is set to 1.
With 1 set, the interrupt request for RLIN3n transmission is generated when the HTRC flag in the RLN3nLST register is set to 1.

ERRIE Bit (Error Detection Interrupt Request Enable)

The ERRIE bit enables or disables an interrupt request upon detection of an error.
With 0 set, the interrupt request for RLIN3n status is not generated when the ERR flag in the RLN3nLST register is set to 1.
With 1 set, the interrupt request for RLIN3n status is generated when the ERR flag in the RLN3nLST register is set to 1.
Error types that are interrupt sources are the bit error, physical bus error, frame/response timeout error, framing error, checksum error, and response preparation error. Detection of the bit error, physical bus error, frame/response timeout error, and framing error can be enabled or disabled using the RLN3nLEDE register.

FRCIE Bit (Successful Frame/Wake-up Reception Interrupt Request Enable)

The FRCIE bit enables or disables an interrupt request upon successful reception of a frame or a wake-up signal (counting of low level width of the input signal).
With 0 set, the interrupt request for successful RLIN3n reception is not generated when the FRC flag in the RLN3nLST register is set to 1.
With 1 set, the interrupt request for successful RLIN3n reception is generated when the FRC flag in the RLN3nLST register is set to 1.

FTCIE Bit (Successful Frame/Wake-up Transmission Interrupt Request Enable)

The FTCIE bit enables or disables an interrupt request upon successful transmission of a frame or a wake-up signal.
With 0 set, the interrupt request for RLIN3n transmission is not generated when the FTC flag in the RLN3nLST register is set to 1.
With 1 set, the interrupt request for RLIN3n transmission is generated when the FTC flag in the RLN3nLST register is set to 1.

13.3.2.10 RLN3nLEDE — LIN Error Detection Enable Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0D_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LTES	—	—	—	FERE	FTERE	PBERE	BERE
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Table 13.20 RLN3nLEDE Register Contents

Bit Position	Bit Name	Function
7	LTES	Timeout Error Select 0: Frame timeout error 1: Response timeout error
6 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	FERE	Framing Error Detection Enable*1 0: Disables framing error detection. 1: Enables framing error detection.
2	FTERE	Timeout Error Detection Enable 0: Disables frame/response timeout error detection. 1: Enables frame/response timeout error detection.
1	PBERE	Physical Bus Error Detection Enable 0: Disables physical bus error detection. 1: Enables physical bus error detection.
0	BERE	Bit Error Detection Enable*1 0: Disables bit error detection. 1: Enables bit error detection.

Note 1. Bits FERE and BERE must be set to 1.

Set the RLN3nLEDE register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode)

LTES Bit (Timeout Error Select)

The LTES bit selects the timeout function to be used.
With 0 set, the timeout function applies to frame timeout.
With 1 set, the timeout function applies to response timeout.
For details on the timeout error, see **Section 13.7.7, Error Status**.

FERE Bit (Framing Error Detection Enable)

The FERE bit enables or disables detection of the framing error.
With 0 set, the framing error is not detected.
With 1 set, the framing error is detected.
With this device, always set this bit to 1. The detection result is indicated in the FER flag in the RLN3nLEST register.
For details on the framing error, see **Section 13.7.7, Error Status**.

FTERE Bit (Timeout Error Detection Enable)

The FTERE bit enables or disables detection of the frame timeout error or the response timeout error.
With 0 set, the frame timeout error or response timeout error is not detected.
With 1 set, the frame timeout error or response timeout error is detected.
When this bit is set to 1, the detection result is indicated in the FTER flag in the RLN3nLEST register.
With the LTES bit, either the frame timeout error or response timeout error can be selected. Do not use the timeout error if response data of 9 bytes or more is to be transmitted or received. For details on the timeout error, see **Section 13.7.7, Error Status**.

PBERE Bit (Physical Bus Error Detection Enable)

The PBERE bit enables or disables detection of the physical bus error.
With 0 set, the physical bus error is not detected.
With 1 set, the physical bus error is detected.
When this bit is set to 1, the detection result is indicated in the PBER flag in the RLN3nLEST register. For details on the physical bus error, see **Section 13.7.7, Error Status**.

BERE Bit (Bit Error Detection Enable)

The BERE bit enables or disables detection of the bit error.
With 0 set, the bit error is not detected.
With 1 set, the bit error is detected.
With this device, always set this bit to 1. The detection result is indicated in the BER flag in the RLN3nLEST register. For details on the bit error, see **Section 13.7.7, Error Status**.

13.3.2.11 RLN3nLCUC — LIN Control Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0E_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OM1	OM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 13.21 RLN3nLCUC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	OM1	LIN Mode Select 0: LIN wake-up mode is caused. 1: LIN operation mode is caused.
0	OM0	LIN Reset 0: LIN reset mode is caused. 1: LIN reset mode is canceled.

Set the RLN3nLCUC register to 01_H to cause a transition to LIN wake-up mode after canceling LIN reset mode, and set the register to 03_H to cause a transition to LIN operation mode.

In LIN self-test mode, set the RLN3nLCUC register to 03_H after a transition to LIN self-test mode is completed.

After a value is written to this register, confirm that the value written is actually indicated in the RLN3nLMST register before writing another value.

OM1 Bit (LIN Mode Select)

The OM1 bit selects the specific operation mode (either LIN wake-up mode or LIN operation mode) after canceling LIN reset mode.
With 0 set, LIN wake-up mode is caused.
With 1 set, LIN operation mode is caused.
This bit is valid only when the OMM0 bit in the RLN3nLMST register is 1.
Writing a value to this bit is disabled while the FTS bit in the RLN3nLTRC register is 1.

OM0 Bit (LIN Reset)

The OM0 bit selects either causing a transition to LIN reset mode or canceling LIN reset mode.
With 0 set, LIN reset mode is caused.
With 1 set, LIN reset mode is canceled.

13.3.2.12 RLN3nLTRC — LIN Transmission Control Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RTS	FTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 13.22 RLN3nLTRC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	RTS	Response Transmission/Reception Start 0: Response transmission/reception is stopped in frame separate mode. 1: Response transmission/reception is started in frame separate mode.
0	FTS	Frame Transmission/Wake-up Transmission /Reception Start 0: Frame Transmission/wake-up transmission/reception is stopped. 1: Frame Transmission/wake-up transmission reception is started.

RTS Bit (Response Transmission/Reception Start)

Set the RTS bit to 1 in frame separate mode after header transmission is started (FTS bit is 1) and response transmission data is ready. Once set, this bit is automatically cleared to 0 upon completion of frame communication (including error detection) or transition to LIN reset mode.
Only 1 can be written to this bit; 0 cannot be written.
To write 1 to this bit, write 02_H to the RLN3nLTRC register using the store instruction.
Writing a value to this bit is disabled when the OMM0 bit of the RLN3nLMST register is 0_B (in LIN reset mode).
Writing a value to this bit is disabled when the FTS bit is 0 (frame transmission or wake-up transmission/reception is halted).
When response data of 9 bytes or more is to be transmitted or received, set this bit to 1 each time a data group (variable from 0 to 8 bytes) is transmitted or received. Once set, this bit is automatically cleared to 0 at the end of communication or transition to LIN reset mode.

FTS Bit (Frame Transmission/Wake-up Transmission/Reception Start)

Set the FTS bit to 1 to start frame transmission and reception.
Also set this bit to 1 to allow wake-up transmission and reception (counting of the low level width of the input signal).
Only 1 can be written to this bit; 0 cannot be written.
Writing a value to this bit is disabled when the OMM0 bit of the RLN3nLMST register is 0_B (in LIN reset mode).
This bit is set to 0 upon error detection of frame or wake-up communication (including error detection) and transition to LIN reset mode.

13.3.2.13 RLN3nLMST — LIN Mode Status Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 11_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OMM1	OMM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 13.23 RLN3nLMST Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned.
1	OMM1	LIN Mode Status Monitor 0: The module is in LIN wake-up mode. 1: The module is in LIN operation mode.
0	OMM0	LIN Reset Status Monitor 0: The module is in LIN reset mode. 1: The module is not in LIN reset mode.

OMM1 Bit (LIN Mode Status Monitor)

The OMM1 bit indicates the current operating mode.

OMM0 Bit (LIN Reset Status Monitor)

The OMM0 bit indicates the current operating mode.

13.3.2.14 RLN3nLST — LIN Status Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 12_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	HTRC	D1RC	—	—	ERR	—	FRC	FTC
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R/W	R/W

Table 13.24 RLN3nLST Register Contents

Bit Position	Bit Name	Function
7	HTRC	Successful Header Transmission Flag 0: Header transmission has not been completed. 1: Header transmission has been completed.
6	D1RC	Successful Data 1 Reception Flag These bits are always read as 0. The write value should always be 0.
4, 5	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	ERR	Error Detection Flag 0: No error has been detected. 1: Error has been detected.
2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	FRC	Frame/Wake-up Reception Complete Flag 0: Frame or wake-up reception has not been completed. 1: Frame or wake-up reception has been completed.
0	FTC	Frame/Wake-up Transmission Complete Flag 0: Frame or wake-up transmission has not been completed. 1: Frame or wake-up transmission has been completed.

The RLN3nLST register is automatically cleared to 00_H upon transition to LIN reset mode and start of the next communication (when the FTS bit of the RLN3nLTRC register is 1).

In LIN reset mode, this register cannot be written to. In LIN reset mode, the register retains 00_H.

To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

HTRC Flag (Successful Header Transmission Flag)

Only 0 can be written to the HTRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The HTRC flag is set to 1 upon completion of header transmission. Here, an interrupt request for RLN3n transmission is generated if the SHIE bit in the RLN3nLIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode.

D1RC Flag (Successful Data 1 Reception Flag)

Only 0 can be written to the D1RC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The D1RC flag is set to 1 upon completion of data 1 reception. Here, an interrupt request is not generated. Here, an interrupt is not generated. To clear the bit to 0 before the next communication Only 0 can be written to the D1RC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The D1RC flag is set to 1 upon completion of data 1 reception. Here, an interrupt request is not generated. Here, an interrupt is not generated. To clear the bit to 0 before the next communication

ERR Flag (Error Detection Flag)

The ERR flag is set to 1 upon detection of an error (when the value of at least one of the flags of the RLN3nLEST register is set to 1). Here, an interrupt request for RLN3n status is generated if the ERRIE bit in the RLN3nLIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the RPER, CSER, FER, FTER, PBER, and BER flags in the RLN3nLEST register in LIN operation mode or LIN wake-up mode. This clears the ERR flag to 0.

FRC Flag (Frame/Wake-up Reception Complete Flag)

Only 0 can be written to the FRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FRC flag is set to 1 upon completion of frame or wake-up reception. Here, an interrupt request for RLN3n reception complete is generated if the FRCIE bit in the RLN3nLIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode or LIN wake-up mode.
When response data of 9 bytes or more is to be received, this bit is set to 1 each time a data group (variable from 0 to 8 bytes) is received. Write 0 before starting reception of the next data group.

FTC Flag (Frame/Wake-up Transmission Complete Flag)

Only 0 can be written to the FTC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FTC flag is set to 1 upon completion of frame or wake-up transmission. Here, an interrupt request for RLN3n transmission is generated if the FTCIE bit in the RLN3nLIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode or LIN wake-up mode.
When response data of 9 bytes or more is to be transmitted, this bit is set to 1 each time a data group (variable from 0 to 8 bytes) is transmitted. Write 0 before starting transmission of the next data group.

13.3.2.15 RLN3nLEST — LIN Error Status Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 13_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	RPER	—	CSER	—	FER	FTER	PBER	BER
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R	R/W	R/W	R/W	R/W

Table 13.25 RLN3nLEST Register Contents

Bit Position	Bit Name	Function
7	RPER	Response Preparation Error Flag 0: Response preparation error has not been detected. 1: Response preparation error has been detected.
6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	CSER	Checksum Error Flag 0: Checksum error has not been detected. 1: checksum error has been detected.
4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	FER	Framing Error Flag 0: Framing error has not been detected. 1: Framing error has been detected.
2	FTER	Timeout Error Flag 0: Frame/response timeout error has not been detected. 1: Frame/response timeout error has been detected.
1	PBER	Physical Bus Error Flag 0: Physical bus error has not been detected. 1: Physical bus error has been detected.
0	BER	Bit Error Flag 0: Bit error has not been detected. 1: Bit error has been detected.

The RLN3nLEST register is automatically cleared to 00_H upon transition to LIN reset mode and start of the next communication (when the FTS bit of the RLN3nLTRC register is 1).

In LIN reset mode, this register cannot be written to. In LIN reset mode, the register retains 00_H.

When the FTS bit in the RLN3nLTRC register is 1 (frame transmission or wake-up transmission/reception is started), do not write a value to this register.

To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

RPER Flag (Response Preparation Error Flag)

Only 0 can be written to the RPER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The RPER flag is set to 1 upon response preparation error detection. To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode.

CSER Flag (Checksum Error Flag)

Only 0 can be written to the CSER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The CSER flag is set to 1 upon checksum error detection. To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode.

FER Flag (Framing Error Flag)

Only 0 can be written to the FER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the value of the FERE bit of the RLN3nLEDE register is 1 (framing error detection enabled), the FER flag is set to 1 upon framing error detection. To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode.

FTER Flag (Timeout Error Flag)

Only 0 can be written to the FTER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the FTERE bit of the RLN3nLEDE register is 1 (frame/response timeout error detection enabled), the FTER flag is set to 1 upon frame timeout error or response timeout error detection. To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode.

PBER Flag (Physical Bus Error Flag)

Only 0 can be written to the PBER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the PBERE bit of the RLN3nLEDE register is 1 (physical bus error detection enabled), the PBER flag is set to 1 upon physical bus error detection. To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode or LIN wake-up mode.

BER Flag (Bit Error Flag)

Only 0 can be written to the BER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the BERE bit of the RLN3nLEDE register is 1 (bit error detection enabled), the BER flag is set to 1 upon bit error detection. To clear the bit to 0 before the next communication (when the FTS bit of the RLN3nLTRC register is 1), write 0 to the bit in LIN operation mode or LIN wake-up mode.

13.3.2.16 RLN3nLDFC — LIN Data Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LSS	FSM	CSM	RFT	RFDL[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.26 RLN3nLDFC Register Contents

Bit Position	Bit Name	Function
7	LSS	Transmission/Reception Continuation Select 0: The data group to be transmitted/received next is the last one. 1: The data group to be transmitted/received next is not the last one. (Checksum is not included.)
6	FSM	Frame Separate Mode Select 0: Frame separate mode is not set. 1: Frame separate mode is set.
5	CSM	Checksum Select 0: Classic checksum mode 1: Enhanced checksum mode
4	RFT	Response Field Communication Direction Select 0: Reception 1: Transmission
3 to 0	RFDL[3:0]	Response Field Length Select b3 b0 0 0 0 0: 0 byte (+ checksum) 0 0 0 1: 1 byte (+ checksum) 0 0 1 0: 2 bytes (+ checksum) : 0 1 1 1: 7 bytes (+ checksum) 1 0 0 0: 8 bytes (+ checksum) Settings other than the above are prohibited.

LSS Bit (Transmission/Reception Continuation Select)

The LSS bit indicates that the data group to be transmitted or received next is not the last data group when response data of 9 bytes or more is to be transmitted or received.

With 0 set, data and checksum are transmitted or received because the next data group to be transmitted or received is the last one.

With 1 set, only data is transmitted or received, and the checksum is not included because the next data group to be transmitted or received is not the last one.

Set the LSS bit only when the FSM bit is 1 (frame separate mode) and response data of 9 bytes or more is to be transmitted or received.

Set the LSS bit only when the RTS bit in the RLN3nLTRC is 0 (response transmit/receive is stopped).

FSM Bit (Frame Separate Mode Select)

The FSM bit sets the response communication mode.
With 0 set, frame separate mode is not selected. In this case, after header transmission is started (the FTS bit in the RLN3nLTRC register is 1), response is transmitted/received without the RTS bit in the RLN3nLTRC register being set.
With 1 set, frame separate mode is selected. If the RTS bit of the RLN3nLTRC register is set to 1 during header transmission, response transmission is executed after header transmission is completed.
For response reception which is 8 bytes or less (the RFT bit is 0), set the FSM bit to 0. When causing a transition to LIN self-test mode, set this bit to 0 before transition.
For details on frame separate mode, see **Section 13.7.4.1, Transmission of LIN Frames**.
Set this bit when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).
When response data of 9 bytes or more is to be transmitted or received, set the FSM bit to 1.

CSM Bit (Checksum Select)

The CSM bit sets the checksum mode.
With 0 set, classic checksum mode is selected.
With 1 set, enhanced checksum mode is selected.
When the timeout error is used (the FTERE bit in the RLN3nLEDE register is 1), the specific timeout time depends on the setting of this bit. For details on the bit error, see **Section 13.7.7, Error Status**.
Set this bit when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).
When response data of 9 bytes or more is to be transmitted or received, do not change the CSM bit setting after the first data group through the last data group.
During communication of response data of 9 bytes or more, only the last data group (the LSS bit is 0) includes the checksum, and no other groups (the LSS bit is 1) include the checksum.

RFT Bit (Response Field Communication Direction Select)

The RFT bits set the direction of the response field/wake-up signal communication.
With 0 set, reception is performed in the response field. In LIN wake-up mode, wake-up reception is performed (low level width of the input signal is counted).
With 1 set, transmission is performed in the response field. In LIN wake-up mode, wake-up transmission is performed.
Set this bit when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).
When response data of 9 bytes or more is to be transmitted or received, do not change the RFT bit setting after the first data group through the last data group.

RFDL[3:0] Bits (Response Field Length Select)

The RFDL bits set the length of the response field data.
The data length can be 0 to 8 bytes excluding the checksum size.
To transmit response data with the FSM bit set to 0 (not frame separate mode), set the RFDL bits before header transmission (the FTS bit in the RLN3nLTRC register is 0).
To transmit response data with the FSM bit set to 1 (frame separate mode), set the RFDL bits before response transmission (the RTS bit in the RLN3nLTRC register is 0).
To receive response data, set the RFDL bits before header transmission (the FTS bit in the RLN3nLTRC register is 0).
When response data of 9 bytes or more is to be transmitted or received, set the RFDL bits before data group transmission/reception (RTS bit in the RLN3nLTRC register is 0).

Only the last data group (the LSS bit is 0) includes the checksum, and no other groups (the LSS bit is 1) include the checksum.

13.3.2.17 RLN3nLIDB — LIN ID Buffer Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 15_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	IDP1	IDP0	ID[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.27 RLN3nLIDB Register Contents

Bit Position	Bit Name	Function
7	IDP1	Parity Setting (P1) Sets the parity bits (P1) to be transmitted in the ID field.
6	IDP0	Parity Setting (P0) Sets the parity bits (P0) to be transmitted in the ID field.
5 to 0	ID[5:0]	ID Setting Sets the 6-bit ID value to be transmitted in the ID field.

Set the RLN3nLIDB register when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

In LIN self-test mode, this register operates as described below.
Write the value to be transmitted before communication. After completion of frame transmission/reception (after loopback), the reversed value of the received value can be read.

For details about the LIN self-test mode, see **Section 13.9, LIN Self-Test Mode**.

IDP[1:0] Bits (Parity Setting)

The IDP bits set the parity bits (P0 and P1) to be transmitted in the ID field of the LIN frame. IDP0 for P0 and IDP1 for P1.
Since parity is not automatically calculated, set the calculation result. Note that if the erroneous result is set, it is transmitted as is.

ID[5:0] Bits (ID Setting)

The ID bit sets the 6-bit ID value to be transmitted in the ID field of the LIN frame.

13.3.2.18 RLN3nLCBR — LIN Checksum Buffer Register

Access: This register can only be read in 8-bit units. In LIN self-test mode, this register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 16_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CKSM[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.28 RLN3nLCBR Register Contents

Bit Position	Bit Name	Function
7 to 0	CKSM[7:0]	Holds the checksum value transmitted or received.

In LIN mode, this register operates as follows:

- When the RFT bit in the RLN3nLDFC register is 1 (transmission):
  The value transmitted can be read from the register. Read the value after transmission is completed.
  Writing to this register is invalid.
- When the RFT bit in the RLN3nLDFC register is 0 (reception):
  The value received can be read from the register. Read the value after reception is completed.
  Writing to this register is invalid.

In LIN self-test mode, this register operates as follows:

- When the RFT bit in the RLN3nLDFC register is 1 (transmission):
  After completion of the frame transmission (after loopback), the reversed value of the received value can be read.
- When the RFT bit in the RLN3nLDFC register is 0 (reception):
  Write the value to be received before communication. After completion of frame transmission/reception (after loopback), the reversed value of the received value can be read.

For details about the LIN self-test mode, see Section **13.9, LIN Self-Test Mode**.

Set the RLN3nLCBR register when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

When response data of at least 9 bytes is to be transmitted or received, the checksum is only appended to the last data group, so this register is not updated for the other data groups.

13.3.2.19 RLN3nLDBRb — LIN Data Buffer b Register (b = 1 to 8)

Access: This register can be read/written in 8-bit units.

Address: RLN3nLDBR1: <RLIN3n_base> + 18_H
RLN3nLDBR2: <RLIN3n_base> + 19_H
RLN3nLDBR3: <RLIN3n_base> + $1A_H$
RLN3nLDBR4: <RLIN3n_base> + $1B_H$
RLN3nLDBR5: <RLIN3n_base> + $1C_H$
RLN3nLDBR6: <RLIN3n_base> + $1D_H$
RLN3nLDBR7: <RLIN3n_base> + $1E_H$
RLN3nLDBR8: <RLIN3n_base> + $1F_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LDB[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.29 RLN3nLDBRb Register Contents

Bit Position	Bit Name	Function
7 to 0	LDB[7:0]	Sets the data for transmission or holds the received data. Setting range: 00_H to FF_H

- For response transmission:
  The LDBRn registers set the data to be transmitted in the response field. Use these registers with the following settings.
  - RFT in RLN3nLDFC register is 1 (transmission)
  - FSM in RLN3nLDFC register is 0 (not frame separate mode)
  - FTS bit in RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted)

  or
  - RFT in RLN3nLDFC register is 1 (transmission)
  - FSM in RLN3nLDFC register is 1 (frame separate mode)
  - RTS in RLN3nLTRC register is 0 (response transmission/reception is halted)
- For response reception:
  The LDBRn registers hold the data received in the response field.
  Any previously received data are overwritten. The data including bytes where an error was found are stored in these registers in response to the detection of an error.
  Do not read these registers when the FTS bit is 1 (frame transmission or wake-up transmission/reception is started).

- For transmission of response data of 9 bytes or more:
  Use the LDBRn registers with the following settings.
  - RFT in RLN3nLDFC register is 1 (transmission)
  - FSM in RLN3nLDFC register is 1 (frame separate mode)
  - RTS in RLN3nLTRC register is 0 (response transmission/reception is halted)
- For reception of response data of 9 bytes or more:
  Do not read these registers when the RTS bit is 1 (response transmission/reception is started).

In LIN self-test mode, these registers operate as described below.

Write the value to be transmitted before communication. After completion of frame transmission/reception (after loopback), the reversed value of the received value can be read.

For details about the LIN self-test mode, see Section **13.9, LIN Self-Test Mode**.

13.3.3 LIN Slave Related Registers

13.3.3.1 RLN3nLWBR — LIN Wake-Up Baud Rate Select Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 01_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NSPB[3:0]				LPRS[2:0]			—
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 13.30 RLN3nLWBR Register Contents

Bit Position	Bit Name	Function
7 to 4	NSPB[3:0]	Bit Sampling Count Select b7 b4 0 0 0 0: 16 sampling 0 0 1 1: 4 sampling 0 1 1 1: 8 sampling 1 1 1 1: 16 sampling Settings other than the above are prohibited.
3 to 1	LPRS[2:0]	Prescaler Clock Select b3 b1 0 0 0: 1/1 0 0 1: 1/2 0 1 0: 1/4 0 1 1: 1/8 1 0 0: 1/16 1 0 1: 1/32 1 1 0: 1/64 1 1 1: 1/128
0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Set the RLN3nLWBR register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

NSPB[3:0] Bits (Bit Sampling Count Select)

The NSPB bits select the number of sampling in one Tbit (reciprocal of the baud rate).
When the frame communication is performed in LIN slave mode (fixed baud rate) (LMD[1:0] bits in the RLN3nLMD register = 11_B), set these bits to "0000_B" or "1111_B" (16 sampling).
When the frame communication is performed in LIN slave mode (auto baud rate) (LMD[1:0] bits in the RLN3nLMD register = 10_B), set these bits to "0011_B" (4 sampling) or "0111_B" (8 sampling).

LPRS[2:0] Bits (Prescaler Clock Select)

The LPRS bits select the frequency division ratio for the prescaler. The LIN communication clock source is divided by this prescaler.
In LIN slave mode (auto baud rate) (LMD[1:0] bits in the RLN3nLMD register = 10_B), set these bits so that the prescaler clock becomes as follows according to the target baud rate.

[Target baud rate]	[Prescaler clock]
1 kbps to 20 kbps	: 4 MHz*1
1 kbps to 2.4 kbps (excluding 2.4 kbps)	: 4 MHz
2.4 kbps to 20 kbps	: 8 MHz to 12 MHz

Note 1. Use the clock with NSPB bits set to "0011_B" (four samplings).

13.3.3.2 RLN3nLBRP01 — LIN Baud Rate Prescaler 01 Register

Access: RLN3nLBRP01 can be read/written in 16-bit units.
RLN3nLBRP0 can be read/written in 8-bit units.
RLN3nLBRP1 can be read/written in 8-bit units.

Address: RLN3nLBRP01: <RLIN3n_base> + 02_H
RLN3nLBRP0: <RLIN3n_base> + 02_H
RLN3nLBRP1: <RLIN3n_base> + 03_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BRP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.31 RLN3nLBRP01 Register Contents

Bit Position	Bit Name	Function
15 to 0	BRP[15:0]	Assuming that the value set in this register is L (0 to 65535), the baud rate prescaler divides the frequency of the prescaler clock by L + 1. Setting range: 0000_H to $FFFF_H$

Set the RLN3nLBRP01 register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

Assuming that the value set in this register is L, the baud rate prescaler divides the frequency of the clock that is selected by the LPRS bits (prescaler clock select bits) in the RLN3nLWBR register by L + 1.

The RLN3nLBRP01 register can be accessed in 8-bit units by registers RLN3nLBRP0 and RLN3nLBRP1.

NOTE

In LIN slave mode [auto baud rate], the system automatically sets the result of baud rate correction to the RLN3nLBRP01 register on successful reception of the sync field.

13.3.3.3 RLN3nLSTC — LIN Self-Test Control Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LSTM[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.32 RLN3nLSTC Register Contents

Bit Position	Bit Name	Function
7 to 0	LSTM[7:0]	LIN Self-Test Mode 00_H: The module is not in LIN self-test mode. 01_H: The module is in LIN self-test mode. Writing $A7_H$, 58_H, and 01_H successively to the RLN3nLSTC register places the module into LIN self-test mode.

The RLN3nLSTC register cancels protection of LIN self-test mode.

Set the RLN3nLSTC register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

Writing $A7_H$, 58_H, and 01_H successively to the RLN3nLSTC register places the module into LIN self-test mode.

When successive writing is completed thus placing LIN self-test mode to be entered, the LSTM bit is set to 1.

Do not write any other value during successive writing.

For making transition to LIN self-test mode, see Section **13.9, LIN Self-Test Mode**.

When read, bits 6 to 1 return "000000_B", and bit 7 returns an undefined value.

LSTM Bit (LIN Self-Test Mode)

When transition to LIN self-test mode is completed, the LSTM bit is set to 1.

For leaving LIN self-test mode, see Section **13.9, LIN Self-Test Mode**.
Writing 1 to this bit does not affect the value of the RLN3nLSTC register if it is not a part of successive writing of $A7_H$, 58_H, and 01_H.

13.3.3.4 RLN3nLMD — LIN Mode Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	LRDNFS	LIOS	—	—	LMD[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 13.33 RLN3nLMD Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	LRDNFS	LIN Reception Data Noise Filtering Disable 0: The noise filter is enabled. 1: The noise filter is disabled.
4	LIOS	LIN Interrupt Output Select 0: RLIN3 interrupt is used. 1: RLIN3n transmission interrupt, RLIN3n successful reception interrupt, and RLIN3n reception status interrupt are used.
3, 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1, 0	LMD[1:0]	LIN/UART Mode Select b1 b0 1 0: LIN Slave mode with Auto Baud rate 1 1: LIN Slave mode with fixed Baud rate

Set the RLN3nLMD register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

LRDNFS Bit (LIN Reception Data Noise Filtering Disable)

The LRDNFS bit enables or disables the noise filter when receiving data.
With 0 set, the noise filter is enabled when receiving data.
With 1 set, the noise filter is disabled when receiving data.

LIOS Bit (LIN Interrupt Output Select)

The LIOS bit selects the number of interrupt outputs from the LIN/UART interface.
With 0 set, the RLIN3 interrupt is generated from the LIN/UART interface.
With 1 set, the RLIN3n transmission interrupt, RLIN3n successful reception interrupt, and RLIN3n reception status interrupt are generated from the LIN/UART interface.
For each interrupt source, see Section **13.4, Interrupt Sources**.

LMD[1:0] Bits (LIN/UART Mode Select)

The LMD bits select the LIN/UART interface mode.

To use this module as an LIN slave, set these bits to "10_B" (auto baud rate) or "11_B" (fixed baud rate).

13.3.3.5 RLN3nLBFC — LIN Break Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 09_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	LBLT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 13.34 RLN3nLBFC Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	LBLT	Reception Break (Low-Level) Detection Width Setting 0: A break (low-level) is detected in 9.5 or 10 Tbits 1: A break (low-level) is detected in 10.5 or 11 Tbits

Set the RLN3nLBFC register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

LBLT Bit (Reception Break (Low-Level) Detection Width Setting)

- When RLN3nLMD.LMD is "10_B" (in LIN slave mode (auto baud rate))
  - 0: Low-level width of 10 Tbits or longer is detected.
  - 1: Low-level width of 11 Tbits or longer is detected.
- When RLN3nLMD.LMD is "11_B" (in LIN slave mode (fixed baud rate))
  - 0: Low-level width of 9.5 Tbits or longer is detected.
  - 1: Low-level width of 10.5 Tbits or longer is detected.

13.3.3.6 RLN3nLSC — LIN Space Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0A_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	IBS[1:0]		—	IBHS[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Table 13.35 RLN3nLSC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5, 4	IBS[1:0]	Inter-Byte Space Select b5 b4 0 0: 0 Tbit 0 1: 1 Tbit 1 0: 2 Tbits 1 1: 3 Tbits
3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2 to 0	IBHS[2:0]	Response Space Setting b2 b0 0 0 0: 0 Tbit 0 0 1: 1 Tbit 0 1 0: 2 Tbits 0 1 1: 3 Tbits 1 0 0: 4 Tbits 1 0 1: 5 Tbits 1 1 0: 6 Tbits 1 1 1: 7 Tbits

Set the RLN3nLSC register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

This register is enabled only during response transmission, and disabled during response reception.

Some combinations of the set values result in the length of a frame or a response exceeding the timeout time. Set the appropriate values in this register.

IBS[1:0] Bits (Inter-Byte Space Select)

These bits set the width of the inter-byte space of the response transmission.
0 Tbit to 3 Tbits can be set.

IBHS[2:0] Bits (Inter-Byte Space (Header)/Response Space Select)

The IBHS bits set the transmission width of the response space.
0 Tbit to 7 Tbits can be set.

13.3.3.7 RLN3nLWUP — LIN Wake-Up Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0B_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	WUTL[3:0]				—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R

Table 13.36 RLN3nLWUP Register Contents

Bit Position	Bit Name	Function
7 to 4	WUTL[3:0]	Wake-up Transmission Low level Width Select b7 b4 0 0 0 0: 1 Tbit 0 0 0 1: 2 Tbits 0 0 1 0: 3 Tbits 0 0 1 1: 4 Tbits : 1 1 0 0: 13 Tbits 1 1 0 1: 14 Tbits 1 1 1 0: 15 Tbits 1 1 1 1: 16 Tbits
3 to 0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Set the RLN3nLWUP register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

WUTL[3:0] Bits (Wake-up Transmission Low Level Width Select)

These bits set the low-level width of the wake-up frame transmission.
1 Tbit to 16 Tbits can be set.

13.3.3.8 RLN3nLIE — LIN Interrupt Enable Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	SHIE	ERRIE	FRCIE	FTCIE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 13.37 RLN3nLIE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	SHIE	Successful Header Reception Interrupt Request Enable 0: Disables successful header reception interrupt request. 1: Enables successful header reception interrupt request.
2	ERRIE	Error Detection Interrupt Request Enable 0: Disables error detection interrupt request. 1: Enables error detection interrupt request.
1	FRCIE	Successful Response/Wake-up Reception Interrupt Request Enable 0: Disables successful Response/wake-up reception interrupt request. 1: Enables successful Response/wake-up reception interrupt request.
0	FTCIE	Successful Response/Wake-up Transmission Interrupt Request Enable 0: Disables successful Response/wake-up transmission interrupt request. 1: Enables successful Response/wake-up transmission interrupt request.

Set the RLN3nLIE register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

SHIE Bit (Successful Header Reception Interrupt Request Enable)

The SHIE bit enables or disables an interrupt request upon successful reception of a header.
With 0 set, the interrupt request for RLIN3n reception complete is not generated when the HTRC flag in the RLN3nLST register is set to 1.
With 1 set, the interrupt request for RLIN3n reception complete is generated when the HTRC flag in the RLN3nLST register is set to 1.

ERRIE Bit (Error Detection Interrupt Request Enable)

The ERRIE bit enables or disables an interrupt request upon detection of an error.
With 0 set, the interrupt request for RLIN3n status is not generated when the ERR flag in the RLN3nLST register is set to 1.
With 1 set, the interrupt request for RLIN3n status is generated when the ERR flag in the RLN3nLST register is set to 1.
Error types that are interrupt sources are the bit error, frame/response timeout error, framing error, sync filed error, ID parity error, checksum error, and response preparation error.
Detection of the bit error, frame/response timeout error, sync filed error, ID parity error, and framing error can be enabled or disabled using the RLN3nLEDE register.

FRCIE Bit (Successful Response/Wake-up Reception Interrupt Request Enable)

The FRCIE bit enables or disables an interrupt request upon successful reception of a response or a wake-up frame (counting of low level width of the input signal).
With 0 set, the interrupt request for successful RLIN3n reception is not generated when the FRC flag in the RLN3nLST register is set to 1.
With 1 set, the interrupt request for successful RLIN3n reception is generated when the FRC flag in the RLN3nLST register is set to 1.

FTCIE Bit (Successful Response/Wake-up Transmission Interrupt Request Enable)

The FTCIE bit enables or disables an interrupt request upon successful transmission of a response or a wake-up frame.
With 0 set, the interrupt request for successful RLIN3n transmission is not generated when the FTC flag in the RLN3nLST register is set to 1.
With 1 set, the interrupt request for successful RLIN3n transmission is generated when the FTC flag in the RLN3nLST register is set to 1.

13.3.3.9 RLN3nLEDE — LIN Error Detection Enable Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0D_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LTES	IPERE	—	SFERE	FERE	TERE	—	BERE
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W

Table 13.38 RLN3nLEDE Register Contents

Bit Position	Bit Name	Function
7	LTES	Timeout Error Select 0: Frame timeout error 1: Response timeout error
6	IPERE	ID Parity Error Detection Enable 0: Disables ID Parity error detection. 1: Enables ID Parity error detection.
5	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
4	SFERE	Sync Field Error Detection Enable 0: Disables Sync Field error detection. 1: Enables Sync Field error detection.
3	FERE	Framing Error Detection Enable*1 0: Disables framing error detection. 1: Enables framing error detection.
2	TERE	Timeout Error Detection Enable 0: Disables frame/response timeout error detection. 1: Enables frame/response timeout error detection.
1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	BERE	Bit Error Detection Enable*1 0: Disables bit error detection. 1: Enables bit error detection.

Note 1. Bits FERE and BERE must be set to 1.

Set the RLN3nLEDE register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

LTES Bit (Timeout Error Select)

The LTES bit selects the timeout function to be used.
With 0 set, the timeout function applies to frame timeout.
With 1 set, the timeout function applies to response timeout.
For details on the timeout error, see **Section 13.7.7, Error Status**.

IPERE Bit (ID Parity Error Detection Enable)

This bit enables or disables detection of the ID parity error.
With 0 set, the ID parity error is not detected.
With 1 set, the ID parity error is detected.
When this bit is set to 1, the detection result is reflected in the IPER flag in the RLN3nLEST register.
For details on the ID parity error, see **Section 13.7.7, Error Status**.

SFERE Bit (Sync Field Error Detection Enable)

This bit enables or disables detection of the sync field error.
With 0 set, the sync field error is not detected.
With 1 set, the sync field error is detected.
Regardless of the setting of this bit, when a sync field error is detected, this module waits for the next header.
When this bit is set to 1, the detection result is reflected in the SFER flag in the RLN3nLEST register.
For details on the sync filed error, see **Section 13.7.7, Error Status**.

FERE Bit (Framing Error Detection Enable)

The FERE bit enables or disables detection of the framing error.
With 0 set, the framing error is not detected.
With 1 set, the framing error is detected.
With this device, always set this bit to 1. The detection result is indicated in the FER flag in the RLN3nLEST register.
For details on the framing error, see **Section 13.7.7, Error Status**.

TERE Bit (Timeout Error Detection Enable)

The TERE bit enables or disables detection of the frame timeout error or the response timeout error.
With 0 set, the frame timeout error or response timeout error is not detected.
With 1 set, the frame timeout error or response timeout error is detected.
When this bit is set to 1, the detection result is indicated in the TER flag in the RLN3nLEST register.
With the LTES bit, either the frame timeout error or response timeout error can be selected.
The timeout error should not be used in LIN slave mode [auto baud rate] (when the LMD[1:0] bits in the RLN3nLMD register are “10_B”).
Do not use the timeout error if response data of 9 bytes or more is to be transmitted or received.
For details on the timeout error, see **Section 13.7.7, Error Status**.

BERE Bit (Bit Error Detection Enable)

The BERE bit enables or disables detection of the bit error.
With 0 set, the bit error is not detected.
With 1 set, the bit error is detected.
With this device, always set this bit to 1. The detection result is indicated in the BER flag in the RLN3nLEST register.
For details on the bit error, see **Section 13.7.7, Error Status**.

13.3.3.10 RLN3nLCUC — LIN Control Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0E_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OM1	OM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 13.39 RLN3nLCUC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	OM1	LIN Mode Select 0: LIN wake-up mode is caused. 1: LIN operation mode is caused.
0	OM0	LIN Reset 0: LIN reset mode is caused. 1: LIN reset mode is canceled.

Set the RLN3nLCUC register to 01_H to cause a transition to LIN wake-up mode after canceling LIN reset mode, and set the register to 03_H to cause a transition to LIN operation mode.

In LIN self-test mode, set the RLN3nLCUC register to 03_H after a transition to LIN self-test mode is completed.

After a value is written to this register, confirm that the value written is actually indicated in the RLN3nLMST register before writing another value.

OM1 Bit (LIN Mode Select)

The OM1 bit selects the specific LIN operation mode (either LIN wake-up mode or LIN operation mode) after canceling LIN reset mode.
With 0 set, LIN wake-up mode is caused.
With 1 set, LIN operation mode is caused.
This bit is valid only when the OMM0 bit in the RLN3nLMST register is 1.
Writing a value to this bit is disabled while the FTS bit in the RLN3nLTRC register is 1.

OM0 Bit (LIN Reset)

The OM0 bit selects either causing a transition to LIN reset mode or canceling LIN reset mode.
With 0 set, LIN reset mode is caused.
With 1 set, LIN reset mode is canceled.

13.3.3.11 RLN3nLTRC — LIN Transmission Control Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	LNRR	RTS	FTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 13.40 RLN3nLTRC Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	LNRR	No LIN Response Request 0: Response for the reception ID 1: No response for the reception ID
1	RTS	Response Transmission/Reception Start 0: Response transmission/reception is stopped. 1: Response transmission/reception is started.
0	FTS	LIN Communication Start 0: Header reception/wake-up transmission/reception is stopped. 1: Header reception/wake-up transmission reception is started.

LNRR Bit (No LIN Response Request)

After receiving the header and checking the received ID, set this bit to 1 if no response is transmitted/received.
Once set, this bit is automatically cleared to 0 upon detection of new sync field or transition to LIN reset mode.
Only 1 can be written to this bit; 0 cannot be written.
To write 1 to this bit, write 04_H using the store instruction.
Do not set this bit and the RTS bit to 1 simultaneously.
Writing a value to this bit is disabled when the OMM0 bit of the RLN3nLMST register is 0_B (LIN reset mode).
Writing a value to this bit is disabled when the FTS bit is 0 (header reception or wake-up transmission/reception is halted).
When response data of 9 bytes or more is to be transmitted or received, use this bit only for the completion of the header. (Do not use this bit for the second or latter data group.)

RTS Bit (Response Transmission/Reception Start)

After receiving the header and checking the received ID, set this bit to 1 at the response transmission or at the start of response reception.
Once set, this bit is automatically cleared to 0 upon completion of response transmission or reception (including error detection) or transition to LIN reset mode.
Only 1 can be written to this bit; 0 cannot be written.
To write 1 to this bit, write 02_H to the RLN3nLTRC register using the store instruction.
Do not set this bit and the LNRR bit to 1 simultaneously
Writing a value to this bit is disabled when the OMM0 bit of the RLN3nLMST register is 0_B (LIN reset mode).
Writing a value to this bit is disabled when the FTS bit is 0 (header reception or wake-up transmission/reception is halted).

When response data of 9 bytes or more is to be transmitted or received, set this bit to 1 each time a data group (variable from 0 to 8 bytes) is transmitted or received. Once set, this bit is automatically cleared to 0 upon completion of data group transmission/reception or transition to LIN reset mode.

FTS Bit (LIN Communication Start)

Set this bit to 1 to start header reception or wake-up transmission/reception.
Only 1 can be written to this bit; 0 cannot be written.
Writing a value to this bit is disabled when the OMM0 bit of the RLN3nLMST register is 0_B (LIN reset mode).
This bit is set to 0 upon completion of frame or wake-up communication and transition to LIN reset mode.

13.3.3.12 RLN3nLMST — LIN Mode Status Register

Access: This register can only be read in 8-bit units

Address: <RLIN3n_base> + 11_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OMM1	OMM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 13.41 RLN3nLMST Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned.
1	OMM1	LIN Mode Status Monitor 0: The module is in LIN wake-up mode. 1: The module is in LIN operation mode.
0	OMM0	LIN Reset Status Monitor 0: The module is in LIN reset mode. 1: The module is not in LIN reset mode.

OMM1 Bit (LIN Mode Status Monitor)

The OMM1 bit indicates the current operating mode.

OMM0 Bit (LIN Reset Status Monitor)

The OMM0 bit indicates the current operating mode.

13.3.3.13 RLN3nLST — LIN Status Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 12_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	HTRC	D1RC	—	—	ERR	—	FRC	FTC
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R/W	R/W

Table 13.42 RLN3nLST Register Contents

Bit Position	Bit Name	Function
7	HTRC	Successful Header Reception Flag 0: Header transmission has not been completed. 1: Header transmission has been completed.
6	D1RC	Successful Data 1 Reception Flag These bits are always read as 0. The write value should always be 0.
5, 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	ERR	Error Detection Flag 0: No error has been detected. 1: Error has been detected.
2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	FRC	Successful Response/Wake-up Reception Flag 0: Response or wake-up reception has not been completed. 1: Response or wake-up reception has been completed.
0	FTC	Successful Response/Wake-up Transmission Flag 0: Response or wake-up transmission has not been completed. 1: Response or wake-up transmission has been completed.

The RLN3nLST register is automatically cleared to 00_H upon transition to LIN reset mode.

In LIN reset mode, writing a value to this register is disabled. In LIN reset mode, the register retains 00_H.

To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

HTRC Flag (Successful Header Reception Flag)

Only 0 can be written to the HTRC flag; when 1 is written, the bit retains the value before 1 is written. The HTRC flag is set to 1 upon completion of header reception. Here, an interrupt request for successful RLIN3n reception is generated if the SHIE bit in the RLN3nLIE register is 1 (interrupt is enabled). However, if header reception is completed while this bit is 1, an interrupt is not generated. To clear this bit to 0, write 0 to the bit.
To detect a new header in the response field upon completion of header reception, clear this bit after it is set to 1.

D1RC Flag (Successful Data 1 Reception Flag)

Only 0 can be written to the D1RC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The D1RC flag is set to 1 upon completion of data 1 reception. Here, an interrupt request is not generated. Write 0 to clear this bit.
When response data of 9 bytes or more is to be received, this bit is set to 1 each time data 1 of a data group (variable from 0 to 8 bytes) is received. Write 0 before starting reception of the next data group.

ERR Flag (Error Detection Flag)

The ERR flag is set to 1 upon detection of an error (when the value of at least one of the flags of the RLN3nLEST register is set to 1). Here, an interrupt request for RLIN3n status is generated if the ERRIE bit in the RLN3nLIE register is 1 (interrupt is enabled). However, if an error is detected while this bit is 1, an interrupt is not generated. To clear the bit to 0, write 0 to the RPER, IPER, CSER, SFER, FER, TER, and BER flags in the RLN3nLEST register.

FRC Flag (Successful Response/Wake-up Reception Flag)

Only 0 can be written to the FRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FRC flag is set to 1 upon completion of response or wake-up reception. Here, an interrupt request for successful RLIN3n reception is generated if the FRCIE bit in the RLN3nLIE register is 1 (interrupt is enabled). However, if response reception or wake-up reception is completed while this bit is 1, an interrupt is not generated. Write 0 to clear this bit.
When response data of 9 bytes or more is to be received, this bit is set to 1 each time a data group (variable from 0 to 8 bytes) is received. Write 0 before starting reception of the next data group.

FTC Flag (Successful Response/Wake-up Transmission Flag)

Only 0 can be written to the FTC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FTC flag is set to 1 upon completion of response or wake-up transmission. Here, an interrupt request for RLIN3n transmission is generated if the FTCIE bit in the RLN3nLIE register is 1 (interrupt is enabled). However, if response reception or wake-up reception is completed while this bit is 1, an interrupt is not generated. Write 0 to clear this bit.
When response data of 9 bytes or more is to be transmitted, this bit is set to 1 each time a data group (variable from 0 to 8 bytes) is transmitted. Write 0 before starting transmission of the next data group.

13.3.3.14 RLN3nLEST — LIN Error Status Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 13_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	RPER	IPER	CSER	SFER	FER	TER	—	BER
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W

Table 13.43 RLN3nLEST Register Contents

Bit Position	Bit Name	Function
7	RPER	Response Preparation Error Flag 0: Response preparation error has not been detected. 1: Response preparation error has been detected.
6	IPER	ID Parity Error Flag 0: ID parity error has not been detected. 1: ID parity error has been detected.
5	CSER	Checksum Error Flag 0: Checksum error has not been detected. 1: Checksum error has been detected.
4	SFER	Sync Field Error Flag 0: Sync field error has not been detected. 1: Sync field error has been detected.
3	FER	Framing Error Flag 0: Framing error has not been detected. 1: Framing error has been detected.
2	TER	Timeout Error Flag 0: Frame/response timeout error has not been detected. 1: Frame/response timeout error has been detected.
1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	BER	Bit Error Flag 0: Bit error has not been detected. 1: Bit error has been detected.

The RLN3nLEST register is automatically cleared to 00_H upon transition to LIN reset mode.

In LIN reset mode, this register cannot be written to. In LIN reset mode, the register retains 00_H.

To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

RPER Flag (Response Preparation Error Flag)

Only 0 can be written to the RPER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The RPER flag is set to 1 upon response preparation error detection. Write 0 to clear this bit.

IPER Flag (ID Parity Error Flag)

Only 0 can be written to this flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the IPERE bit of the RLN3nLEDE register is 1 (ID parity error detection enabled), this bit is set to 1 upon ID parity error detection. Write 0 to clear this bit.

CSER Flag (Checksum Error Flag)

Only 0 can be written to the CSER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The CSER flag is set to 1 upon checksum error detection. Write 0 to clear this bit.

SFER Flag (Sync Field Error Flag)

Only 0 can be written to this flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the SFERE bit of the RLN3nLEDE register is 1 (sync field error detection enabled), this bit is set to 1 upon sync field error detection. Write 0 to clear this bit.

FER Flag (Framing Error Flag)

Only 0 can be written to the FER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the FERE bit of the RLN3nLEDE register is 1 (framing error detection enabled), the FER flag is set to 1 upon framing error detection. Write 0 to clear this bit.

TER Flag (Timeout Error Flag)

Only 0 can be written to the TER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the TERE bit of the RLN3nLEDE register is 1 (frame/response timeout error detection enabled), this flag is set to 1 upon frame timeout error or response timeout error detection. Write 0 to clear this bit.

BER Flag (Bit Error Flag)

Only 0 can be written to the BER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the BERE bit of the RLN3nLEDE register is 1 (bit error detection enabled), the BER flag is set to 1 upon bit error detection. Write 0 to clear this bit.

13.3.3.15 RLN3nLDFC — LIN Data Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LSS	—	LCS	RCDS	RFDL[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.44 RLN3nLDFC Register Contents

Bit Position	Bit Name	Function
7	LSS	Transmission/Reception Continuation Select 0: The data group to be transmitted/received next is the last one. 1: The data group to be transmitted/received next is not the last one. (Data transmission/reception continues without waiting for reception of the next header.)
6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	LCS	Checksum Select 0: Classic checksum mode 1: Enhanced checksum mode
4	RCDS	Response Field Communication Direction Select 0: Reception 1: Transmission
3 to 0	RFDL[3:0]	Response Field Length Select b3 b0 0 0 0 0: 0 byte (+ checksum) 0 0 0 1: 1 byte (+ checksum) 0 0 1 0: 2 bytes (+ checksum) : 0 1 1 1: 7 bytes (+ checksum) 1 0 0 0: 8 bytes (+ checksum) Settings other than the above are prohibited.

LSS Bit (Transmission/Reception Continuation Select)

The LSS bit indicates that the data group to be transmitted or received next is not the last data group when response data of 9 bytes or more is to be transmitted or received.

With 0 set, data and checksum are transmitted or received because the next data group to be transmitted or received is the last one.

With 1 set, only data is transmitted or received, and the checksum is not included because the next data group to be transmitted or received is not the last one.

When multi-byte response transmission/reception function is not used, set it to “0”.

This should be set when the RTS bit is 0 (response transmission/reception stopped).

LCS Bit (Checksum Select)

The LCS bit sets the checksum mode.
With 0 set, classic checksum mode is selected.
With 1 set, enhanced checksum mode is selected.
When the timeout error is used (the TERE bit in the RLN3nLEDE register is 1), the specific timeout time depends on the setting of this bit. For details on the bit error, see **Section 13.7.7, Error Status**.
When the length of the response field data is 0 byte (the RFDL bit is 0), do not set this bit to "1" (enhanced).
When response data of 9 bytes or more is to be transmitted or received, do not change the LCS bit setting after the first data group through the last data group.
During transmission or reception of response data of 9 bytes or more, only the last data group (the LSS bit is 0) includes the checksum, and no other groups (the LSS bit is 1) include the checksum.
This should be set when the RTS bit is 0 (response transmission/reception stopped).

RCDS Bit (Response Field Communication Direction Select)

This bit selects the direction of the response field/wake-up signal communication.
With 0 set, reception is performed in the response field. In LIN wake-up mode, wake-up reception is performed (low-level width of the input signal is counted).
With 1 set, transmission is performed in the response field. In LIN wake-up mode, wake-up transmission is performed.
This bit should be set when the RTS bit in the RLN3nLTRC register is 0 in LIN operation mode (response transmission/reception stopped) or when the FTS bit is 0 in LIN wake-up mode (header reception or wake-up transmission/reception stopped).
When response data of 9 bytes or more is to be transmitted or received, do not change this bit setting after the first data group through the last data group.

RFDL[3:0] Bits (Response Field Length Select)

The RFDL bits set the length of the response field data.
The data length can be 0 to 8 bytes excluding the checksum size.
These bits should be set when the RTS bit in the RLN3nLTRC register is 0 (response transmission/reception stopped).
When response data of 9 bytes or more is to be transmitted or received, only the last data group (the LSS bit is 0) includes the checksum, and no other groups (the LSS bit in the RLN3nLDFC register is 1) include the checksum.

13.3.3.16 RLN3nLIDB — LIN ID Buffer Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 15_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	IDP1	IDP0	ID[5:0]					
Value after reset	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.45 RLN3nLIDB Register Contents

Bit Position	Bit Name	Function
7	IDP1	Parity Setting (P1) Stores the parity bits (P1) to be received in the ID field.
6	IDP0	Parity Setting (P0) Stores the parity bits (P0) to be received in the ID field.
5 to 0	ID[5:0]	ID Setting Stores the 6-bit ID value to be received in the ID field.

The value in the RLN3nLIDB register is enabled after the completion of header reception. In LIN mode (LIN operation mode, LIN wake-up mode), writing to this register is disabled.

In LIN self-test mode, the operation is as follows.

Write the value to be transmitted before communication. After completion of frame transmission/reception (after loopback), the reversed value of the received value can be read.

For details about the LIN self-test mode, see **Section 13.9, LIN Self-Test Mode**.

IDP[1:0] Bits (Parity Setting)

The IDP bits store the parity bits (P0 and P1) to be received in the ID field of the LIN frame. IDP0 is for P0 and IDP1 is for P1.
When the IPERE bit in the RLN3nLEDE register is 1 (ID parity detection enable), the received value and the value calculated internally are checked. If they do not match, IPER (ID parity error flag) is set.

ID[5:0] Bits (ID Setting)

The ID bits store the 6-bit ID value to be received in the ID field of the LIN frame.

13.3.3.17 RLN3nLCBR — LIN Checksum Buffer Register

Access: This register can only be read in 8-bit units.
However, in LIN self-test mode, this register can be read and written in 8-bit units.

Address: <RLIN3n_base> + 16_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CKSM[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.46 RLN3nLCBR Register Contents

Bit Position	Bit Name	Function
7 to 0	CKSM[7:0]	Holds the checksum value transmitted or received.

In LIN mode, this register operates as follows:

- When the RCDS bit in the RLN3nLDFC register is 1 (transmission):
  The value transmitted can be read from the register. Read the value after transmission is completed.
  Writing to this register is invalid.
- When the RCDS bit in the RLN3nLDFC register is 0 (reception):
  The value received can be read from the register. Read the value after reception is completed. Writing to this register is invalid.

In LIN self-test mode, this register operates as follows:

- When the RCDS bit in the RLN3nLDFC register is 1 (transmission):
  After completion of the frame transmission (after loopback), the reversed value of the received value can be read.
- When the RCDS bit in the RLN3nLDFC register is 0 (reception):
  Write the value to be received before communication. After completion of frame transmission/reception (after loopback), the reversed value of the received value can be read.

For details about the LIN self-test mode, see **Section 13.9, LIN Self-Test Mode.**

Set the RLN3nLCBR register when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

When response data of at least 9 bytes is to be transmitted or received, the checksum is only appended to the last data group, so this register is not updated for the other data groups.

13.3.3.18 RLN3nLDBRb — LIN Data Buffer b Register (b = 1 to 8)

Access: This register can be read/written in 8-bit units.

Address: RLN3nLDBR1: <RLIN3n_base> + 18_H
RLN3nLDBR2: <RLIN3n_base> + 19_H
RLN3nLDBR3: <RLIN3n_base> + $1A_H$
RLN3nLDBR4: <RLIN3n_base> + $1B_H$
RLN3nLDBR5: <RLIN3n_base> + $1C_H$
RLN3nLDBR6: <RLIN3n_base> + $1D_H$
RLN3nLDBR7: <RLIN3n_base> + $1E_H$
RLN3nLDBR8: <RLIN3n_base> + $1F_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LDB[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.47 RLN3nLDBRb Register Contents

Bit Position	Bit Name	Function
7 to 0	LDB[7:0]	Sets the data for transmission or holds the received data. Setting range: 00_H to FF_H

- For response transmission:
  The RLN3nLDBRb registers set the data to be transmitted in the response field.
  These registers should be set when the RTS bit in the RLN3nLTRC register is 0 (response transmission/reception is halted).
- For response reception:
  The RLN3nLDBRb registers hold the data received in the response field.
  Any previously received data are overwritten. The data including bytes where an error was found are stored in these registers in response to the detection of an error.
  Do not read these registers when the RTS bit is 1 (response transmission/reception is started)

In LIN self-test mode, the operation is as follows.

Write the value to be transmitted before communication. After completion of frame transmission/reception (after loopback), the reversed value of the received value can be read.

For details about the LIN self-test mode, see **Section 13.9, LIN Self-Test Mode**.

13.3.4 UART Related Registers

13.3.4.1 RLN3nLWBR — LIN Wake-Up Baud Rate Select Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 01_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NSPB[3:0]				LPRS[2:0]			—
Value after reset	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 13.48 RLN3nLWBR Register Contents

Bit Position	Bit Name	Function
7 to 4	NSPB[3:0]	Bit Sampling Count Select b7 b4 0 0 0 0: 16 sampling 0 1 0 1: 6 sampling 0 1 1 0: 7 sampling 0 1 1 1: 8 sampling 1 0 0 0: 9 sampling 1 0 0 1: 10 sampling 1 0 1 0: 11 sampling 1 0 1 1: 12 sampling 1 1 0 0: 13 sampling 1 1 0 1: 14 sampling 1 1 1 0: 15 sampling 1 1 1 1: 16 sampling Settings other than the above are prohibited.
3 to 1	LPRS[2:0]	Prescaler Clock Select b3 b1 0 0 0: 1/1 0 0 1: 1/2 0 1 0: 1/4 0 1 1: 1/8 1 0 0: 1/16 1 0 1: 1/32 1 1 0: 1/64 1 1 1: 1/128
0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Set the LN3nLWBR register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

NSPB[3:0] Bits (Bit Sampling Count Select)

The NSPB bits select the number of sampling in one Tbit (reciprocal of the bit rate).
In UART mode, it is possible to set the NSPB bits from 6 sampling to 16 sampling.

LPRS[2:0] Bits (Prescaler Clock Select)

The LPRS bits select the frequency division ratio for the prescaler.
The LIN communication clock source is divided by this prescaler.

13.3.4.2 RLN3nLBRP01 — UART Baud Rate Prescaler 01 Register

Access: RLN3nLBRP01 register can be read/written in 16-bit units.
RLN3nLBRP0 register can be read/written in 8-bit units.
RLN3nLBRP1 register can be read/written in 8-bit units.

Address: RLN3nLBRP01: <RLIN3n_base> + 02_H
RLN3nLBRP0: <RLIN3n_base> + 02_H
RLN3nLBRP1: <RLIN3n_base> + 03_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BRP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.49 RLN3nLBRP01 Register Contents

Bit Position	Bit Name	Function
15 to 0	BRP[15:0]	Assuming that the value set in this register is L (0 to 65535), the baud rate prescaler divides the frequency of the prescaler clock by L + 1. Setting range: 0000_H to $FFFF_H$

Set the RLN3nLBRP01 register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

Assuming that the value set in this register is L, the baud rate prescaler divides the frequency of the clock that is selected by the LPRS bits (prescaler clock select bits) in the RLN3nLWBR register by L + 1.

The RLN3nLBRP01 register can be accessed in 8-bit units by the registers RLN3nLBRP0 and RLN3nLBRP1.

13.3.4.3 RLN3nLMD — UART Mode Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	LRDNFS	—	—	—	LMD[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R/W	R/W

Table 13.50 RLN3nLMD Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	LRDNFS	UART Reception Data Noise Filtering Disable 0: The noise filter is enabled. 1: The noise filter is disabled.
4 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1, 0	LMD[1:0]	LIN/UART Mode Select b1 b0 0 1: UART mode

Set the RLN3nLMD register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

LRDNFS Bit (UART Reception Data Noise Filtering Disable)

The LRDNFS bit enables or disables the noise filter when receiving data.
With 0 set, the noise filter is enabled when receiving data.
With 1 set, the noise filter is disabled when receiving data.

LMD[1:0] Bits (LIN/UART Mode Select)

The LMD bits select the LIN/UART interface mode.
To use the LIN/UART interface as an UART, set these bits to 01_B.

13.3.4.4 RLN3nLBFC — UART Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 09_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	UTPS	URPS	UPS[1:0]		USBLS	UBOS	UBLS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.51 RLN3nLBFC Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
6	UTPS	UART Output Polarity Switch 0: Transmit data normal output 1: Transmit data with inversion output
5	URPS	UART Input Polarity Switch 0: Reception data normal output 1: Reception data with inversion output
4, 3	UPS[1:0]	UART Parity Select 00: Parity prohibited 01: Even Parity 10: 0 Parity 11: Odd parity
2	USBLS	UART Stop Bit length Select 0: Stop bit:1 bit 1: Stop bit: 2 bits
1	UBOS	UART Transfer Format Order Select 0: LSB First 1: MSB First
0	UBLS	UART Character Length Select 0: UART 8 bits communication 1: UART 7 bits communication

Set the RLN3nLBFC register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

UTPS Bit (UART Output Polarity Switch)

Sets the output polarity for UART communication.
With 0 set, transmit data is output without inversion.
With 1 set, receive data is output with inversion.
The setting of this bit is valid in all the bits of the UART frame.
In half-duplex communication, this setting should match with the setting of URPS bit.

URPS Bit (UART Input Polarity Switch)

Sets the input polarity for UART communication.
With 0 set, receive data is input without inversion.
With 1 set, receive data is input with inversion.
The setting of this bit is valid in all the bits of the UART frame.
In half-duplex communication, this setting should match with the setting of UTPS bit.
When setting this bit to “1” and expansion bit reception ((with expansion bit comparison) or (with data comparison)) is performed, set the inverse of the expected value to the UEBDL bit in the RLN3nLUOR1 register and RLN3nLIDB register for comparison of the inverted values of the received values.

UPS[1:0] Bits (UART Parity Select)

Sets the UART parity.

- When these bits are set to 00_B, data is communicated without the parity.

  [Transmission]
  A parity bit is not added to transmit data.

  [Reception]
  Data is received without parity processing. Therefore, a parity error does not occur.

- When these bits are set to 01_B, data is communicated with the even parity.

  [Transmission]
  If the number of 1s in transmit data is odd, “1” is added to the parity bit. If the number of 1s in transmit data is even, “0” is added to the parity bit.

  [Reception]
  If the number of 1s in receive data including the parity bit is odd, a parity error occurs.

- When these bits are set to 10_B, data is communicated with 0 parity.

  [Transmission]
  Regardless of the number 1s in transmit data, “0” is added to the parity bit.

  [Reception]
  The value of the parity bit is not judged. Therefore, no parity error occurs.

- When these bits are set to 11_B, data is communicated with the odd parity.

  [Transmission]
  If the number of 1s in transmit data is odd, “0” is added to the parity bit. If the number of 1s in transmit data is even, “1” is added to the parity bit.

  [Reception]
  If the number of 1s in receive data including the parity bit is even, a parity error occurs.

USBLS Bit (UART Stop Bit Length Select)

Sets the stop bit length of data for UART communication.
With 0 set, stop bit length of 1 bit is selected.
With 1 set, stop bit length of 2 bits is selected.

UBOS Bit (UART Transfer Format Select)

Sets the bit order of data for UART communication.
With 0 set, LSB first is selected.
With 1 set, MSB first is selected.

UBLS Bit (UART Character Length Select)

Sets the character length of one frame for UART communication.
With 0 set, the character length is 8 bits.
With 1 set, the character length is 7 bits.
When the character length of one frame is 9 bits (the UEBE bit in the RLN3nLUOR1 register is 1), the setting of this bit is ignored.

13.3.4.5 RLN3nLSC — UART Space Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0A_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	IBS[1:0]		—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R

Table 13.52 RLN3nLSC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5, 4	IBS[1:0]	Inter-Byte Space Select b5 b4 0 0: 0 Tbit 0 1: 1 Tbit 1 0: 2 Tbits 1 1: 3 Tbits
3 to 0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Set the RLN3nLSC register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

IBS[1:0] Bits (Inter-Byte Space Select)

The IBS bits set the width of the space between the UART frame in UART buffer transmit
0 Tbit to 3 Tbits can be set.

When the UART buffer is not in use, set the IBS[1:0] bits to 00_B.

The setting of these bits has no effect when data is transferred from the UART transmit data register (RLN3nLUTDR) and UART wait transmit data register (RLN3nLUWTDR). These bits must be set to 00_B.

13.3.4.6 RLN3nLEDE — UART Error Detection Enable Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0D_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	FERE	OERE	—	BERE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R/W

Table 13.53 RLN3nLEDE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	FERE	Framing Error Detection Enable 0: Disables framing error detection. 1: Enables framing error detection.
2	OERE	Overrun Error Detection Enable 0: Disables overrun error detection. 1: Enables overrun error detection.
1	Reserved	When read, the value after reset is returned. When writing to this bit, write the value after reset.
0	BERE	Bit Error Detection Enable 0: Disables bit error detection. 1: Enables bit error detection.

Set the RLN3nLEDE register when the OMM0 bit in the RLN3nLMST register is 0_B (in LIN reset mode).

FERE Bit (Framing Error Detection Enable)

The FERE bit enables or disables detection of the framing error.
With 0 set, the framing error is not detected.
With 1 set, the framing error is detected.
When this bit is set to 1, the detection result is indicated in the FER flag in the RLN3nLEST register.
For details on the framing error, see **Section 13.8.5, Error Status**.

OERE Bit (Overrun Error Detection Enable)

This bit enables or disables detection of the overrun error.
With 0 set, the overrun error is not detected.
With 1 set, the overrun error is detected.
When this bit is set to 1, the detection result is reflected in the OER flag in the RLN3nLEST register.
For details on the overrun error, see **Section 13.8.5, Error Status**.

BERE Bit (Bit Error Detection Enable)

The BERE bit enables or disables detection of the bit error.
With 0 set, the bit error is not detected.
With 1 set, the bit error is detected.
When this bit is set to 1, the detection result is indicated in the BER flag in the RLN3nLEST register.
In full-duplex communication, do not set this bit to "1".
Do not set this register when the NSPB bits in the RLN3nLWBR register is 0101_B (6 sampling) and the LRDNFS bit in the RLN3nLMD register is 0 (noise filter is enabled).
For details on the bit error, see **Section 13.8.5, Error Status**.

13.3.4.7 RLN3nLCUC — UART Control Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + $0E_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 13.54 RLN3nLCUC Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	OM0	LIN Reset 0: LIN reset mode is caused. 1: LIN reset mode is canceled.

After a value is written to this register, confirm that the value written is actually indicated in the RLN3nLMST register before writing another value.

OM0 Bit (LIN Reset)

The OM0 bit selects either causing a transition to reset mode or canceling reset mode.
With 0 set, LIN reset mode is caused.
With 1 set, LIN reset mode is canceled.

13.3.4.8 RLN3nLTRC — UART Transmission Control Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RTS	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R

Table 13.55 RLN3nLTRC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	RTS	UART Buffer Transmission Start 0: UART Buffer transmission is stopped. 1: UART Buffer transmission is started.
0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

RTS Bit (UART Buffer Transmission Start)

When transmitting data from the UART buffer, set this bit to “1”.
Only 1 can be written to this bit; 0 cannot be written.
Write to this bit when the UTOE bit in the RLN3nLUOER register is 1 (transmission enable) and the UTS bit in the RLN3nLST register is 0 (transmission is not in progress).
Once set, regardless of errors, this bit is automatically cleared to 0 upon completion of the number of data transmission specified by the MDL bit in the RLN3nLDFC register. Moreover, this bit is automatically cleared to 0 upon transition to LIN reset mode.
Writing a value to this bit is disabled when the OMM0 bit of the RLN3nLMST register is 0_B (in LIN reset mode).
When writing 1 to this bit while the UTSW bit in the RLN3nLRFC register is 1 (when UART buffer transmission is requested, the start of transmission is delayed until the stop bit of reception data is completed), write only during the reception of stop bit.

13.3.4.9 RLN3nLMST — UART Mode Status Register

Access: This register can only be read in 8-bit units

Address: <RLIN3n_base> + 11_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OMM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 13.56 RLN3nLMST Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is returned.
0	OMM0	LIN Reset Status Monitor 0: The module is in LIN reset mode. 1: The module is not in LIN reset mode.

OMM0 Bit (LIN Reset Status Monitor)

The OMM0 bit indicates the current operating mode.

13.3.4.10 RLN3nLST — UART Status Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 12_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	URS	UTS	ERR	—	—	FTC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 13.57 RLN3nLST Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	URS	Reception Status Flag 0:Reception is not operated. 1:Reception is operated.
4	UTS	Transmission Status Flag 0: Transmission is not operated. 1: Transmission is operated.
3	ERR	Error Detection Flag 0: No error has been detected. 1: Error has been detected.
2, 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	FTC	Successful UART Buffer Transmission Flag 0: UART buffer transmission has not been completed. 1: UART buffer transmission has been completed.

The RLN3nLST register is automatically cleared to "00_H" upon transition to LIN reset mode. In LIN reset mode, this register cannot be written to. In LIN reset mode, the register retains "00_H". To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

URS Flag (Reception Status Flag)

At the start of the reception, this flag is set to 1. The reception is started under the following condition.

- When the start bit is detected

At the end of reception, this flag is cleared to 0. While reception is stopped, this flag retains 0.
The reception is ended under the following conditions.

- Sampling point of the first bit of the stop bits

UTS Flag (Transmission Status Flag)

At the start of the transmission, this flag is set to 1. During the transmission, this flag retains 1.
The transmission is started under the following conditions.

- When transmission data is set to the RLN3nLUTDR or RLN3nLUWTDR register
- When the RTS bit in the RLN3nLTRC register is set to 1

This flag is cleared to 0 at the end of transmission.
The transmission is ended under the following conditions.

- When transmission of data set in the RLN3nLUTDR or RLN3nLUWTDR register is completed and next data is not set
- When transmission from UART buffer is completed (when the RTS bit in the RLN3nLTRC register is cleared to 0)

ERR Flag (Error Detection Flag)

This flag is set to 1 upon detection of an error, detection of an expansion bit, or upon ID matching (when the value of at least one of the flags of the RLN3nLEST register is 1). At this time, an interrupt request for RLIN3n status is generated. However, when this bit is 1, an interrupt is not generated upon detection of an error, detection of an expansion bit, or upon ID matching. To clear the bit to 0, write 0 to the UPER, IDMT, EXBT, FER, OER, and BER flags in the RLN3nLEST register.

FTC Flag (Successful UART Buffer Transmission Flag)

Only 0 can be written to the FTC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
Regardless of errors, this bit is set to 1 upon completion of the number of data transmission specified by the MDL bit in the RLN3nLDFC register from the UART buffer. At this time, an interrupt request for RLIN3n transmission is generated. Write 0 to the bit to be cleared.

13.3.4.11 RLN3nLEST — UART Error Status Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 13_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	UPER	IDMT	EXBT	FER	OER	—	BER
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R/W

Table 13.58 RLN3nLEST Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
6	UPER	Parity Error Flag 0: Parity error has not been detected. 1: Parity error has been detected.
5	IDMT	ID Matching Flag 0: The receive data does not match with the ID value. 1: The receive data matches with the ID value.
4	EXBT	Expanded Bit Detection Flag 0: Expanded bit has not been detected. 1: Expanded bit has been detected.
3	FER	Framing Error Flag 0: Framing error has not been detected. 1: Framing error has been detected.
2	OER	Overrun Error Flag 0: Overrun error has not been detected. 1: Overrun error has been detected.
1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	BER	Bit Error Flag 0: Bit error has not been detected. 1: Bit error has been detected.

The RLN3nLEST register is automatically cleared to 00_H when the module transitions to LIN reset mode. In LIN reset mode, this register cannot be written to, and the value of 00_H is held. To clear certain bits in this register, write 0 to those bits, and write 1 to the bits not to be cleared by using the store instruction.

UPER Flag (Parity Error Flag)

Only 0 can be written to this flag; when 1 is written, the bit retains the value that has been retained before 1 is written. This flag is set to 1 upon parity error detection. To clear the bit, write 0 to the bit.

IDMT Flag (ID Matching Flag)

Only 0 can be written to this flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The IDMT flag becomes 1 when all the following conditions are met:

- The UEBE bit in the RLN3nLUOR1 register is 1 (expansion bit enabled)
- The UECD bit in the RLN3nLUOR1 register is 0 (expansion bit comparison enabled)
- The UEBDCE bit in the RLN3nLUOR1 register is 1 (expansion bit/data comparison enabled)
  - The received expansion bit and the value of the UEBDL bit of the RLN3nLUOR1 register are matched.
  - The 8-bit receive data excluding the expansion bit and the value of the RLN3nLIDB register are matched.

To clear the bit, write 0 to the bit.

EXBT Flag (Expanded Bit Detection Flag)

Only 0 can be written to this flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
When the UEBE bit in the RLN3nLUOR1 register is 1 (expansion bit enable), if the received expansion bit matches with the UEBDL bit in the RLN3nLUOR1 register, this flag is set to 1.
To clear the bit, write 0 to the bit.

FER Flag (Framing Error Flag)

Only 0 can be written to this flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FER flag is set to 1 upon framing error detection while the FERE bit of the RLN3nLEDE register is 1 (framing error detection enabled).
To clear the bit, write 0 to the bit.

OER Flag (Overrun Error Flag)

Only 0 can be written to this flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The 0ER flag is set to 1 upon overrun error detection while the OERE bit of the RLN3nLEDE register is 1 (overrun error detection enabled).
To clear the bit, write 0 to the bit.

BER Flag (Bit Error Flag)

Only 0 can be written to the BER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The BER flag is set to 1 when the transmitted data and the data monitored by the receive pin do not match while the BERE bit of the RLN3nLEDE register is 1 (bit error detection enabled).
To clear the bit, write 0 to the bit.

13.3.4.12 RLN3nLDFC — UART Data Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	UTSW	—	MDL[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Table 13.59 RLN3nLDFC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	UTSW	Transmission Start Wait 0: When UART transmission is requested, transmission is started immediately. 1: When UART transmission is requested, transmission is not started until reception of the stop bit is completed.
4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3 to 0	MDL[3:0]	UART Buffer Data Length Select b3 b0 0 0 0 0: 9 data 0 0 0 1: 1 data 0 0 1 0: 2 data 0 0 1 1: 3 data 0 1 0 0: 4 data 0 1 0 1: 5 data 0 1 1 0: 6 data 0 1 1 1: 7 data 1 0 0 0: 8 data 1 0 0 1: 9 data Settings other than the above are prohibited.

UTSW Bit (Transmission Start Wait)

This bit controls the transmission start timing of UART buffer.
With 0 set, transmission is started as soon as the start of UART buffer transmit is requested.
With 1 set, transmission is started after the completion of the stop bit reception.
Note that the wait time is only 1 bit even if the stop bit length is set to 2 bits with the USBLS bit in the RLN3nLBFC register.
This bit is enabled when the RTS bit in the RLN3nLTRC register is set to 1. In addition, writing a value to this bit is disabled when the RTS bit is 1 (UART buffer transmission started).
Set this bit to 1 only to switch from reception to transmission in half-duplex communication.

MDL[3:0] Bits (UART Buffer Data Length Select)

This bit specifies the data length of the UART buffer.
Writing a value to these bits is disabled when the RTS bit in the RLN3nLTRC register is 1 (UART buffer transmission started).

13.3.4.13 RLN3nLIDB — UART ID Buffer Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 15_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	ID[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.60 RLN3nLIDB Register Contents

Bit Position	Bit Name	Function
7 to 0	ID[7:0]	ID value that is referred for the expansion bit data comparison is set

ID Bit (ID Bit)

When the UEBE bit in the RLN3nLUOR1 register is set to 1 (expansion bit enabled), the UECD bit is set to 0 (expansion bit comparison enabled), and the UEBDCE bit is set to 1 (expansion bit/data comparison enabled), set the receive data and the value to be compared. Write to the RLN3nLIDB register when the URS bit in the RLN3nLST register is 0 (receive operation is not in progress).

13.3.4.14 RLN3nLUDB0 — UART Data 0 Buffer Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 17_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	UDB[7:0]							
Value after reset	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.61 RLN3nLUDB0 Register Contents

Bit Position	Bit Name	Function
7 to 0	UDB[7:0]	Transmission data is set Setting range: 00_H to FF_H

If the data length selection corresponds to 9 data bytes (RLN3nLDFC.MDL bit is "0_H" or "9_H") for multi-byte UART transmission, then the first data value for UART communication is present in this buffer.

Write to the RLN3nLUDB0 register when the RTS bit of the RLN3nLTRC register is 0 (UART buffer transmission stopped).

Table 13.62, Bit Arrangement of the RLN3nLUDB0 Register According to Each Communication Format, shows the bit arrangement according to the set communication format.

For details about the UART buffer, see Section **13.8.1.2, UART Buffer Transmission, (1) UART Buffer Transmission**.

Table 13.62 Bit Arrangement of the RLN3nLUDB0 Register According to Each Communication Format

	RLN3nLUDB0							
	b7	b6	b5	b4	b3	b2	b1	b0
7-bit; LSB first	—	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
7-bit; MSB first	—	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6
8-bit; LSB first	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
8-bit; MSB first	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7

13.3.4.15 RLN3nLDBRb — UART Data Buffer b Register (b = 1 to 8)

Access: This register can be read/written in 8-bit units.

Address: RLN3nLDBR1: <RLIN3n_base> + 18_H
RLN3nLDBR2: <RLIN3n_base> + 19_H
RLN3nLDBR3: <RLIN3n_base> + $1A_H$
RLN3nLDBR4: <RLIN3n_base> + $1B_H$
RLN3nLDBR5: <RLIN3n_base> + $1C_H$
RLN3nLDBR6: <RLIN3n_base> + $1D_H$
RLN3nLDBR7: <RLIN3n_base> + $1E_H$
RLN3nLDBR8: <RLIN3n_base> + $1F_H$

Value after reset: 00_H

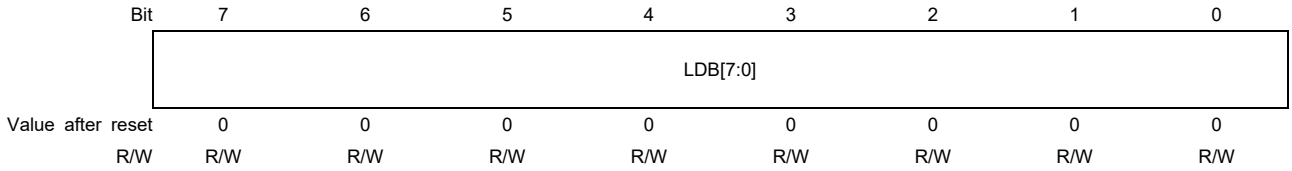


Table 13.63 RLN3nLDBRb Register Contents

Bit Position	Bit Name	Function
7 to 0	LDB[7:0]	Sets the data to be transmitted. Setting range: 00_H to FF_H

This register specifies the data transmitted from the UART buffer.

Write to these registers when the RTS bit of the RLN3nLTRC register is 0 (UART buffer transmission stopped).

Table 13.64, Bit Arrangement of the RLN3nLDBRb Register According to Each Communication Format, shows the bit arrangement according to the set communication format.

For details about the UART buffer, see **Section 13.8.1.2, UART Buffer Transmission, (1) UART Buffer Transmission**.

Table 13.64 Bit Arrangement of the RLN3nLDBRb Register According to Each Communication Format

	RLN3nLDBRb							
	b7	b6	b5	b4	b3	b2	b1	b0
7-bit; LSB first	—	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
7-bit; MSB first	—	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6
8-bit; LSB first	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
8-bit; MSB first	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7

13.3.4.16 RLN3nLUOER — UART Operation Enable Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	UROE	UTOE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 13.65 RLN3nLUOER Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	UROE	Reception Enable 0: Disables reception. 1: Enables reception.
0	UTOE	Transmission Enable 0: Disables transmission. 1: Enables transmission.

The RLN3nLUOER register is automatically cleared to 00_H upon transition to LIN reset mode. In LIN reset mode, this register cannot be written to.

In LIN reset mode, the register retains 00_H.

UROE Bit (Reception Enable)

The UROE bit enables or disables reception.
With 0 set, reception is disabled.
With 1 set, reception is enabled.
Do not clear this bit during reception. If the communication is suspended during reception, set the OM0 bit in the RLN3nLCUC register to 0 (LIN reset mode) to shift to the LIN reset mode. However, the transmit operation is also suspended at this time.
Do not set this bit to 1 when data transmission from the UART buffer is in progress.

UTOE Bit (Transmission Enable)

The UTOE bit enables or disables transmission.
With 0 set, transmission is disabled.
With 1 set, transmission is enabled.
Do not clear this bit during transmission. If the communication is suspended during transmission, set the OM0 bit in the RLN3nLCUC register to 0 (LIN reset mode) to shift to the LIN reset mode. However, the receive operation is also suspended at this time.

13.3.4.17 RLN3nLUOR1 — UART Option Register 1

Access: This register can be read/written in 8-bit units.

Address: <RLIN3n_base> + 21_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	UECD	UTIGTS	UEBDCE	UEBDL	UEBE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 13.66 RLN3nLUOR1 Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
4	UECD	Expansion Bit Comparison Disable 0: Enables comparison between the received expansion bit and the UEBDL bit value. 1: Disables comparison between the received expansion bit and the UEBDL bit value.
3	UTIGTS	Transmission Interrupt Generation Timing Select 0: Transmission interrupt is generated at the start of transmission. 1: Transmission interrupt is generated at the completion of transmission.
2	UEBDCE	Expansion Bit Data Comparison Enable 0: Disables data comparison after an expansion bit is detected. 1: Enables data comparison after an expansion bit is detected.
1	UEBDL	Expansion Bit Detection Level Select 0: Selects expansion bit value 0 as the expansion bit detection level. 1: Selects expansion bit value 1 as the expansion bit detection level.
0	UEBE	Expansion Bit Enable 0: Disables expansion bit operation. 1: Enables expansion bit operation.

UECD Bit (Expansion Bit Comparison Disable)

The UECD bit enables or disables comparison between the received expansion bit and the UEBDL bit value when the UEBE bit is 1 (expansion bit operation is enabled).

With 0 set, comparison between the received expansion bit and the UEBDL bit value is enabled when the expansion bit is received.

With 1 set, comparison between the received expansion bit and the UEBDL bit value is disabled when the expansion bit is received.

Set this bit when the OMM0 bit of the RLN3nLMST register is 0_B (in LIN reset mode).

Do not set this bit to 1 when the UART buffer is used.

Do not set this bit to 1 when the UEBDCE bit is 1 (expansion bit/data comparison enable).

UTIGTS Bit (Transmission Interrupt Generation Timing Select)

The UTIGTS bit sets the generation timing of the transmission interrupt.
With 0 set, the transmission interrupt is generated at the start of transmission.
With 1 set, the transmission interrupt is generated at the completion of transmission.
When transmission from the UART buffer is performed with 0 set, the transmission interrupt is generated only at the start of the transmission of the last data of the data length set with the MDL bits in the RLN3nLDFC register.
When transmission from the UART buffer is performed with 1 set, the transmission interrupt is generated only at the completion of the transmission of the last data of the data length set with the MDL bits in the RLN3nLDFC register.

UEBDCE Bit (Expansion Bit Data Comparison Enable)

After an expansion bit is detected, this bit enables or disables the comparison between the 8-bit receive data excluding the expansion bit and the value of the RLN3nLIDB register.
With 0 set, when the level selected by the UEBDL bit is detected as an expansion bit, the comparison between the receive value in the RLN3nLURDR register and the value of the RLN3nLIDB register is disabled.
With 1 set, when the level selected by the UEBDL bit is detected as an expansion bit, the comparison between the receive value in the RLN3nLURDR register and the value of the RLN3nLIDB register is enabled.
Set this bit when the OMM0 bit of the RLN3nLMST register is 0_B (in LIN reset mode).
Do not set this bit to 1 when the UEBE bit is 0 (expansion bit operation disabled).
Do not set this bit to 1 when the UECD bit is 1 (expansion bit comparison disabled).
Do not set this bit to 1 when the UART buffer is used.

UEBDL Bit (Expansion Bit Detection Level Select)

The UEBDL bit sets the level to be detected as the expansion bit when the UEBE bit is 1 (expansion bit operation is enabled) and the UECD bit is 0 (comparison of the expansion bit is enabled).
With 0 set, expansion bit value 0 is the level to be detected as the expansion bit.
With 1 set, expansion bit value 1 is the level to be detected as the expansion bit.
Set this bit when the OMM0 bit of the RLN3nLMST register is 0_B (in LIN reset mode).
Do not set this bit to 1 when the UART buffer is used.

UEBE Bit (Expansion Bit Enable Bit)

The UEBE bit enables or disables expansion bit operation.
With 0 set, expansion bit operation is disabled.
With 1 set, expansion bit operation is enabled.
Set this bit when the OMM0 bit of the RLN3nLMST register is 0_B (in LIN reset mode).
Do not set this bit to 1 when the UART buffer is used.

13.3.4.18 RLN3nLUTDR — UART Transmission Data Register

Access: RLN3nLUTDR register can be read/written in 16-bit units.
RLN3nLUTDRL register can be read/written in 8-bit units.
RLN3nLUTDRH register can be read/written in 8-bit units.

Address: RLN3nLUTDR: <RLIN3n_base> + 24_H
RLN3nLUTDRL: <RLIN3n_base> + 24_H
RLN3nLUTDRH: <RLIN3n_base> + 25_H

Value after reset: 0000_h

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UTD[8:0]								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.67 RLN3nLUTDR Register Contents

Bit Position	Bit Name	Function
15 to 9	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
8 to 0	UTD[8:0]	Sets the data to be transmitted from the transmission buffer. Setting range: 000_H to $1FF_H$

The RLN3nLUTDR register sets the data to be transmitted from the transmit data register. Writing data to this register with the UTOE bit in the RLN3nLUOER register set to 1 starts transmission.

This register can be accessed in 8 bits.

In 9-bit communication mode, do not attempt 8-bit access.

Do not write data to this register when data transmission from the UART buffer is in progress.

Also, do not write data to this register when a transmission request is being generated due to write access to the RLN3nLUWTDR register.

When transmitting data continuously, do not set another piece of transmission data in this register before the generation of transmission interrupt.

The table below shows the bit arrangement according to the set communication format.

Table 13.68 Bit Arrangement of the RLN3nLUTDR Register According to Each Communication Format

	RLN3nLUTDR								
	b8	b7	b6	b5	b4	b3	b2	b1	b0
7-bit; LSB first	—	—	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
7-bit; MSB first	—	—	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6
8-bit; LSB first	—	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
8-bit; MSB first	—	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
9-bit; LSB first	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
9-bit; MSB first	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8

13.3.4.19 RLN3nLURDR — UART Reception Data Register

Access: RLN3nLURDR register can only be read in 16-bit units.
RLN3nLURDRL register can only be read in 8-bit units
RLN3nLURDRH register can only be read in 8-bit units

Address: RLN3nLURDR: <RLIN3n_base> + 26_H
RLN3nLURDRL: <RLIN3n_base> + 26_H
RLN3nLURDRH: <RLIN3n_base> + 27_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	URD[8:0]								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 13.69 RLN3nLURDR Register Contents

Bit Position	Bit Name	Function
15 to 9	Reserved	When read, the value after reset is returned.
8 to 0	URD[8:0]	Hold the received data. Setting range: 000_H to $1FF_H$

The RLN3nLURDR allows the reception data to be read from the receive data register.

When the UROE bit in the RLN3nLUOER register is 1, the reception data is stored in this register and can be read out.

This register is updated at the stop bit of the reception data.

This register is also updated when an error is caused by the parity or stop bit. However, the value of this register is not updated upon occurrence of an overrun error when the OERE bit of the RLN3nLEDE register is 1 (overrun detection enabled). The value of this register is updated upon occurrence of an overrun error when the OERE bit is 0 (overrun detection disabled).

Read this register upon occurrence of a receive error (overrun error, framing error, parity error) when the OERE bit of the RLN3nLEDE register is 1 (overrun error detection enabled). If the next data is received without reading this register, an overrun error occurs.

This register can be accessed in 8 bits. However, during expansion bit use (UEBE bit of the RLN3nLUOR1 register is 1 (expansion bit operation enabled), do not attempt 8-bit access.

The table below shows the bit arrangement according to the set communication format.

Table 13.70 Bit Arrangement of the RLN3nLURDR Register According to Each Communication Format

	RLN3nLURDR								
	b8	b7	b6	b5	b4	b3	b2	b1	b0
7-bit; LSB first	—	—	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
7-bit; MSB first	—	—	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6
8-bit; LSB first	—	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
8-bit; MSB first	—	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
9-bit; LSB first	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
9-bit; MSB first	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8

13.3.4.20 RLN3nLUWTDR — UART Wait Transmission Data Register

Access: RLN3nLUWTDR register can be read/written in 16-bit units.
RLN3nLUWTDRL register can be read/written in 8-bit units.
RLN3nLUWTDRH register can be read/written in 8-bit units.

Address: RLN3nLUWTDR: <RLIN3n_base> + 28_H
RLN3nLUWTDRL: <RLIN3n_base> + 28_H
RLN3nLUWTDRH: <RLIN3n_base> + 29_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UWTD[8:0]								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.71 RLN3nLUWTDR Register Contents

Bit Position	Bit Name	Function
15 to 9	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
8 to 0	UWTD[8:0]	Sets the data to be transmitted from the wait transmit data register after waiting for the stop bit reception to be completed. Setting range: 000_H to $1FF_H$

The RLN3nLUWTDR register sets the data to be transmitted from the UART wait transmit data register.

Writing data to this register with the UTOE bit in the RLN3nLUOER register set to 1 starts transmission.

Use this register only to switch from reception to transmission in half-duplex communication. The user should write to this register only while the stop bit is being received.

Note that the wait time is only 1 bit even if the stop bit length is set to 2 bits with the USBLS bit in the RLN3nLBFC register.

When this register is read, the RLN3nLUTDR register value is actually read.

In 9-bit communication mode, do not attempt 8-bit access.

Do not write data to this register when data transmission from the UART buffer is in progress.

The table below shows the bit arrangement according to the set communication format.

Table 13.72 Bit Arrangement of the RLN3nLUWTDR Register According to Each Communication Format

	RLN3nLUWTDR								
	b8	b7	b6	b5	b4	b3	b2	b1	b0
7-bit; LSB first	—	—	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
7-bit; MSB first	—	—	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6
8-bit; LSB first	—	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
8-bit; MSB first	—	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
9-bit; LSB first	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
9-bit; MSB first	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8

13.4 Interrupt Sources

The LIN/UART interface generates 4 types of interrupt request signals.

- RLIN3n successful transmission interrupt
- RLIN3n successful reception interrupt
- RLIN3n status interrupt
- RLIN3n interrupt

Setting the LIOS bit in the RLN3nLMD register to 0 allows to perform logical OR operation on all of the interrupt sources, outputting the interrupt request from the RLIN3n interrupt.

If the LIOS bit of the RLN3nLMD register is set to 1, either RLIN3n transmit interrupt, RLIN3n receive completion interrupt, or RLIN3n status interrupt request signal is output according to the interrupt source.

Table 13.73 lists the sources for each interrupt.

Table 13.73 Interrupt Sources

		LIOS bit in RLN3nLMD register is 0	LIOS bit in RLN3nLMD register is 1*1		
		RLIN3n Interrupt	RLIN3n Transmission Interrupt	RLIN3n Successful Reception Interrupt	RLIN3n Status Interrupt
LIN mode	LIN master mode	• Successful frame transmission • Successful frame reception • Successful wake-up transmission • Successful wake-up reception • Successful header transmission • Bit error • Physical bus error • Frame/response timeout error • Framing error • Checksum error • Response preparation error	• Successful frame transmission • Successful wake-up transmission • Successful header transmission	• Successful frame reception • Successful wake-up reception	• Bit error • Physical bus error • Frame/response timeout error • Framing error • Checksum error • Response preparation error
	LIN slave mode	• Successful response transmission • Successful response reception • Successful wake-up transmission • Successful wake-up reception • Successful header reception • Bit error • Frame/response timeout error • Framing error • Sync field error • Checksum error • ID parity error • Response preparation error	• Successful response transmission • Successful wake-up transmission	• Successful response reception • Successful wake-up reception • Successful header reception	• Bit error • Frame/response timeout error • Framing error • Sync field error • Checksum error • ID parity error • Response preparation error
UART mode		—	• Transmission start/successful transmission	• Successful reception • Expansion bit mismatch	• Bit error • Overrun error • Framing error • Expansion bit match • ID match • Parity error

Note 1. The LIOS bit setting is valid in LIN Mode. In UART mode, setting the LIOS bit is not required.

In LIN mode, each interrupt request is output when the corresponding bit in the RLN3nLIE register is 1 (interrupt is enabled) and the corresponding flag in the RLN3nLST register is 1.

13.5 Modes

The LIN/UART interface provides the following four modes, depending upon the specific function to be performed:

- LIN reset mode
- LIN mode
  - LIN master mode
  - LIN slave mode [auto baud rate]
  - LIN slave mode [fixed baud rate]
- UART mode
- LIN self-test mode

Figure 13.2 shows mode transitions. **Table 13.74** describes mode transition conditions. **Table 13.75** lists operations available in each mode.

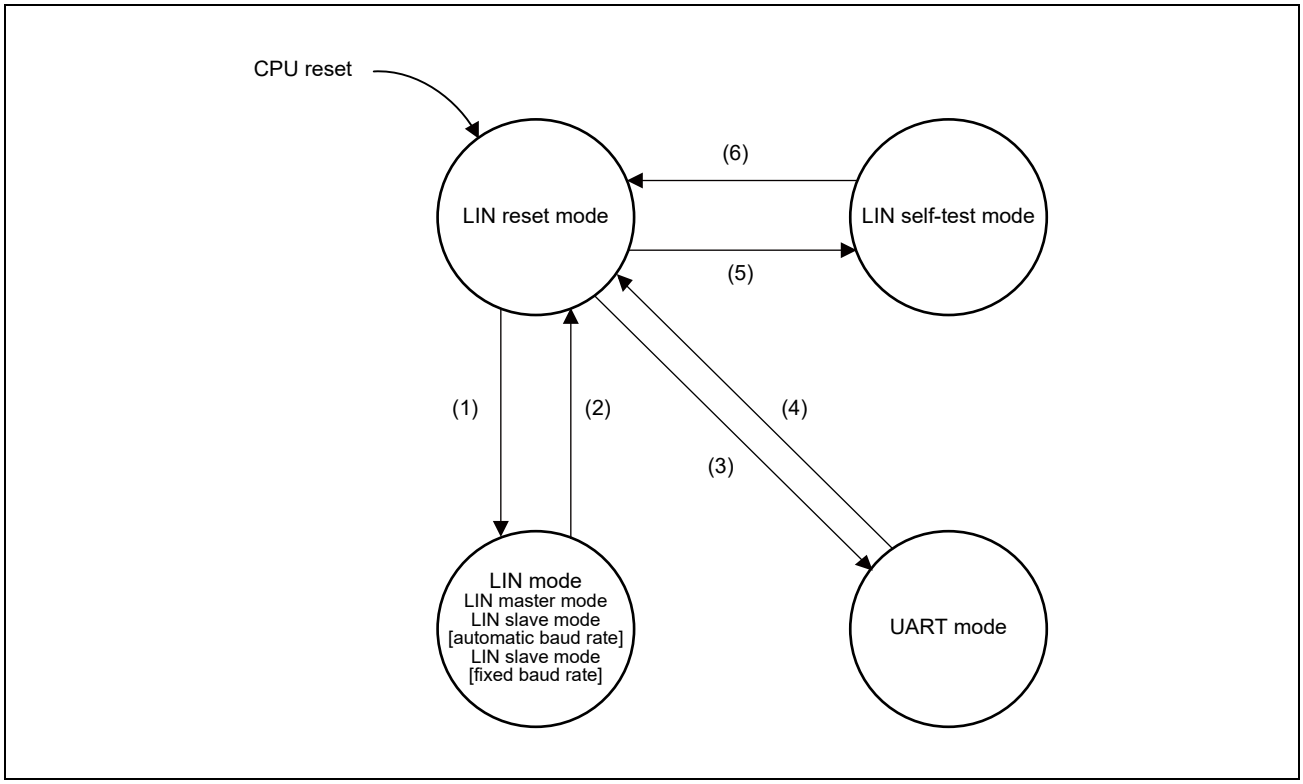


Figure 13.2 Mode Transitions

Table 13.74 Transition Condition of Each Mode

	Mode Transition			Transition Condition
(1)	LIN reset mode	→	LIN mode • LIN master mode • LIN slave mode [auto baud rate] • LIN slave mode [fixed baud rate]	• RLN3nLMD.LMD = “00_B” and RLN3nLCUC.OM1, OM0 = ”01_B” or “11_B” • RLN3nLMD.LMD = “11_B” and RLN3nLCUC.OM1,OM0 = “01_B” or “11_B” • RLN3nLMD.LMD = “10_B” and RLN3nLCUC.OM1,OM0 = “01_B” or “11_B”
(2)	LIN mode	→	LIN reset mode	RLN3nLCUC.OM0 = “0_B”
(3)	LIN reset mode	→	UART mode	RLN3nLMD.LMD = “01_B” and RLN3nLCUC.OM0 = “1_B”
(4)	UART mode	→	LIN reset mode	RLN3nLCUC.OM0 = “0_B”
(5)	LIN reset mode	→	LIN self-test mode	See **Section 13.9, LIN Self-Test Mode**.
(6)	LIN self-test mode	→	LIN reset mode	See **Section 13.9, LIN Self-Test Mode**.

Table 13.75 Operations Available in Each Mode

LIN Mode			
LIN Master Mode	LIN Slave Mode [auto baud rate] LIN Slave Mode [fixed baud rate]	UART Mode	LIN Self-Test Mode
Header transmission Response transmission Response reception Wake-up transmission Wake-up reception Error detection	Header reception Response transmission Response reception Wake-up transmission Wake-up reception Error detection	UART transmission UART reception Error detection	Self test

Whether a transition has been caused to LIN reset mode, the LIN mode, or the UART mode can be verified by reading the LMD bits in the RLN3nLMD register or the OMM0 bit in the RLN3nLMST register.

For a description of the LIN self-test mode, see **Section 13.9, LIN Self-Test Mode**.

13.6 LIN Reset Mode

Setting the OM0 bit in the RLN3nLCUC register to 0 (LIN reset mode) causes a transition to LIN reset mode. The change to LIN reset mode can be verified by determining that the OMM0 bit in the RLN3nLMST register has been set to 0 (LIN reset mode). In this mode, the LIN communication and the UART communication functions are halted.

From LIN reset mode, transitions to LIN mode, UART mode, and LIN self-test mode can be made.

When the mode changes to LIN reset mode, the following registers are initialized to their reset values, and as long as LIN reset mode is in effect, they retain their initial values.

- RLN3nLTRC register
- RLN3nLST register
- RLN3nLEST register
- RLN3nLUOER register

The following registers retain their previous values even when a transition to LIN reset mode is made:

- RLN3nLWBR register
- RLN3nLBRP0 register
- RLN3nLBRP1 register
- RLN3nLMD register
- RLN3nLBFC register
- RLN3nLSC register
- RLN3nLWUP register
- RLN3nLIE register
- RLN3nLEDE register
- RLN3nLDFC register
- RLN3nLIDB register
- RLN3nLCBR register
- RLN3nLUDB0 register
- RLN3nLDBRb register (b = 1 to 8)
- RLN3nLUOR1 register
- RLN3nLUTDR register
- RLN3nLURDR register
- RLN3nLUWTDR register

13.7 LIN Mode

LIN mode can operate in the following submodes: LIN master mode, LIN slave mode [auto baud rate], and LIN slave mode [fixed baud rate].

In LIN master mode, the following operations can be performed: header transmission, response transmission, response reception, wake-up transmission, wake-up reception, and error detection. In LIN reset mode, setting the LMD bits in the RLN3nLMD register to 00_B (LIN master mode) and the OM1 and OM0 bits in the RLN3nLCUC register to either 01_B or 11_B sets LIN master mode, turning the OMM1 and OMM0 bits in the RLN3nLMST register to either 01_B to 11_B.

In LIN slave mode [auto baud rate] and LIN slave mode [fixed baud rate], header reception, response transmission, response reception, wake-up transmission, wake-up reception, and error detection can be performed.

The LIN slave mode [auto baud rate] allows automatic detection of the break field and the sync field, and sets a baud rate based on the results of measurement of a sync field. The baud rate can be set to the range from 1kbps to 20kbps. Set the LPRS[2:0] bits in the RLN3nLWBR register so that the prescaler clock (with the frequency of the LIN communication clock source divided by the prescaler) becomes as follows according to the target baud rate.

[Target baud rate]	[Prescaler clock]
1 kbps to 20 kbps	: 4 MHz*1
1 kbps to 2.4 kbps (excluding 2.4 kbps)	: 4 MHz
2.4 kbps to 20 kbps	: 8 MHz to 12 MHz

Note 1. Use the clock with NSPB[3:0] bits in the RLN3nLWBR register set to "0011_B" (four samplings).

In LIN slave mode [fixed baud rate] allows automatic detection of the break field, the sync field, and the ID filed at a baud rate that is set in advance by the baud rate generator.

In LIN reset mode, setting the LMD bits in the RLN3nLMD register to 10_B (LIN slave mode [auto baud rate] and setting the OM1 and OM0 bit in the RLN3nLCUC register to 01_B or 11_B sets LIN slave mode [auto baud rate]; and setting the LMD bits in the RLN3nLMD register to 11_B (LIN slave mode [fixed baud rate]), and setting the OM1 and OM0 bits in the RLN3nLCUC register to 01_B or 11_B sets LIN slave mode [fixed baud rate], turning the OMM1 and OMM0 bits in the RLN3nLMST register to 01_B or 11_B.

When changing a submode to another submode within LIN mode, a transition to LIN reset mode should first be made and change the LMD bits in the RLN3nLMD register.

The LIN mode provides the following two operation modes:

- LIN operation mode
- LIN wake-up mode

Figure 13.3 shows the transition of operation modes. **Table 13.76** describes the transition conditions of operation modes.

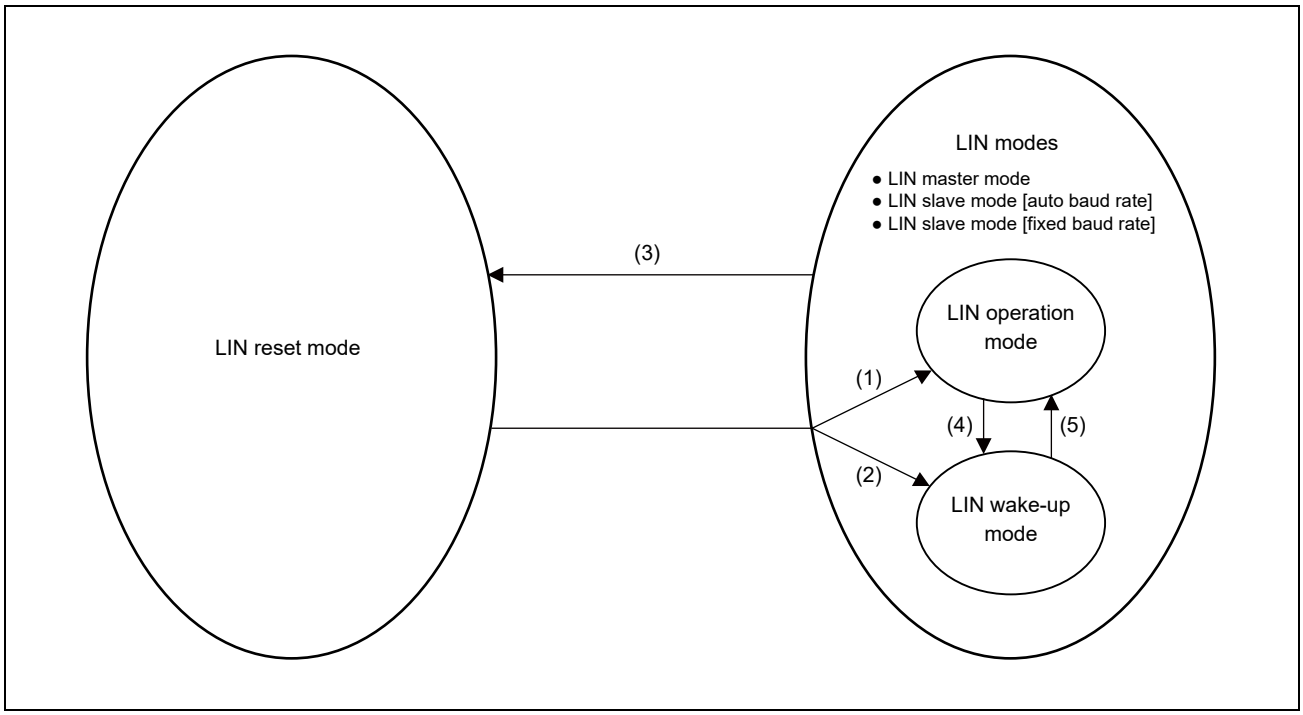


Figure 13.3 Transition of Operation Modes

Table 13.76 Transition Condition for Operation Mode

	Operation Mode Transition			Transition Condition
(1)	LIN reset mode	→	LIN mode • LIN operation mode	LMD bit in RLN3nLMD register = 00_B or 10_B or 11_B and OM1 and OM0 bits in RLN3nLCUC register = 11_B
(2)	LIN reset mode	→	LIN mode • LIN wake-up mode	LMD bit in RLN3nLMD register = 00_B or 10_B or 11_B and OM1 and OM0 bits in RLN3nLCUC register = 01_B
(3)	LIN mode • LIN operation mode • LIN wake-up mode	→	LIN reset mode	OM0 bit in RLN3nLCUC register = 0_B
(4)*1	LIN mode	→	LIN mode • LIN wake-up mode	OM1 and OM0 bits in RLN3nLCUC register = 01_B
(5)*1	LIN mode • LIN wake-up mode	→	LIN mode • LIN operation mode	OM1 and OM0 bits in RLN3nLCUC register = 11_B

Note 1. Transition between LIN operation mode and LIN wake-up mode cannot be made when communication is going on (when the FTS bit in the RLN3nLTRC register is 1).

(1) LIN Operation Mode

In LIN operation mode, frame processing (header transmission, header reception, response transmission, response reception, and error detection) can be performed.

During a transition from LIN reset mode to LIN mode, setting the OM1 and OM0 bits in the RLN3nLCUC register to 11_B changes the mode to LIN operation mode, changing the OMM1 and OMM0 bits in the RLN3nLMST register to 11_B. Communication settings should be performed after the OMM1 and OMM0 bits have become 11_B.

(2) LIN Wake-up Mode

In LIN wake-up mode, wake-up signal processing (wake-up transmission, wake-up reception, and error detection) can be performed.

During a transition from LIN reset mode to LIN mode, setting the OM1 and OM0 bits in the RLN3nLCUC register to 01_B changes the mode to LIN wake-up mode, changing the OMM1 and OMM0 bits in the RLN3nLMST register to 01_B. Communication settings should be performed after the OMM1 and OMM0 bits have become 01_B.

13.7.1 LIN Master Mode

13.7.1.1 Header Transmission

Figure 13.4 shows the operation of the LIN/UART interface (LIN master mode) in header transmission. **Table 13.77** provides processing in header transmission.

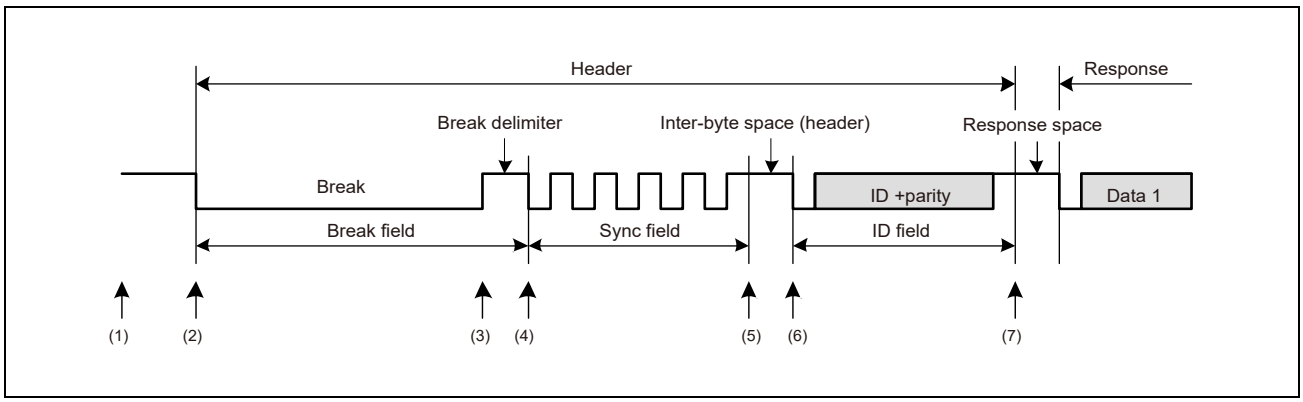


Figure 13.4 Operation in Header Transmission

Table 13.77 Processing in Header Transmission

	Software Processing	LIN/UART Interface Processing
(1)	• Sets a baud rate • Sets noise filter ON/OFF • Enables interrupt • Enables error detection • Sets frame configuration parameters • Changes the LIN/UART interface to the LIN maste rmode: LIN operation mode • Sets information on the frame to be transmitted (ID,parity, data length, response direction, Checksum method, and transmission data)	Waits for the setting of the FTS bit in the RLN3nLTRC register by software (idle)
(2)	Sets the FTS bit in the RLN3nLTRC register to 1 (frame transmission or wake-up transmission/reception started)	Transmits a break.
(3)	Waits for an interrupt request	Transmits a break delimiter.
(4)		Transmits a sync field (55_H).
(5)		Transmits an inter-byte space (header).
(6)		Transmits an ID field.
(7)		Sets a successful header transmission flag.

NOTE

For information about error detection conditions, see **Section 13.7.7, Error Status**.

13.7.1.2 Response Transmission

Figure 13.5 shows the operation of the LIN/UART interface (LIN master mode) in response transmission. **Table 13.78** provides processing in response transmission.

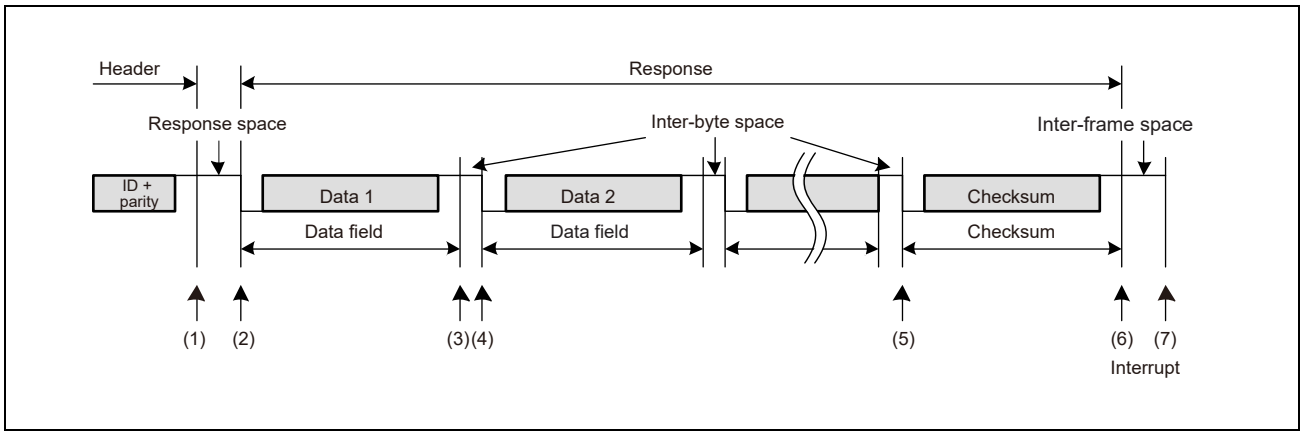


Figure 13.5 Operation in Response Transmission

Table 13.78 Processing in Response Transmission

	Software Processing	LIN/UART Interface Processing
(1)	[When in frame separate mode] • Sets the RTS bit in the RLN3nLTRCregister to 1 (response transmission/reception started) [When not in frame separate mode] • Waits for an interrupt request	[When in frame separate mode] • Waits for the setting of the RTS bit in the RLN3nLTRC register to 1 by software. • When the bit is set to 1, sends a response space. [When not in frame separate mode] • Sends a response space.
(2)	Waits for an interrupt request	Transmits data 1.
(3)		Transmits an inter-byte space.
(4)		• Transmits data 2. • Transmits an inter-byte space • Transmits data 3. • Transmits an inter-byte space (Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RFC register). : :
(5)		Transmits the checksum.
(6)		• Sets a successful frame/wake-up transmission flag. • Sets the FTS bit in the RLN3nLTRC register to 0 (frame transmission or wake-up transmission/reception stopped) [When in frame separate mode] • Sets the RTS bit in the RLN3nLTRC register to 0 (response transmission/reception is halted).
(7)	• Processing after communication Checks the RLN3nLST register, and clears flags.	Idle

NOTE

For information about error detection, see **Section 13.7.7, Error Status**.

13.7.1.3 Response Reception

Figure 13.6 shows the operation of the LIN/UART interface (LIN master mode) on response reception. **Table 13.79** provides processing in response reception.

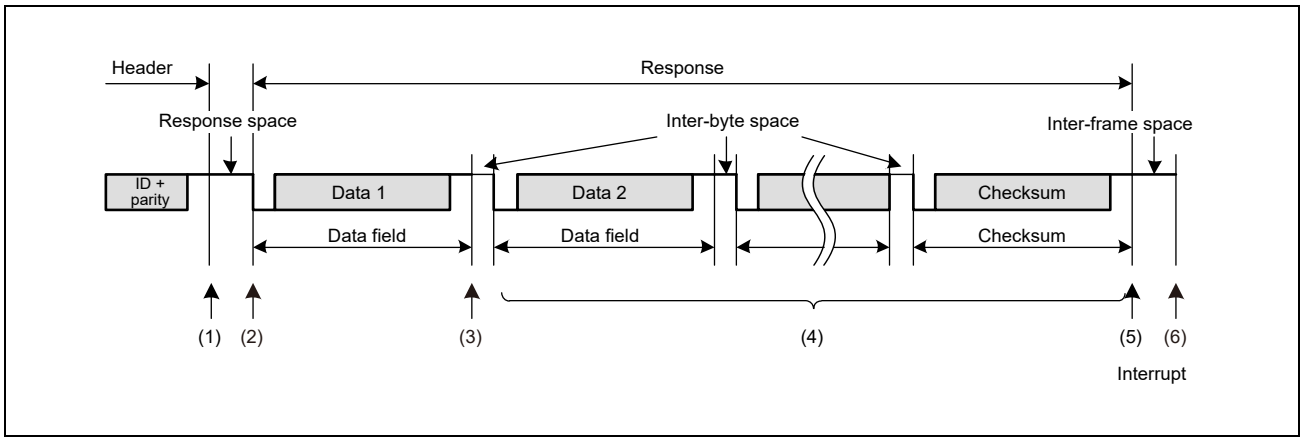


Figure 13.6 Operation in Response Reception

Table 13.79 Processing in Response Reception

	Software Processing	LIN/UART Interface Processing
(1)	Waits for an interrupt request (no processing)	Waits for detection of a start bit.
(2)		Receives data 1 when the start bit is detected.
(3)		Sets the successful data 1 reception flag.
(4)		• Receives data 2 when the start bit is detected. • Receives data 3 when the start bit is detected. Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RLN3nLDFC register). : : • Receives the checksum when the start bit is detected.
(5)		• Determines the checksum. • Sets the successful frame/wake-up reception flag. • Sets the FTS bit in the RLN3nLTRC register to 0 (frame transmissionor wake-up transmission/reception stopped).
(6)	• Processing after communication Reads the received data. Checks the RLN3nLST register, and clears flags.	Idle

NOTE

For information about error detection, see **Section 13.7.7, Error Status**.

13.7.2 LIN Slave Mode

13.7.2.1 Header Reception

Figure 13.7 shows the operation of the LIN/UART interface (LIN slave mode) in header reception.**Table 13.80** provides processing in header reception.

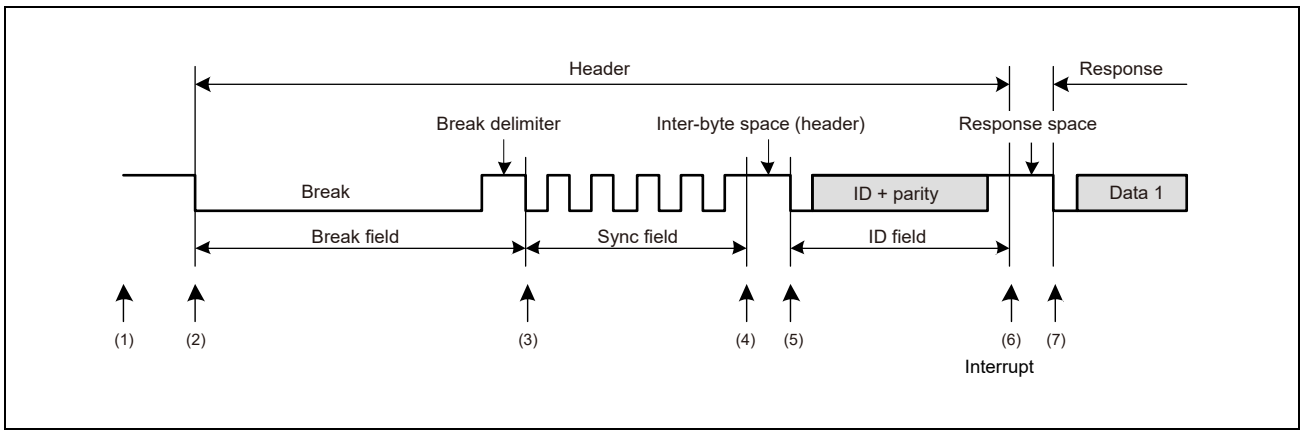


Figure 13.7 Operation in Header Reception

Table 13.80 Processing in Header Reception

	Software Processing	LIN/UART Interface Processing
(1)	• Sets a baud rate • Sets noise filter ON/OFF • Enables interrupt • Enables error detection • Sets frame configuration parameters • Changes the LIN/UART interface to the LIN slave mode: LIN operation mode • Sets the FTS bit in the RLN3nLTRC register to 1 (header reception or wake-up transmission/reception started)	Waits for the setting of the FTS bit in the RLN3nLTRC register by software.
(2)	Waits for an interrupt request.	Waits for detection of break field
(3)		Detects a break field. In the case of break field detection (LIN slave mode [fixed baud rate]). For details about the break field detection timing in the case of LIN slave mode [auto baud rate], see [Auto Baud Rate Correction Function])
(4)		• Detects a sync field (55_H) • Baud rate generator setting (in the case of LIN slave mode [auto baud rate]) • Clears the no-response request bit (LNRR bit).
(5)		• Receives an ID field. • Checks an ID parity bit
(6)		Sets a header reception complete flag.
(7)	• Checks the RLN3nLST register, and clearsflags. • Checks the RLN3nLIDB register, and prepares a response.	• Completes a header reception process. • Waits for a response request.

NOTE

The LIN/UART interface allows reception of break fields during frame transmission/reception. In that case, a framing error, bit error or other error may be detected at the stop bit position of the frame before the break field is received, and a status interrupt may occur as a result. However, reception of a new header (the following Sync field and ID field) continues regardless of whether an error occurred. For information about error detection conditions, see **Section 13.7.7, Error Status**.

[Auto Baud Rate Correction Function]

In LIN slave mode [auto baud rate], the system always measures the low-level widths that are received. If the first "Low level" width is 10 times (if the BLT bit of the RLN3nLBFC register is "0") or 11 times (if the BLT bit of the RLN3nLBFC register is "1") or greater calculated from the average of the starting 2 bits (the period of the consecutive fall edges from the beginning of the sync field) of the sync field, the system concludes that the detection of break field was successful, the system verifies that the data in the sync field is 55_H. If the data in the sync field is indeed 55_H and the system judges that sync field reception was successful, the system automatically sets the baud rate correction result to the RLN3nLBRP01 register.

If data is received up to the ID field without error, a successful header reception interrupt is generated at the stop bit position.

On the other hand, if the data in the sync field is not 55_H and the system judges that sync field reception failed, the system sets the sync field error flag and an error interrupt is generated. In that case, baud rate correction is not performed and the LIN/UART interface waits for the detection of the next break field (low level).

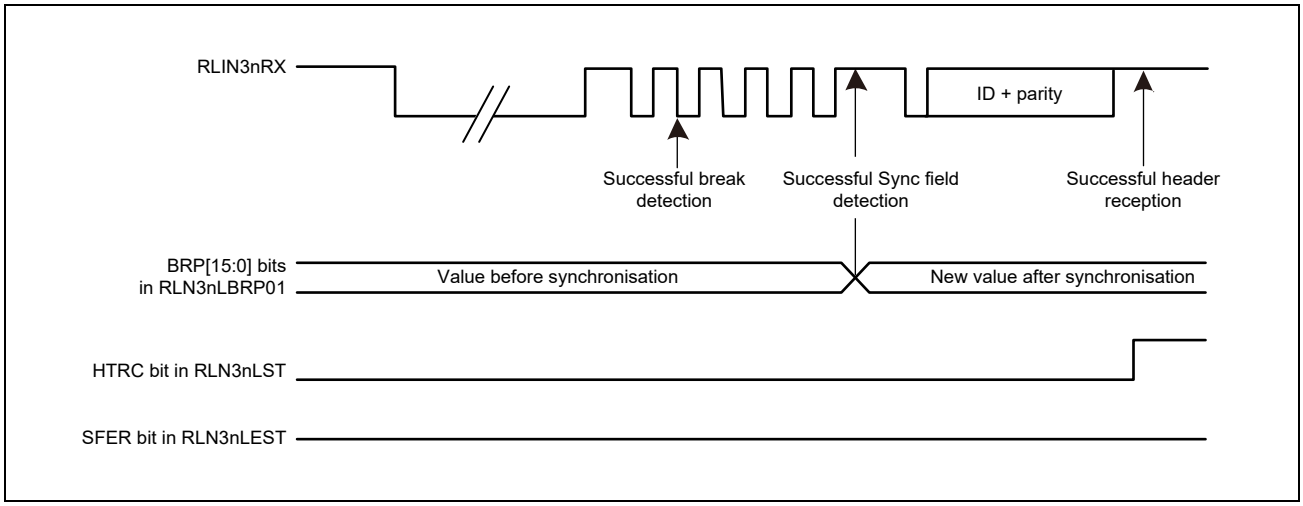


Figure 13.8 Header Reception in LIN Slave Mode [Auto Baud Rate] (in Normal Operation)

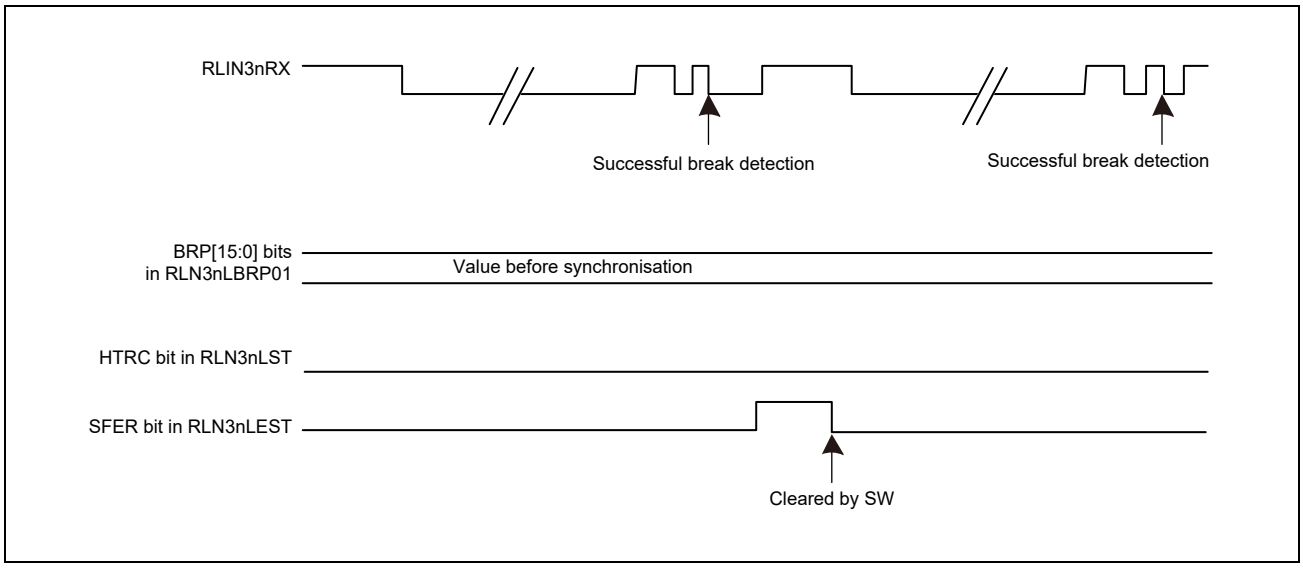


Figure 13.9 Header Reception in LIN Slave Mode [Auto Baud Rate] (Sync Field Error)

13.7.2.2 Response Transmission

Figure 13.10 shows the operation of the LIN/UART interface (in LIN slave mode) in response transmission. **Table 13.81** provides processing in response transmission.

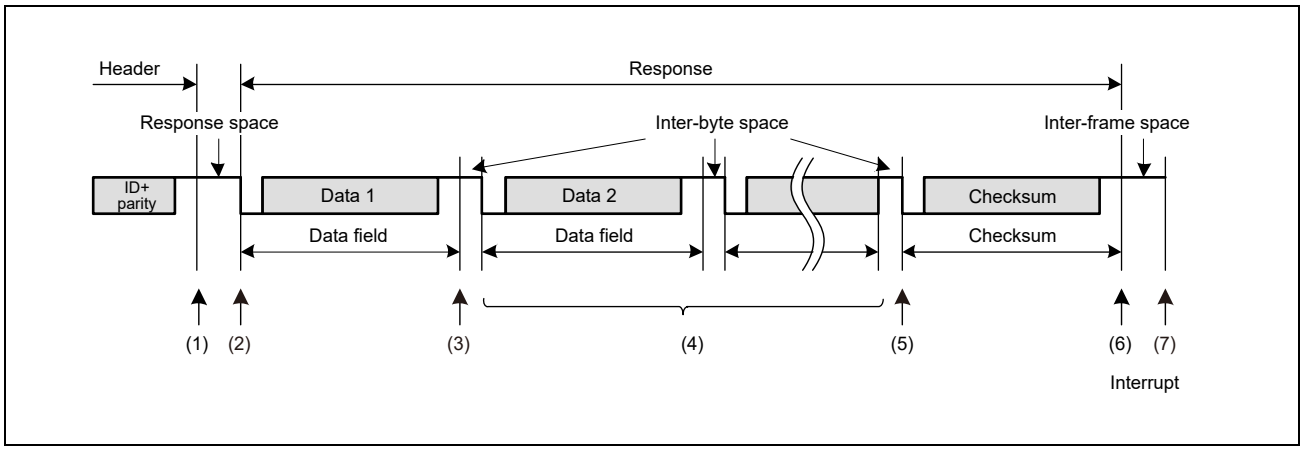


Figure 13.10 Operation in Response Transmission

Table 13.81 Processing in Response Transmission

	Software Processing	LIN/UART Interface Processing
(1)	• Sets the RLN3nLDFC register. • Sets the RLN3nLDBRbregisters.(b = 1 to 8) • Sets the RTS bit in theRLN3nLTRC register to 1 (response transmission/reception started)	• Waits for the setting of the RTS or LNRR bit of the RLN3nLTRC register by software • Transmits the response space after the RTS bit of the RLN3nLTRC register is set to 1
(2)	Waits for an interrupt request.	Transmits data 1.
(3)		Transmits the inter-byte space.
(4)		• Transmits data 2. • Transmits an inter-byte space • Transmits data 3. • Transmits an inter-byte space (Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RLN3nLDFC register). : :
(5)		Transmits the checksum.
(6)		• Sets a successful response/wake-up transmission flag. • Sets the RTS bit in the RLN3nLTRC register to 0 (response transmission/reception stopped).
(7)	• Processing after communication Checks the RLN3nLST register, and clears flags.	• Completes the response transmission process. • Waits for a new break.

NOTE

- For information about error detection, see **Section 13.7.7, Error Status**.
- The LIN/UART interface allows reception of break fields during frame transmission/reception. In that case, a framing error, bit error or other error may be detected at the stop bit position of the frame before the break field is received, and a status interrupt may occur as a result. However, reception of a new header (the following Sync field and ID field) continues regardless of whether an error occurred.

13.7.2.3 Response Reception

Figure 13.11 shows the operation of the LIN/UART interface (LIN slave mode) in response reception. **Table** 13.82 provides processing in response reception.

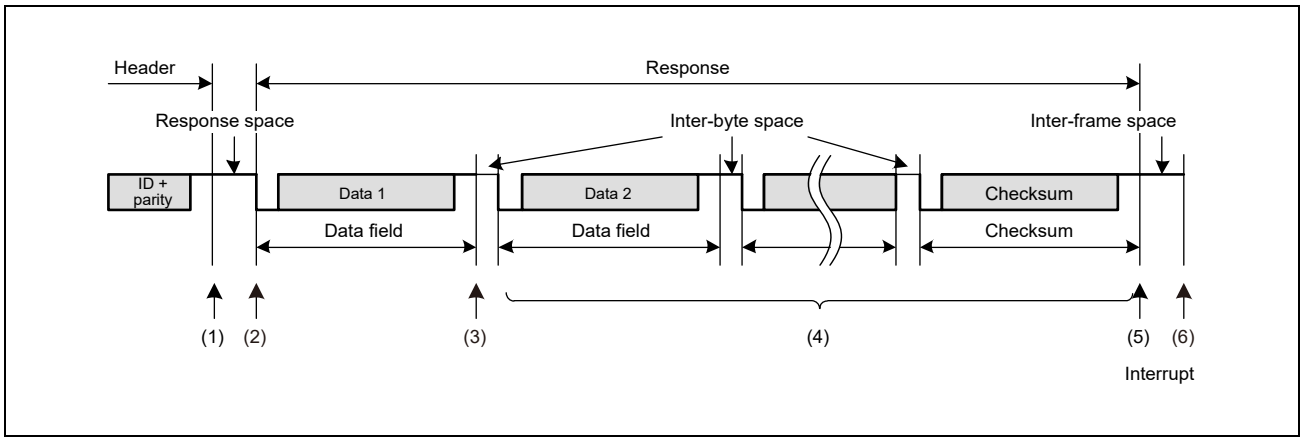


Figure 13.11 Operation in Response Reception

Table 13.82 Processing in Response Reception

	Software Processing	LIN/UART Interface Processing
(1)	• Sets the RLN3nLDFC register. • Sets the response transmission/reception start bit (RTS bit) to 1.	• Waits for the setting by software of the response transmission/reception start bit (RTS bit) or the no-response request bit (LNRR bit). • Waits for detection of the start bit.
(2)	Waits for an interrupt request.	Receives data 1 when the start bit is detected.
(3)		Sets the successful data 1 reception flag.
(4)		• Receives data 2 when the start bit is detected. • Receives data 3 when the start bit is detected. Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RLN3nLDFC register. : : • Receives the checksum when the start bit is detected.
(5)		• Determines the checksum. • Sets a successful response/wake-up reception flag or an error flag. • Sets the RTS bit in the RLN3nLTRC register to 0 (response transmission/reception stopped).
(6)	• Processing after communication Reads the received data. Checks the RLN3nLST register, and clears flags.	• Completes the response process. • Waits for a new break.

NOTE

- For information about error detection conditions, see **Section 13.7.7, Error Status**.
- The LIN/UART interface allows reception of break fields during frame transmission/reception. In that case, a framing error, bit error or other error may be detected at the stop bit position of the frame before the break field is received, and a status interrupt may occur as a result. However, reception of a new header (the following Sync field and ID field) continues regardless of whether an error occurred.

13.7.2.4 No-Response Request

Figure 13.12 shows the operation of the LIN/UART interface (LIN slave mode) when no response is requested. **Table 13.83** shows the processing that occurs when no response is requested.

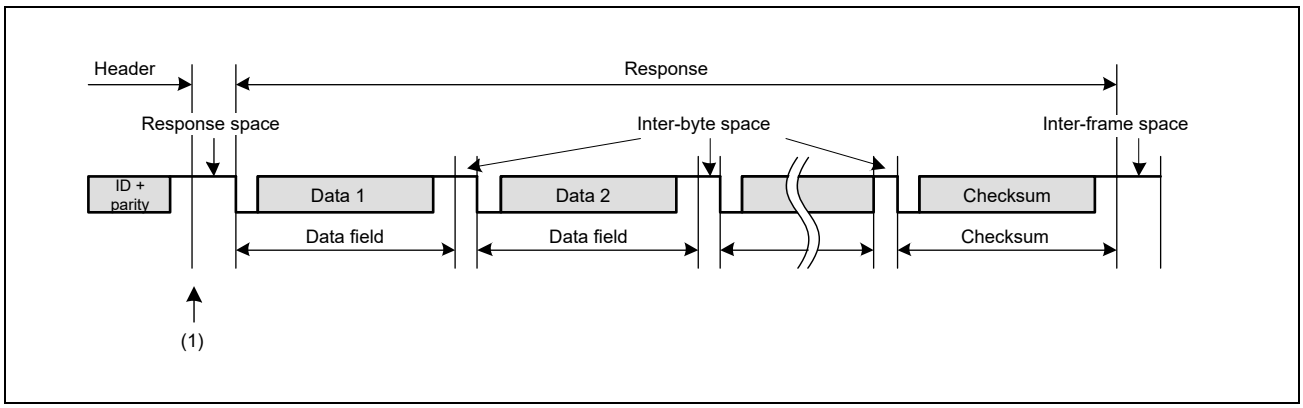


Figure 13.12 Operation when No Response is Requested

Table 13.83 Processing when No Response is Requested

	Software Processing	LIN/UART Interface Processing
(1)	• Sets the no-response request bit (LNRR bit) to 1.	• Waits for setting of the no-response request bit (LNRR bit) by software • Completes the frame reception process • Waits for a new break

13.7.3 Data Transmission/Reception

13.7.3.1 Data Transmission

One bit of data is transmitted per Tbit.

The data that is transmitted returns to the reception data input pin via the LIN transceiver. The received data and the transmitted data is compared bit by bit, and the results are stored in the BER flag in the RLN3nLEST register (see **Section 13.7.7, Error Status**).

In LIN master mode and LIN slave mode [fixed baud rate], 1 Tbit is generated to be 16fLIN, and thus the sampling point for received data is at the 13th clock cycle (81.25% position).

In LIN slave mode [auto baud rate], if 1 Tbit is generated to be 4fLIN, the sampling point for received data is at the third clock cycle (75% position). If 1 Tbit is generated to be 8fLIN, the sampling point for received data is at the 7th clock cycle (87.5% position).

Figure 13.13 shows an example of data transmission timing.

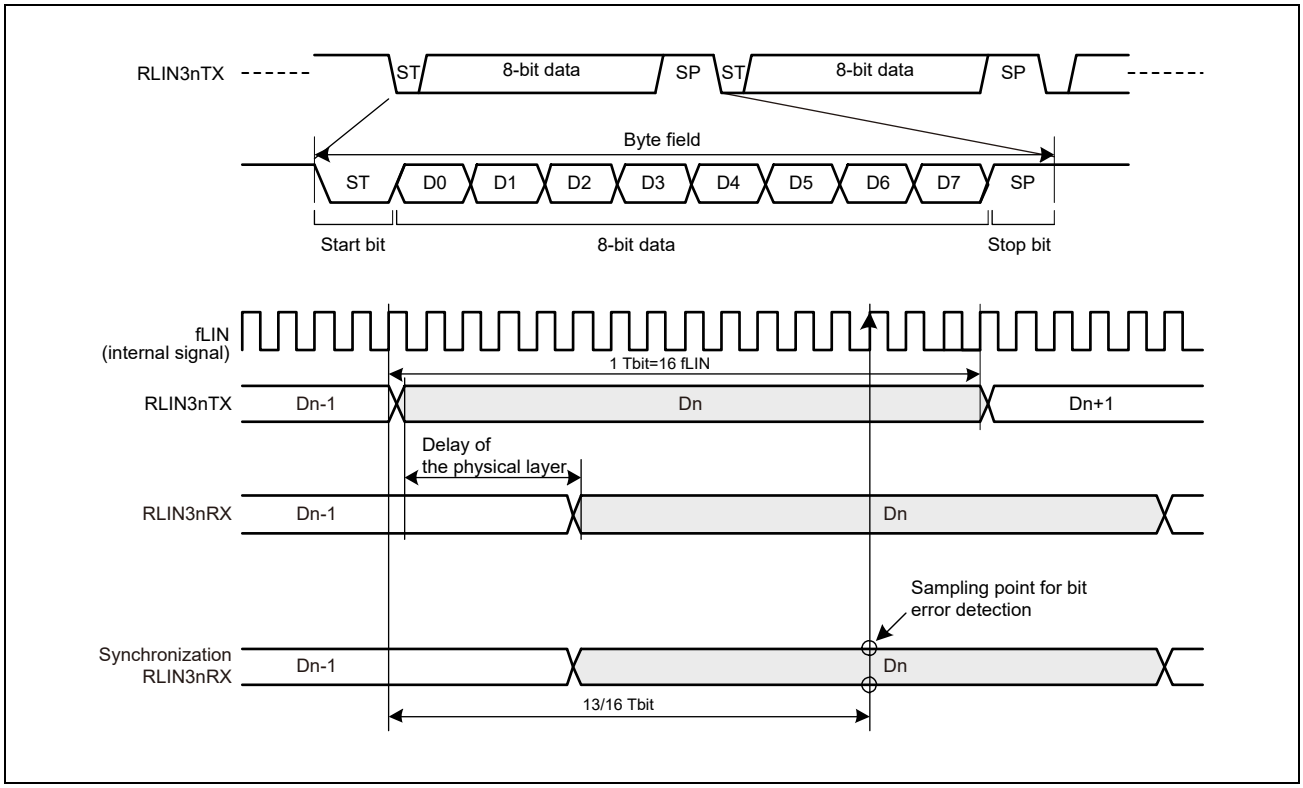


Figure 13.13 Example of Data Transmission Timing (LIN Master Mode, LIN Slave Mode [Fixed Baud Rate])

13.7.3.2 Data Reception

Data reception is performed by using the synchronized RLIN3nRX signal (an internal signal) that is the input from the RLIN3nRX pin synchronized with prescaler clock.

The byte field is synchronized at the falling edge of the start bit for the synchronized RLIN3nRX signal. After the falling edge is detected, sampling is performed again 0.5 Tbit later, and the falling edge is recognized as a start bit if the synchronized RLIN3nRX signal is low level. The falling edge is not recognized as a start bit if the RLIN3nRX signal after the clearing of the resetting is low-level-fixed or if a high level is detected on re-sampling.

After the start bit is detected, the system samples 1 bit per Tbit.

The LIN/UART interface has a noise filter function with respect to reception data. If the LRDNFS bit in the RLN3nLMD register is 0, the LIN/UART interface uses a noise filter, and for a sampling value the value determined by a 3-sampling majority rule on prescaler clocks is used. If the LRDNFS bit in the RLN3nLMD register is 1, the LIN/UART interface does not use a noise filter, and for a sampling value the value of the synchronized RLIN3nRX value at the sampling position is used as is.

Figure 13.14 shows an example of data reception timing.

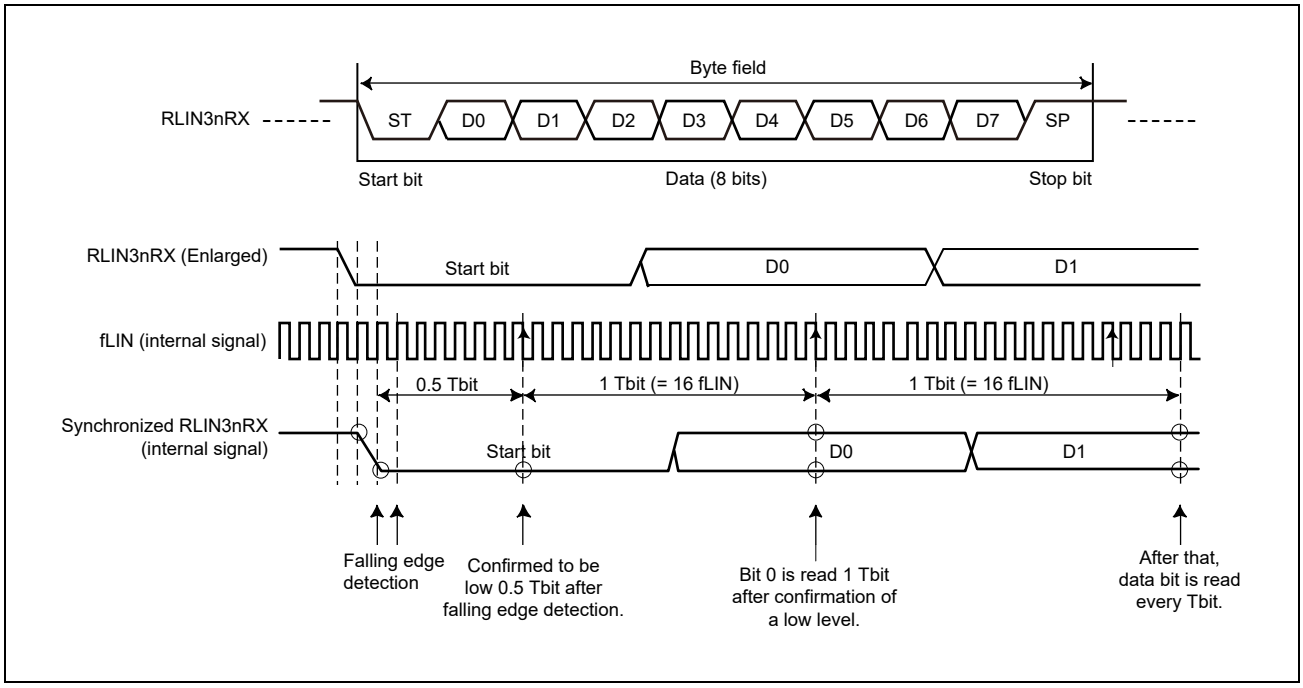


Figure 13.14 Example of Data Reception Timing (LIN Master Mode, LIN Slave Mode [Fixed Baud Rate])

13.7.4 Transmission/Reception Data Buffering

This section explains the buffer processing that takes place when the LIN/UART interface sends or receives data continuously.

13.7.4.1 Transmission of LIN Frames

For an 8-byte transmission, the contents stored in registers RLN3nLDBR1 to RLN3nLDBR8 are sequentially transmitted to data areas 1 to 8 of the LIN frame. In the case of a 4-bytes transmission, the contents stored in registers RLN3nLDBR1 to RLN3nLDBR4 are transmitted to data areas 1 to 4 of the LIN frame, but the contents of registers RLN3nLDBR5 to RLN3nLDBR8 are not transmitted. The transmitted checksum data is stored in the RLN3nLCBR register.

Figure 13.15 depicts the LIN transmission processing and the required buffer.

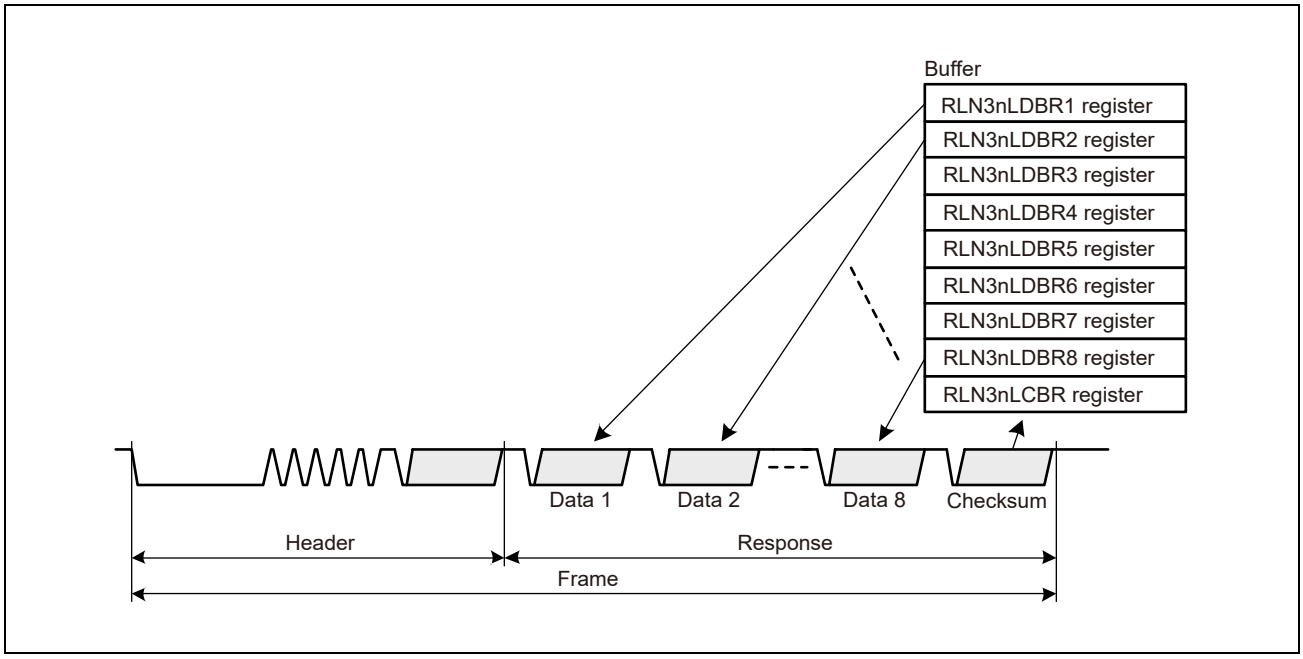


Figure 13.15 LIN Transmission Processing and Required Buffer

[Frame Separate Mode]

Setting the FSM bit in the RLN3nLDFC register to 1 turns on the frame separate mode.

In frame separate mode, a header and a response are transmitted when prompted by separate transmission start requests.

When the transmission of a header is finished, the HTRC flag in the RLN3nLST register turns 1 (successful header transmission).

Use frame separate mode when sending or receiving response data of 9 bytes or greater in LIN master mode.

13.7.4.2 Reception of LIN Frames

For an 8-byte reception, the contents of data areas 1 to 8 of the LIN frame is stored in registers RLN3nLDBR1 to RLN3nLDBR8, respectively, upon receipt of a stop bit. In the case of a 4-byte reception, the contents of data areas 1 to 4 of the LIN frame are stored in registers RLN3nLDBR1 to RLN3nLDBR4, respectively; however, no data is stored in registers RLN3nLDBR5 to RLN3nLDBR8. Also, the received checksum data is stored in the RLN3nLCBR register.

Figure 13.16 depicts the LIN reception processing and the required buffer.

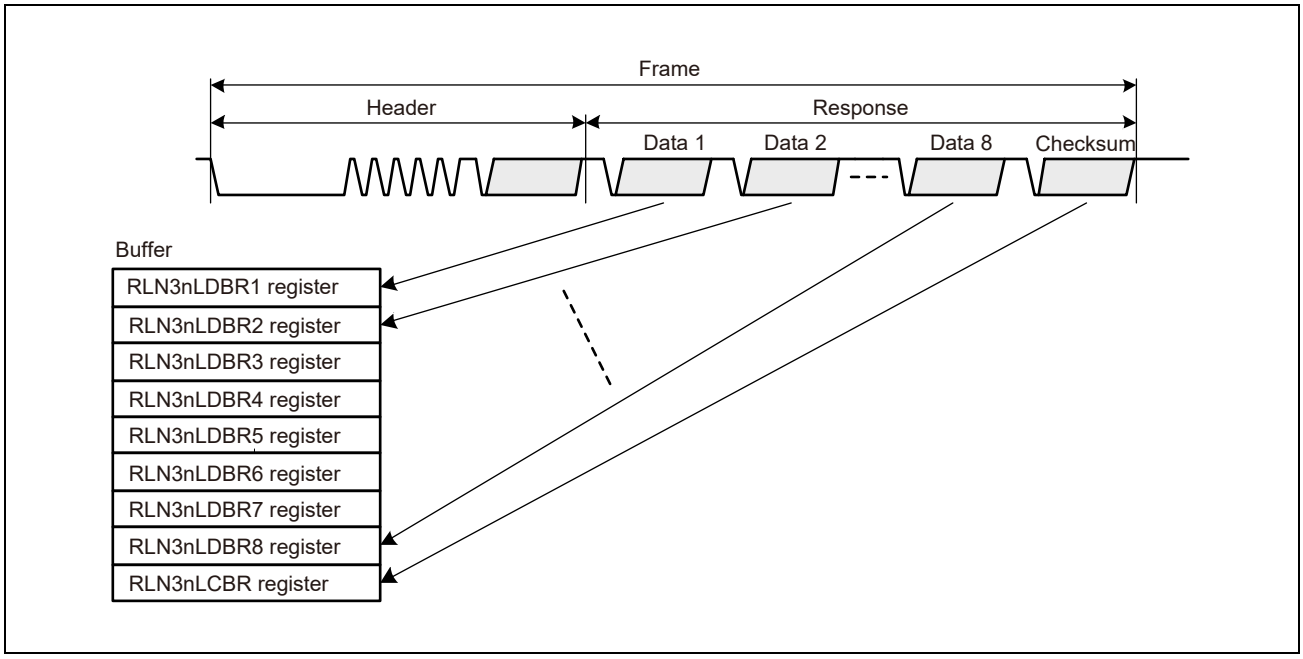


Figure 13.16 LIN Reception Processing and Required Buffer

[Reception of Data 1]

When the reception of the first byte of data is finished, the D1RC flag in the RLN3nLST register turns 1 (successful data 1 reception).

13.7.4.3 Multi-Byte Response Transmission/Reception Function

Normally in LIN communications, a response is 9 bytes or less including a checksum field; however, responses in 10 bytes or greater can also be sent and received.

In such a case, the bit error, framing error, response preparation error detection, and auto checksum functions are enabled.

If the data length is greater than 8 bytes, the LSS bit in RLN3nLDFC register should be set to 1

(indicating that the next data group to be sent or received is not the final data group) in the first data group (variable in 0 to 8 bytes) before sending or receiving the data group. After the transmission or reception, the user should determine whether the next data group is the final data group. If it is the final data group, the LSS bit should be set to 0 (indicating that the next data group to be sent or received is the final data group), and a checksum should be appended to the final data group.

By changing the RFDL bit in RLN3nLDFC register settings when the RTS bit in RLN3nLTRC register is 0, the user can change the data length for each data group.

When performing multi-byte response transmission/reception in LIN master mode, set the FSM bit in RLN3nLDFC register in the RLN3nLDFC register to 1 (frame separate mode).

NOTE

In LIN slave mode, the LIN/UART interface can detect a new break field during the transmission or reception of a response.

13.7.5 Wake-Up Transmission/Reception

The wake-up transmission/reception can be used in LIN wake-up mode.

13.7.5.1 Wake-Up Transmission

In LIN wake-up mode, setting the RFT bit in the RLN3nLDFC register to 1 (LIN master mode: response transmission), or setting the RCDS bit in the RLN3nLDFC register to 1 (LIN slave mode: response transmission), and then the FTS bit in the RLN3nLTRC register to 1 (frame transmission, header reception or wake-up transmission/reception started) causes a wake-up signal to be output from the output pin. The low width of the wake-up signal should be set using the WUTL[3:0] bits in the RLN3nLWUP register. However, if the LWBR0 bit of the RLN3nLWBR register in LIN master mode is 1 (LIN2.x use), the LIN system clock (fLIN) becomes low level width at fa regardless of the setting of the LCKS bit of the RLN3nLMD register. By setting the baud rate to 19200 bps while fa is selected and setting the WUTL[3:0] bit of the RLN3nLWUP register to 0100_B (5 Tbits), 260 µs low width can be output in LIN wake-up mode regardless of the setting of the LCKS bit of the RLN3nLMD register.

If a wake-up low is output without any bit error, the FTC flag in the RLN3nLST register turns 1 (successful frame response or wake-up transmission); when the FTCIE bit in the RLN3nLIE register is 1 (successful frame response/wakeup transmission interrupt enabled), an interrupt request for RLIN3n transmission is generated.

If RLN3nLEDE.BERE is set and a bit error is detected, wake-up transmission is canceled and the BER flag in the RLN3nLEST register is set to 1 (bit error detection).

When RLN3nLEDE.PBERE is set in LIN master mode, RLN3nLEST.PBER flag is set to 1 (physical bus error detection) at the same time of a bit error.

Figure 13.17 shows the wake-up transmission timing.

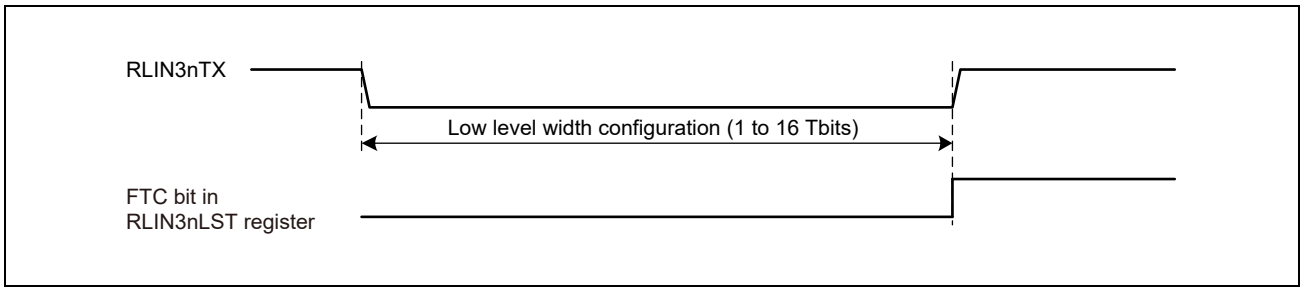


Figure 13.17 Wake-up Transmission Timing

13.7.5.2 Wake-Up Reception

The detection of a wake-up involves the use of an input signal low level width count function. The input signal low level width count function measures the low level width of the input signal to the RLIN3nRX pin, using the same sampling point as data reception. This allows the 2.5-Tbit or longer low-level width of the input signal of fLIN to be measured.

In LIN master mode, by setting the LWBR0 bit in the RLN3nLWBR register, operation is executed without changing the baud rate generator setting at a transition between LIN operation mode and LIN wake-up mode.

When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the RLN3nLWBR register to 0. When LIN Specification Package Revision 2.x is used, set the LWBR0 bit to 1. Setting the LWBR0 bit to 1 selects the LIN system clock (fLIN) to fa regardless of the setting of the LCKS bit in the RLN3nLMD register. (The LCKS bit is not changed). By setting the baud rate to 19200bps while fa is selected, the 130 µs or longer low-level width of the input signal to be measured regardless of the setting of the LCKS bit in the RLN3nLMD register.

When using this function, in LIN wake-up mode set the RFT bit in the RLN3nLDFC register to 0 (LIN master mode: response reception), the RCDS bit in the RLN3nLDFC register to 0 (LIN slave mode: response reception), and then the FTS bit in the RLN3nLTRC register to 1 (frame transmission (header reception) or wake-up transmission/reception started).

When the low level width to be measured is reached, the FRC flag in the RLN3nLST register turns 1 (successful frame response/wake-up reception). If the FRCIE bit in the RLN3nLIE register is 1 (successful frame response or wake-up reception interrupt enabled), an interrupt request for successful RLIN3n reception is generated.

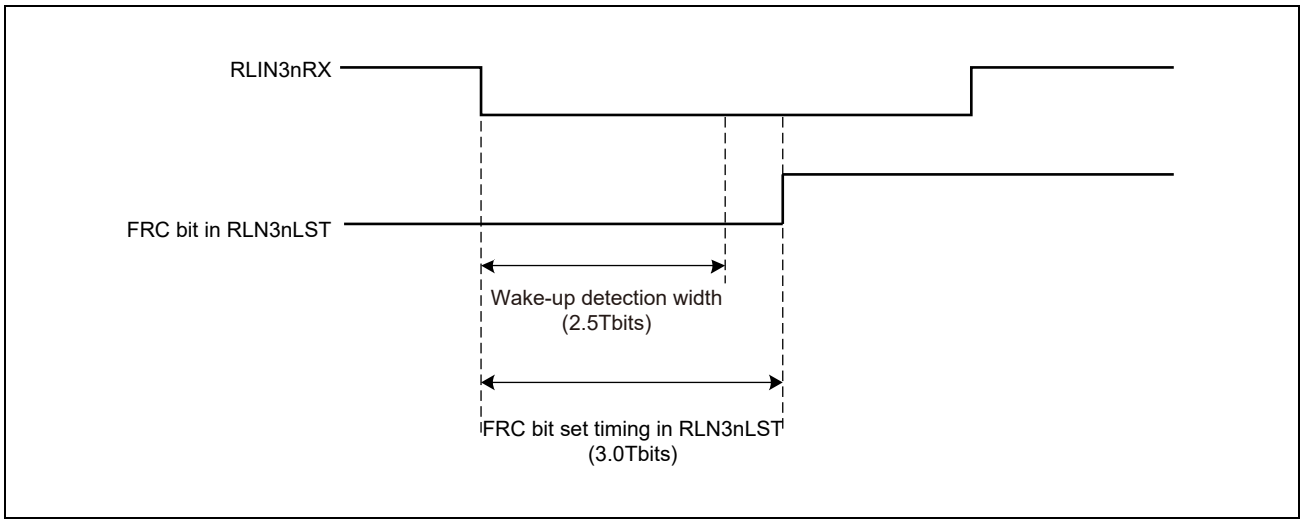


Figure 13.18 Input Signal Low level Count Function

13.7.5.3 Wakeup Collision

If the master node and the slave node transmit wakeup signals simultaneously, a collision will occur on the LIN bus, though a collision of wakeup signals is not detected in the LIN/UART interface.

13.7.6 Status

During LIN mode operation, the LIN/UART interface can detect seven types of statuses.

The four statuses, successful frame/wake-up transmission, successful frame/wake-up reception, error detection, successful header transmission/reception, can generate interrupt requests.

Table 13.84 shows the types of statuses available in LIN master mode. **Table 13.85** lists the types of statuses available in LIN slave mode [auto baud rate] and in LIN slave mode [fixed baud rate].

Table 13.84 Types of Statuses in LIN Master Mode

Status	Status Set Condition	Status Clear Condition	Operation Mode Capable of Status Detection	Corresponding Bit	Interrupt
Reset	After the OM0 bit in the RLN3nLCUC register is set to not-LIN–reset-mode, if actually the LIN/UART interface is cleared from LIN reset mode.	After the OM0 bit in the RLN3nLCUC register is set to LIN reset mode, if actually the LIN/UART interface enters LIN reset mode.	All modes	OMM0 bit in RLN3nLMST register	—
Operation mode	After the OM1 bit in the RLN3nLCUC register is set to LIN operation mode, if actually the LIN/UART interface enters LIN operation mode.	After the OM1 bit in the RLN3nLCUC register is set to LIN wake-up mode, if actually the LIN/UART interface enters LIN wake-up mode.	• LIN operation mode • LIN wake-up mode	OMM1 bit in RLN3nLMST register	—
Frame/ wake-up transmission end	When a frame (header transmission + response transmission), a wake-up signal, or a data group is transmitted successfully.	• When another communication is started (When the FTS bit in the RLN3nLTRC register is set) • When cleared by software • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	FTC flag in RLN3nLST register	√
Frame/wake-up reception end	When a frame (header transmission + response reception), a wake-up signal, or a data group is received successfully.	• When another communication is started (When the FTS bit in the RLN3nLTRC register is set) • When cleared by software • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	FRC flag in RLN3nLST register	√
Error detection	If any of the RPER flag, CSER flag,FER flag, FTER flag, PBER flag, and BER flags in the RLN3nLEST register turns 1 (error detected).	• When another communication is started (When the FTS bit in the RLN3nLTRC register is set) • When cleared by software*1 • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	ERR flag in RLN3nLST register	√
Data 1 reception end	The RFT bit in the RLN3nLDFC register is 0 (reception) and the first byte of the response field or the first byte of each data group is received.*2	• When another communication is started (When the FTS bit in the RLN3nLTRC register is set) • When cleared by software • After transition to LIN reset mode	LIN operation mode	D1RC flag in RLN3nLST register	—
Header transmission end	When a header field is transmitted successfully.	• When another communication is started (When the FTS bit in the RLN3nLTRC register is set) • When cleared by software • After transition to LIN reset mode	LIN operation mode	HTRC flag in RLN3nLST register	√

Note 1. In LIN wake-up mode or LIN operation mode, the ERR flag in the RLN3nLST register is cleared to 0 bywriting 0 to the RPER flag, CSER flag, FER flag, FTER flag, PBER flag, or BER flags in the RLN3nLEST register.

Note 2. Not detected when the RFDL [3:0] bits in the RLN3nLDFC register are 0000_B (0-byte + checksum).

Table 13.85 Types of Statuses in LIN Slave Mode

Status	Status Set Condition	Status Clear Condition	Operation Mode Capable of Status Detection	Corresponding Bit	Interrupt
Reset	After the OM0 bit in the RLN3nLCUC register is set to not-LIN–reset-mode, if actually the LIN/UART interface is cleared from LIN reset mode.	After the OM0 bit of the RLN3nLCUC register is set to LIN reset mode, if actually the LIN/UART interface enters LIN reset mode.	All modes	OMM0 bit in RLN3nLMST register	—
Operation mode	After the OM1 bit in the RLN3nLCUC register is set to LIN operation mode, if actually the LIN/UART interface enters LIN operation mode.	After the OM1 bit in the RLN3nLCUC register is set to LIN wake-up mode, if actually the LIN/UART interface enters LIN wake-up mode.	• LIN operation mode • LIN wake-up mode	OMM1 bit in RLN3nLMST register	—
Frame/ wake-up transmission end	When a response field, a wake-up signal, or a data group is transmitted successfully.	• When cleared by software • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	FTC flag in RLN3nLST register	√
Frame/wake-up reception end	When a response field, a wake-up signal, or a data group is received successfully.	• When cleared by software • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	FRC flag in RLN3nLST register	√
Error detection	If any of the RPER flag, IPER flag, CSER flag, SFER flag, FER flag, TER flag, and BER flag in the RLN3nLEST register turns 1 (error detected).	• When cleared by software*1 • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	ERR flag in RLN3nLST register	√
Data 1 reception end	The RCDS bit in the RLN3nLDFC register is 0 (reception) and the first byte of the response field or the first byte for each data group is received.*2	• When cleared by software • After transition to LIN reset mode	LIN operation mode	D1RC flag in RLN3nLST register	—
Header transmission end	When a header field is received successfully.	• When cleared by software • After transition to LIN reset mode	LIN operation mode	HTRC flag in RLN3nLST register	√

Note 1. In LIN wake-up mode or LIN operation mode, the ERR flag in the RLN3nLST register is cleared to 0 by writing 0 to the RPER flag, IPER flag, CSER flag, SFER flag, FER flag, TER flag, or BER flag in the RLN3nLEST register.

Note 2. Not detected when the RFDL [3:0] bits in the RLN3nLDFC register are 0000_B (0-byte + checksum).

13.7.7 Error Status

13.7.7.1 LIN Master Mode

(1) Types of Error Statuses

The LIN/UART interface can detect six types of error statuses in LIN master mode. The condition of these error statuses can be checked by means of the corresponding bits in the RLN3nLEST register.

All error statuses represent interrupt events.

Table 13.86 shows the types of error statuses.

Table 13.86 Types of Error Statuses in LIN Master Mode

Status	Error Detection Condition	Operation Mode Capable of Error Detection	Communication	Enable/ Disable Detection	Corresponding Bit
Bit error	The transmitted data and the data on the LIN bus monitored by the receive pin do not match *1*2	• LIN operation mode • LIN wake-up mode	Cancel	√	BER flag in RLN3nLEST register
Physical bus error	• LIN bus is detected to be high level when sending a break • LIN bus is detected to be low level when sending a break delimiter • LIN bus is detected to be high level when sending a wake-up	• LIN operation mode • LIN wake-up mode	Cancel	√	PBER flag in RLN3nLEST register
Timeout error	A frame or response transmission/reception does not terminate within a given time*3	LIN operation mode	Cancel	√	FTER flag in RLN3nLEST register
Framing error	In response field reception, a stop bit of each data byte is low level	LIN operation mode	Cancel	√	FER flag in RLN3nLEST register
Checksum error	In response field reception, the result of checksum test gives an error	LIN operation mode	—	×	CSER flag in RLN3nLEST register
Response preparation error	One of the following conditions occurs in frame separate mode during a multi-byte response reception: • The first reception data byte is received after completion of header transmission but before a response transmission/reception request is set • The first reception data byte is received after the completion of previous data group reception before a transmission/reception request for another data group is set.	LIN operation mode	Cancel	×	RPER flag in RLN3nLEST register

Note 1. If a bit error is detected, processing is stopped after a stop bit is sent. If a bit error is detected in a non-dataarea, such as an inter-byte space, the transmission is suspended immediately after the bit which had the error is sent. If a bit error is detected during the transmission of a wake-up, the transmission of the wake-up is canceled after the error-causing bit is sent.

Note 2. In a multi-byte response transmission, bit errors are detected also between data groups.

Note 3. The timeout time depends on the response field data length (the RFDL [3:0] bits in the RLN3nLDFC register) and the checksum selection (the CSM bit in the RLN3nLDFC register), and can be calculated from the following formula.
When the setting of the FSM bit in the RLN3nLDFC register is 1 (i.e., frame separation mode), the timeout time is that for eight bytes until the RTS bit of the RLN3nLTRC register is set. Once the RTS bit is set, the timeout time is re-set to the time based on the response field data length (the RFDL[3:0] bits in the RLN3nLDFC register).

[Frame timeout]

On classic selection (when the LCS bit in RLN3nLDFC is 0): Timeout time = 49 + (number of data bytes + 1) × 14 [Tbit]

On enhanced selection (when the LCS bit in RLN3nLDFC is 1): Timeout time = 48 + (number of data bytes + 1) × 14 [Tbit]

The aforementioned timeout time is a time greater than the TFRAME_MAX of LIN Specification Package Revision 1.3 on classic selection, or the TFRAME_MAX of LIN Specification Package Revision 2.x on enhanced selection.

[Response timeout]

Timeout time = (number of data bytes + 1) × 14 [Tbit]

When en error is detected, time-out error detection function stops.

The error status is cleared when the next communication is started (when the FTS bit in the RLN3nLTRC register is set), by software, or at a transition to LIN reset mode.

(2) Target Time Area for LIN Error Detection

Figure 13.19 shows the time domain in which the LIN/UART interface in LIN master mode performsmonitoring for error detection.

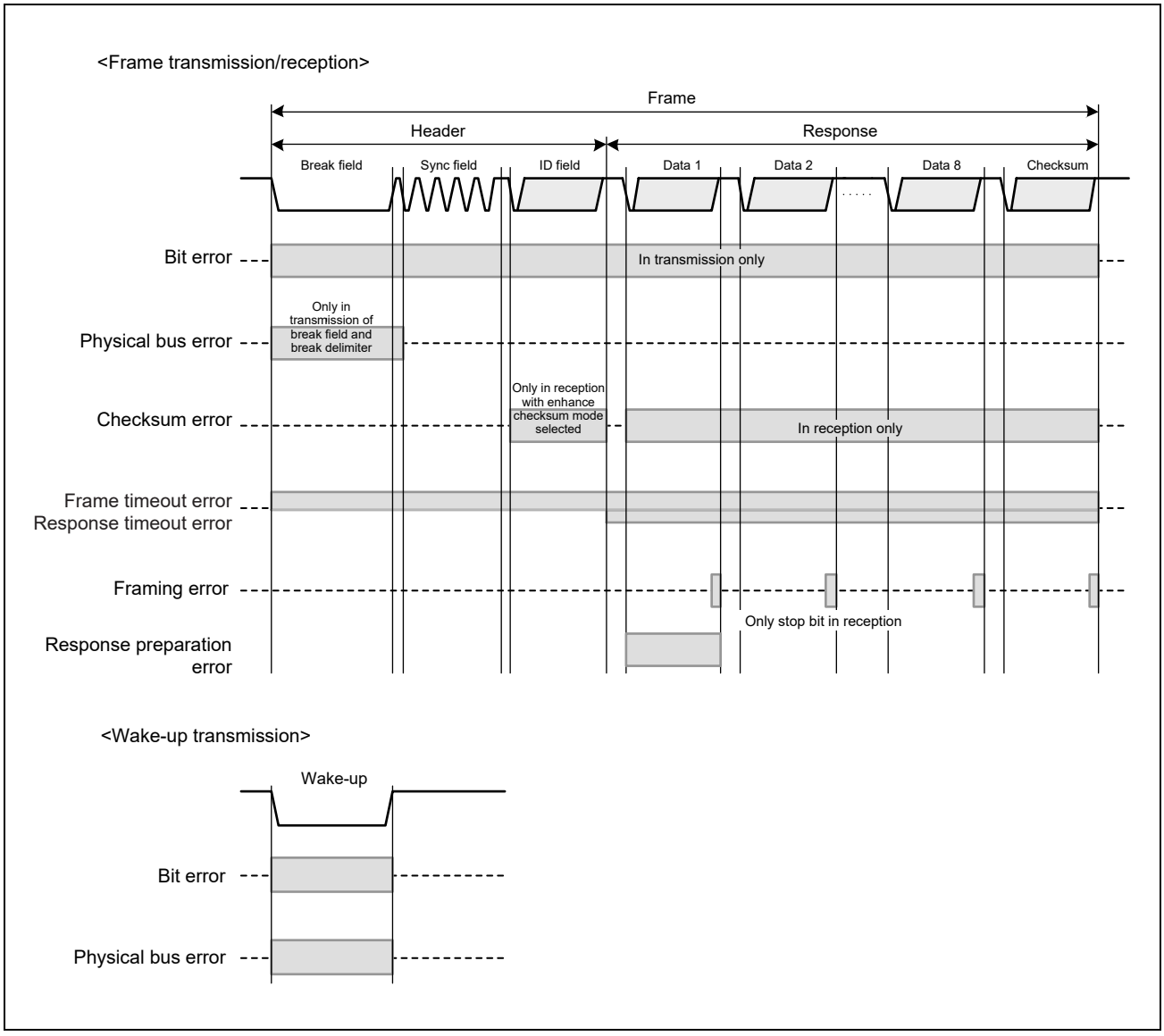


Figure 13.19 Target Time Area for LIN Error Detection (LIN Master Mode)

13.7.7.2 LIN Slave Mode

(1) Types of Error Statuses

The LIN/UART interface can detect seven types of error statuses in LIN slave mode [auto baud rate] or in LIN slave mode [fixed baud rate]. These error statuses can be verified by checking the corresponding bits in the RLN3nLEST register.

Table 13.87 shows the types of error statuses.

Table 13.87 Types of Error Statuses in LIN Slave Mode

Status	Error Detection Condition	Operation Mode Capable of Error Detection	Communication	Enable/ Disable Detection	Corresponding Bit
Bit error	The transmitted data and the data on the LIN bus monitored by the receive pin do not match *1*2	• LIN operation mode • LIN wake-up mode	Cancel	√	BER flag in RLN3nLEST register
Timeout error	A frame or response transmission/reception does not terminate within a given time*3	LIN operation mode	Cancel	√	TER flag in RLN3nLEST register
Framing error	In response field reception, a stop bit of each data byte is low level	LIN operation mode	Cancel	√	FER flag in RLN3nLEST register
Sync field error	If the width of the break low level is greater than the width set by the LBLT bit in the RLN3nLBFC register and the sync field is not 55_H	LIN operation mode	Cancel	√*4	SFER flag in RLN3nLEST register
Checksum error	In response field reception, the result of checksum test gives an error	LIN operation mode	— *5	×	CSER flag in RLN3nLEST register
ID parity error	If the received ID parity bit does not match the value that is automatically calculated by the LIN/UART interface	LIN operation mode	Cancel	√	IPER flag in RLN3nLEST register
Response preparation error	• After the reception of a header, before the first reception data byte is received, response preparation is not made in time. • Before the completion of receiving the first reception data byte for the next data group in a multi-byte response reception, response preparation for the next group is not made in time	LIN operation mode	Cancel	×	RPER flag in RLN3nLEST register

Note 1. If a bit error is detected, processing is stopped after a stop bit is sent. If a bit error is detected in a non-data area, such as an inter-byte space, the transmission is suspended immediately after the bit which had the error is sent. If a bit error is detected during the transmission of a wake-up, the transmission of the wake-up is canceled after the error-causing bit is sent.

Note 2. In a multi-byte response transmission, bit error can be detected between data groups.

Note 3. The timeout time depends on the response field data length (the RFDL [3:0] bits in the RLN3nLDFC register) and the checksum selection (the LCS bit in the RLN3nLDFC register), and this can be calculated according to the following formula. The time-out period until the RTS or LNRR bit of the RLN3nLTRC register is set is 8 data bytes. When the RTS bit is set, the timeout time is reset to the time based on the response field data length (RFDL[3:0] bit of the RLN3nLDFC register). When the LNRR bit is set, the timeout function stops.

[Frame timeout]

On classic selection (when the LCS bit in RLN3nLDFC is 0):

Timeout time = 49 + (number of data bytes + 1) × 14 [Tbit]

On enhanced selection (when the LCS bit in RLN3nLDFC is 1):

Timeout time = 48 + (number of data bytes + 1) × 14 [Tbit]

The aforementioned timeout time is a time greater than the TFRAME_MAX of LIN Specification Package Revision 1.3 on classic selection, or the TFRAME_MAX of LIN Specification Package Revision 2.x on enhanced selection.

[Response timeout]

Timeout time = (number of data bytes + 1) × 14 [Tbit]

When en error is detected, time-out error detection function stops.

Note 4. Only reflection of the result to the SFER flag can be enabled/disabled. Error detection cannot be enabled/disabled.

Note 5. Checksum judgment is performed upon completion of response frame reception. In case of an error, thereceive complete flag is not set to 1.

The error status is cleared by software or at a transition to LIN reset mode.

(2) Target Time Area for LIN Error Detection

Figure 13.20 shows the time domain in which the LIN/UART interface in slave mode performs monitoring for error detection.

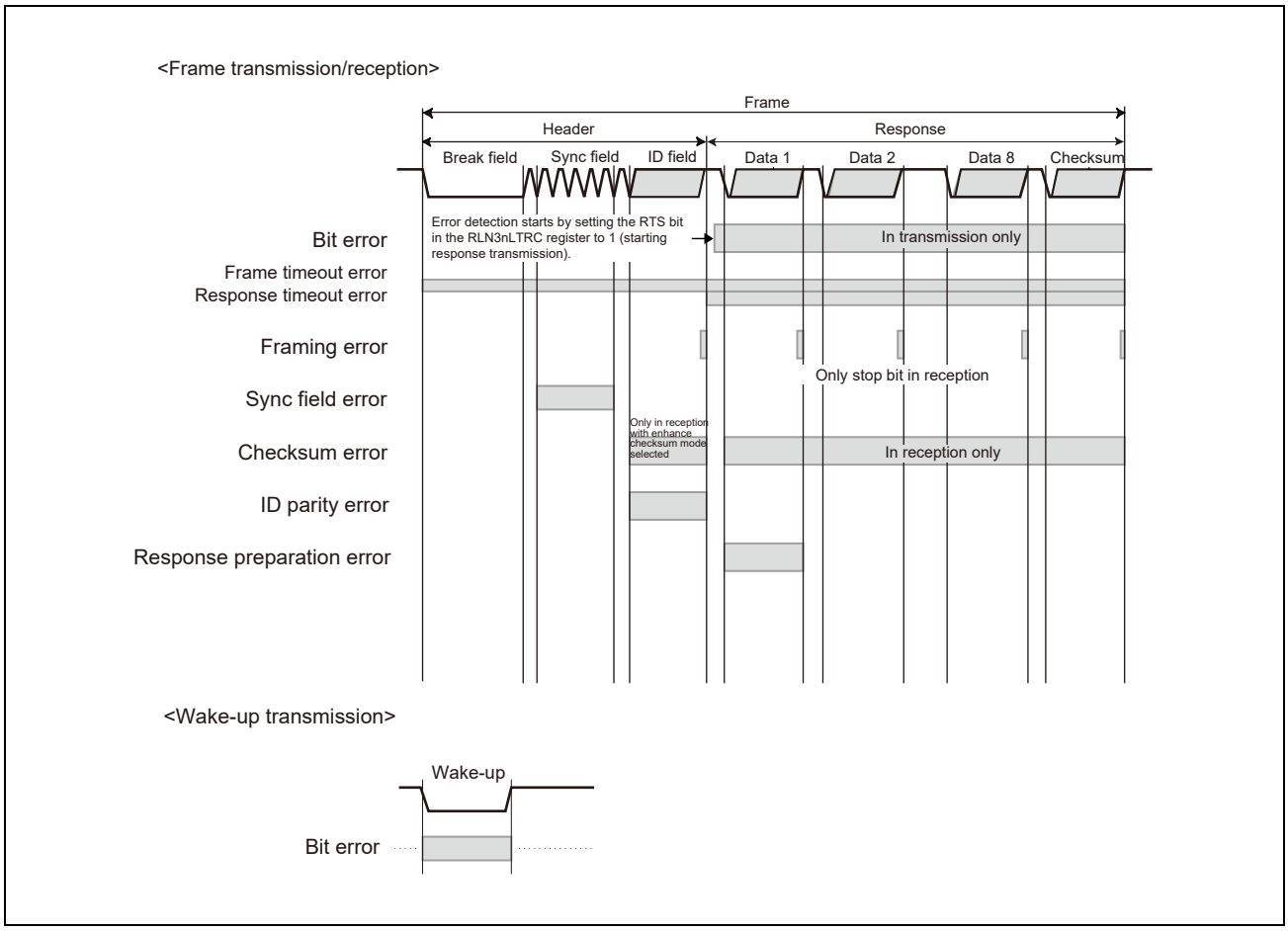


Figure 13.20 Target Time Area for LIN Error Detection (LIN Slave Mode)

13.8 UART Mode

In LIN reset mode, setting the LMD bits in the RLN3nLMD register to 01_B (UART mode) and the OM0 bit in the RLN3nLCUC register to 1 changes the mode to UART mode, turning the OMM0 bit in the RLN3nLMST register to 1.

13.8.1 Transmission

Figure 13.21 shows LIN/UART interface (in UART mode) transmission operations; **Table 13.88** shows LIN/UART interface (in UART mode) transmission processing.

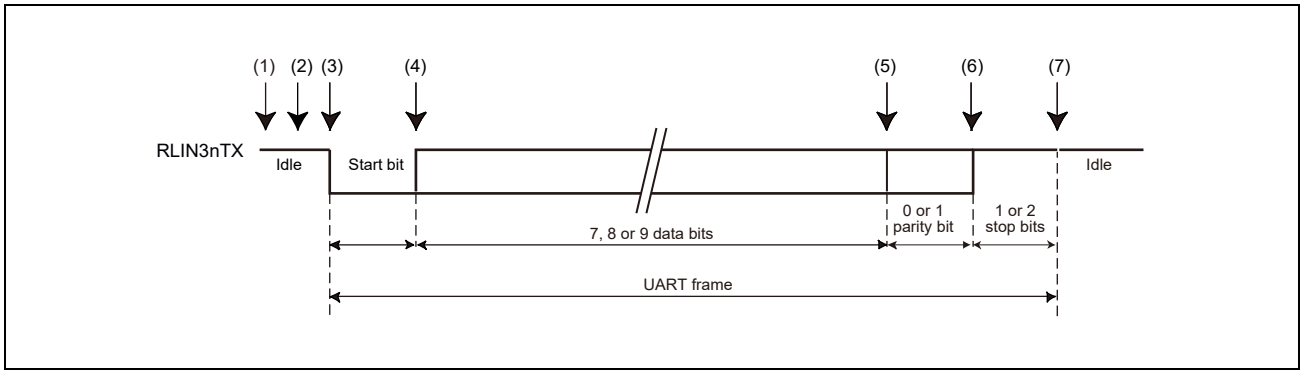


Figure 13.21 LIN/UART Interface (in UART Mode) Transmission Operation

Table 13.88 LIN/UART Interface (UART Mode) Transmission Processing (1/2)

	Software Processing	LIN/UART Interface Processing
(1)	• Sets a baud rate. • Sets noise filter ON/OFF. • Sets error detection enable. • Sets data format. • Sets an interrupt generation timing. • Clears the LIN/UART interface from LIN reset mode. • Sets the transmit enable bit (UTOE bit) to 1.	• Waits for a transmission trigger (RLN3nLUTDR register) by software.
(2)	• Sets the transmit data to the UART transmit dataregister (RLN3nLUTDR) or UART wait transmit data register (RLN3nLUWTDR).	• Sets the transmit status flag.
(3)	• Waits an interrupt request. [When the UTIGTS bit is 0 (a transmission interrupt request is generated upon start of transmission)]	• Transmits a start bit (for switching between transmission and reception in half duplex communication, transmits a start bit after receiving 1 stop bit. For details about this function, see **Section 13.8.1.4, Transmission Start Wait Function**. [When the UTIGTS bit is 0 (a transmission interrupt is generated)] • Outputs a transmission interrupt.
(4)	• When transmitting data continuously, setsanother piece of transmission data in the UART transmit data register (RLN3nLUTDR register), waits for the generation of an interrupt request.	Transmits the data set in the UART (for wait) transmit data register.
(5)		Transmits a parity bit when parity is used.
(6)		Transmits 1 or 2 stop bits.

Table 13.88 LIN/UART Interface (UART Mode) Transmission Processing (2/2)

	Software Processing	LIN/UART Interface Processing
(7)	[When the UTIGTS bit is 0 (a transmission interrupt request is generated upon start of transmission)] • If another piece of transmission data is set, goes to step (3).	[When the UTIGTS bit is 0 (a transmission interrupt request is generated upon start of transmission)] • If another piece of transmission data is set, goes to step (3). • If another piece of transmission data is not set, clears the transmit status flag.
	[When the UTIGTS bit is 1 (a transmission interrupt is output upon end of transmission)] • When transmitting data continuously, goes to step (2).	[When the UTIGTS bit is 1 (a transmission interrupt is output upon end of transmission)] • Generates a RLIN3n transmission interrupt request. • Clears the transmission status flag.

13.8.1.1 Continuous Transmission

The LIN/UART interface (in UART mode) can transmit multiple sets of data continuously by using the RLN3nLUTDR register. **Figure 13.22** shows an operation example where the transmission interrupt generation timing is the start of transmission and an operation example where the transmission interrupt generation timing is the end of transmission.

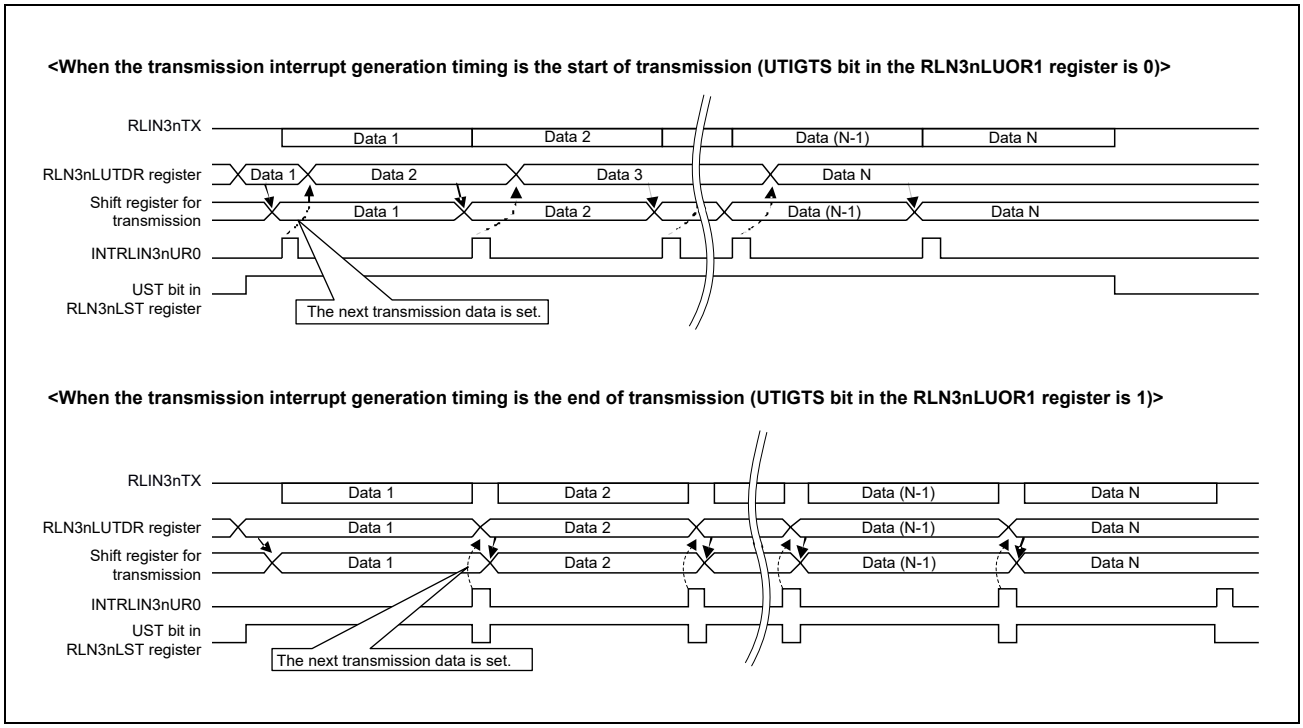


Figure 13.22 Operation Example of LIN/UART Interface (UART Mode) Continuous Transmission

An interrupt can be generated at the end of a transmission by changing the UTIGTS bit in the RLN3nLUOR1 register from 0 to 1 after the start of transmission of final data, provided only that the transmission interrupt generation timing is the start of transmission and the end of transmission of final data needs to be known.

13.8.1.2 UART Buffer Transmission

The LIN/UART interface (in UART mode) has a maximum of nine bytes of UART buffers, and thus it is capable of performing continuous transmissions through the use of UART buffers.

Figure 13.23 shows the UART buffer transmission operation in the LIN/UART interface (in UART mode). **Table 13.89** shows the UART buffer transmission processing.

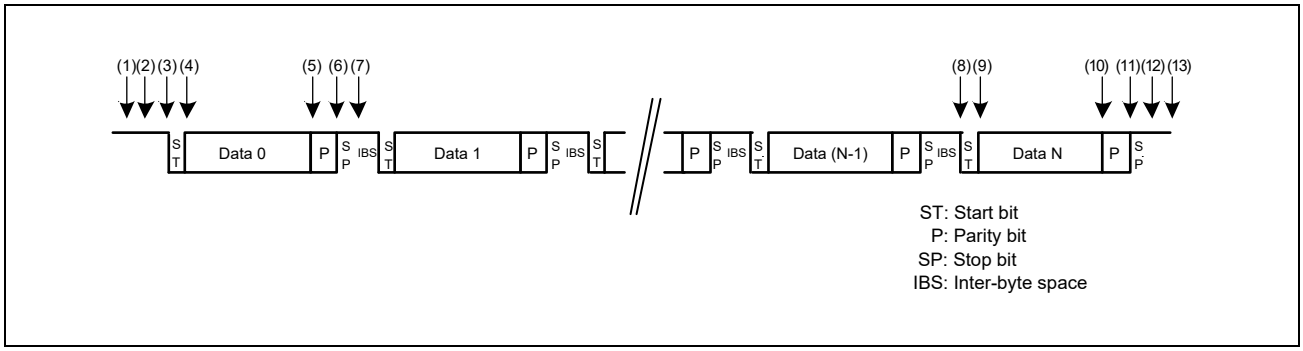


Figure 13.23 UART Buffer Transmission in LIN/UART Interface (in UART Mode)

Table 13.89 UART Buffer Transmission Processing in LIN/UART Interface (in UART Mode) (1/2)

	Software Processing	LIN/UART Interface Processing
(1)	• Sets a baud rate • Sets noise filter ON/OFF • Sets error detection enable • Sets data format • Sets an interrupt generation timing to the end of transmission. • Clears the LIN/UART interface from LIN reset mode. • Sets the transmit enable bit (UTOE bit) to 1	• Waits for a transmission trigger (RTS bit) by software
(2)	• Sets the UART buffer data length and whether the system must wait for the start of transmission. • Sets the transmission data in the UART data 0buffer register (RLN3nLUDB0) and the LIN data buffer m register (RLN3nLDBRb). (b =1 to 8) • Sets the UART buffer transmission start bit (RTS).	• Sets the transmit status flag.
(3)	Waits for an interrupt request.	Transmits a start bit. (When switching from reception to transmission during half-duplex communication, transmits the start bit upon completion of the stop bit for reception. For details about this function, see **Section 13.8.1.4, Transmission Start Wait Function**.
(4)		Transmits the data set in the UART data buffer 0 register (RLN3nLUDB0) and the LIN/UART data buffer b register (RLN3nLDBRb).
(5)		Transmits a parity bit when parity is used.
(6)		Transmits 1 or 2 stop bits (When the number of data set in UART buffer data length select bits, proceed to (12).)
(7)		Transmits an inter-byte space (idle). Repeats steps (3) to (7) until the number of data –1 that was set in the UART buffer data length select bits is reached.
(8)		Transmits a start bit.

Table 13.89 UART Buffer Transmission Processing in LIN/UART Interface (in UART Mode) (2/2)

	Software Processing	LIN/UART Interface Processing
(9)		Transmits the data set in the LIN/UART data buffer b register (RLN3nLDBRb).
(10)		Transmits a parity bit when parity is used.
(11)		Transmits 1 or 2 stop bits.
(12)		• Sets the buffer transmission end flag. • Clears the UART buffer transmit start bit (RTS). • A transmission interrupt request signal. • Clears the transmission status flag.
(13)	• Checks the RLN3nLST register, and clears flags • In the case of continuous data transmission, goes to step (2).	

(1) UART Buffer Transmission

For a 9-byte transmission, the contents stored in the RLN3nLUDB0 and RLN3nLDBR1 to RLN3nLDBR8 registers are transmitted to data areas 0 to 8. The RLN3nLUDB0 register is used only if 9-byte transmission is set. In other cases, the RLN3nLDBR1 to RLN3nLDBR8 registers are selected depending upon the length of data involved. For a 4-byte transmission, the contents stored in the RLN3nLDBR1 to RLN3nLDBR4 registers are transmitted to data areas 1 to 4, but the contents of the RLN3nLDBR5 to RLN3nLDBR8 registers are not transmitted. An RLIN3n transmission interrupt is generated after the transmission of the data that is set in the MDL [3:0] bits of the RLN3nLDFC register. The spaces between transmission data items can be set in the IBS bit in the RLN3nLSC register.

Figure 13.24 shows a 9-byte UART buffer and the transmission processing.

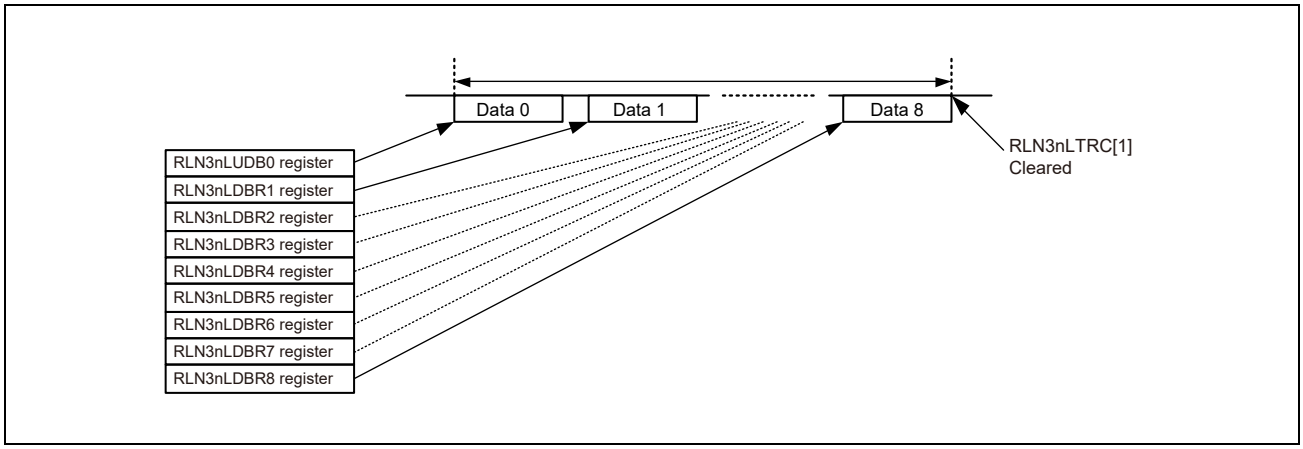


Figure 13.24 UART Buffer and Transmission Processing (for 9-Byte Transmission)

13.8.1.3 Data Transmission

One bit of data is transmitted per Tbit.

In half-duplex communication, if the BERE bit in the RLN3nLEDE register is 1 (bit error detection enabled), the transmission data and the input pin level are compared bit by bit during data transmission, and the results are stored in the BER flag in the RLN3nLEST register (see **Section 13.8.5, Error Status**). The timing at which the input pin is sampled during data transmission can vary depending upon the settings of the LPRS[2:0] and NSPB[3:0] bits in the RLN3nLWBR register.

The bit error detection timing in UART mode is shown in **Table 13.90**.

Table 13.90 Error Detection Timing in UART Mode

Sampling Count Per Bit	Bit Error Detection Timing
6 samples	3rd clock cycle + 1 prescaler clock
7 samples	4th clock cycle + 1 prescaler clock
8 samples	4th clock cycle + 1 prescaler clock
9 samples	5th clock cycle + 1 prescaler clock
10 samples	5th clock cycle + 1 prescaler clock
11 samples	6th clock cycle + 1 prescaler clock
12 samples	6th clock cycle + 1 prescaler clock
13 samples	7th clock cycle + 1 prescaler clock
14 samples	7th clock cycle + 1 prescaler clock
15 samples	8th clock cycle + 1 prescaler clock
16 samples	8th clock cycle + 1 prescaler clock

Example of Data Transmission Timing (when 1 Tbit = 16 Sampling) is shown in **Figure 13.25**.

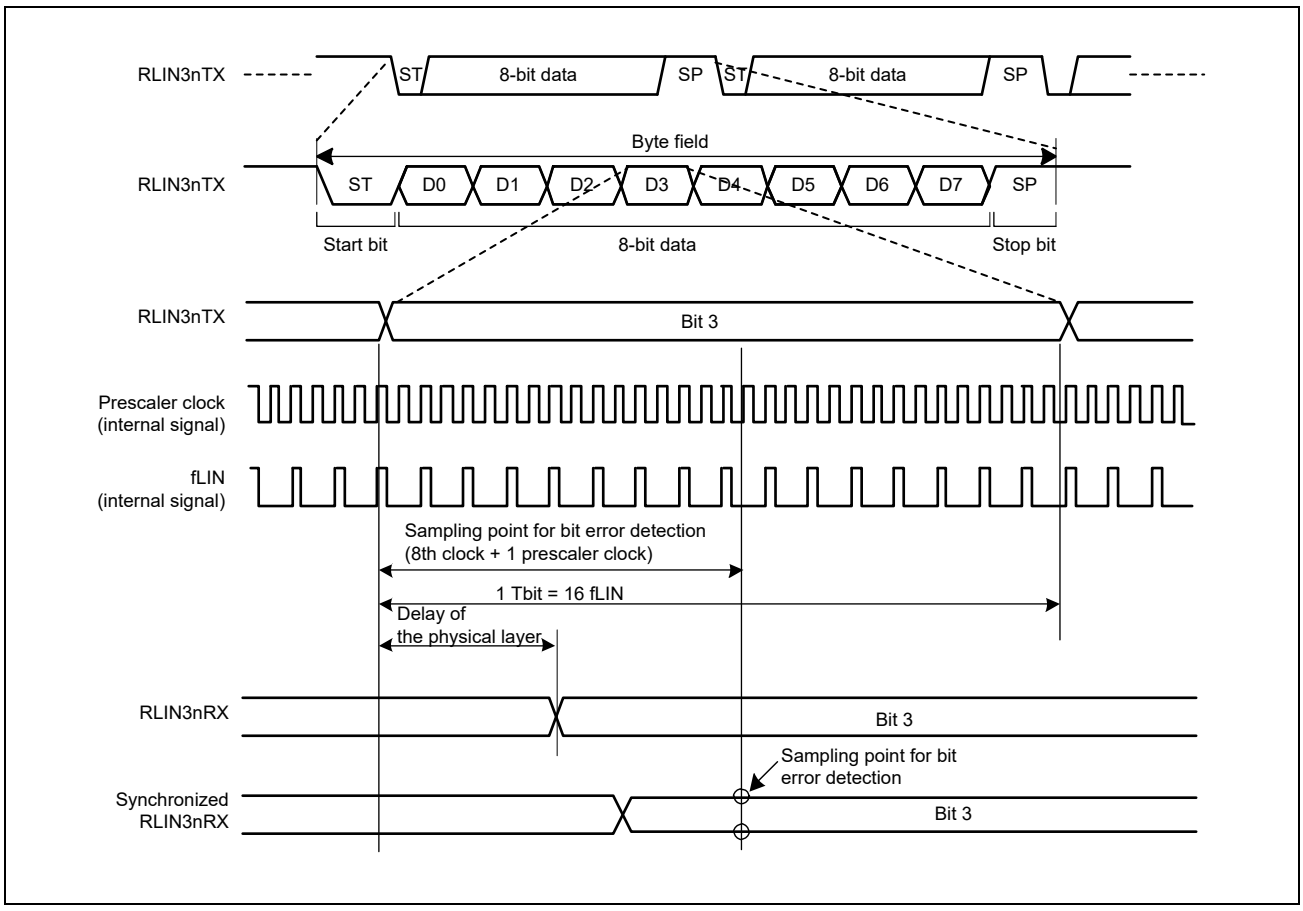


Figure 13.25 Example of Data Transmission Timing (When 1 Tbit = 16 Sampling)

13.8.1.4 Transmission Start Wait Function

For performing half-duplex communication, the LIN/UART interface (in UART mode) has the function of securing the reception stop bit length when switching from reception to transmission.

If it is desired to delay the start of transmission until the stop bits for the reception are completed, set data in the RLN3nLUWTDR register, which is used only for the wait function, instead of setting transmission data in the RLN3nLUTDR register as a start-of-transmission request. When transmitting from the UART buffer, set 1 (UART buffer transmission started) in the RTS bit in the RLN3nLTRC register with 1 set in the UTSW bit in the RLN3nLDFC register.

In such a case, the LIN/UART interface delays the start of transmission until the stop bits of reception data are completed.

It should be noted that even if the UART stop bit length select bit (USBLS) in RLN3nLBFC register is 1 (stop bits = 2 bits), delay is made only for 1 bits.

Figure 13.26 shows the operation of transmission wait function.

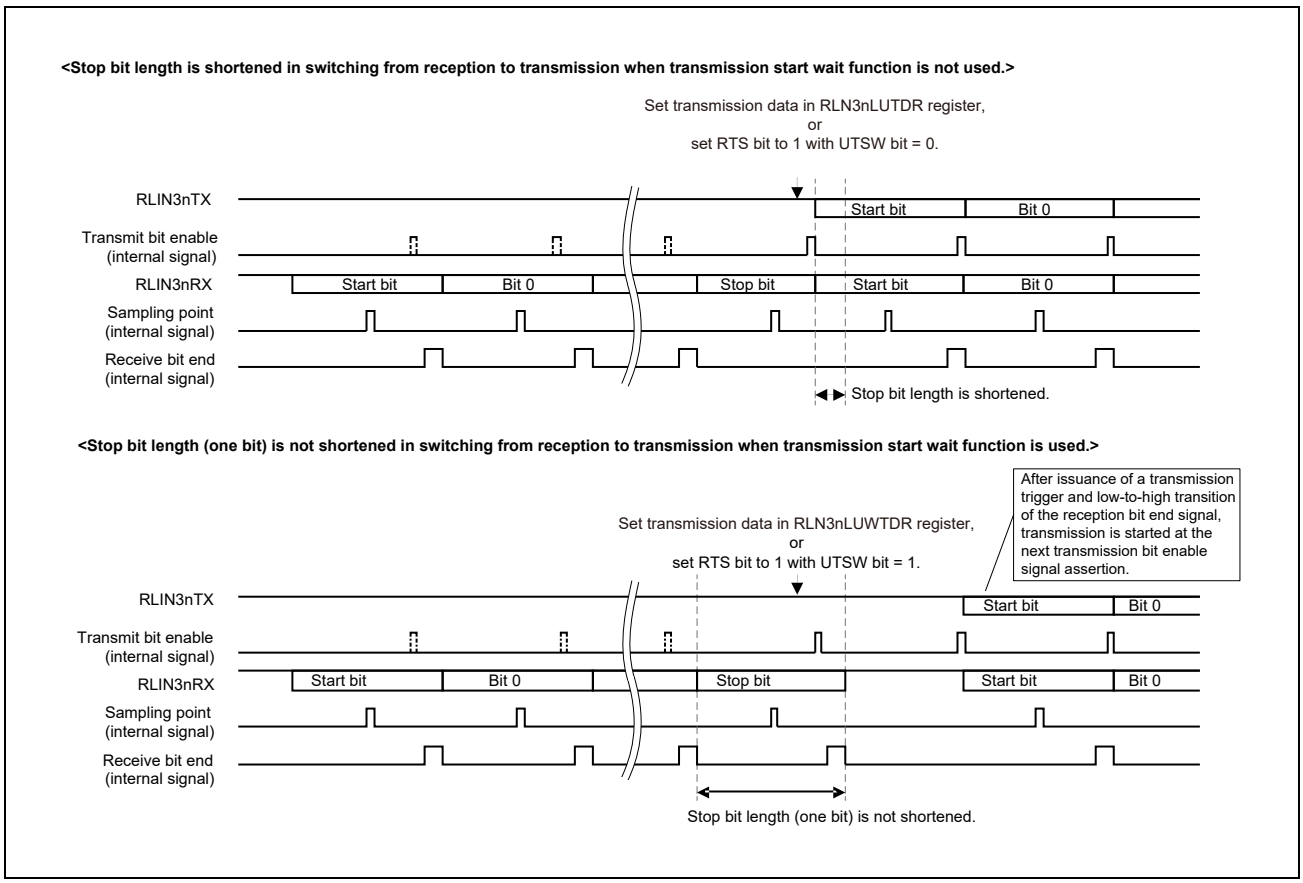


Figure 13.26 Transmission Wait Function (If Transmission Data is Set during the Stop Bits in the Received Data)

13.8.2 Reception

Figure 13.27 shows the LIN/UART interface (in UART mode) reception operation. **Table 13.91** shows the LIN/UART interface (in UART mode) reception processing.

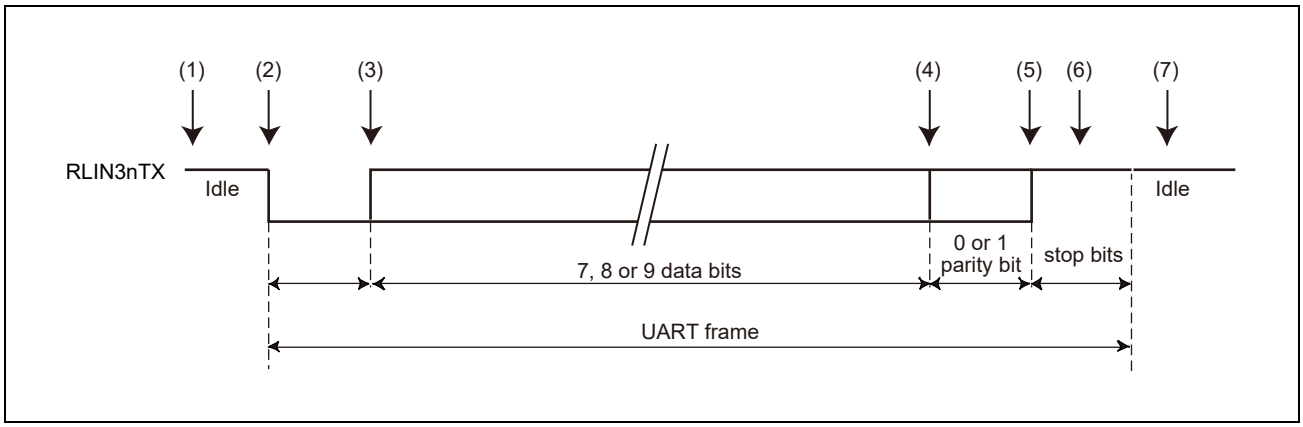


Figure 13.27 LIN/UART Interface (in UART Mode) Reception Operation

Table 13.91 LIN/UART Interface (in UART Mode) Reception Processing

	Software Processing	LIN/UART Interface Processing
(1)	• Sets a baud rate. • Sets noise filter ON/OFF. • Sets error detection enable. • Sets data format. • Clears the LIN/UART interface from LIN reset mode. • Sets the receive enable bit (UROE bit) to1.	• Waits for reception enable state switching by software. • Waits for detection of a start bit.
(2)	Waits for an interrupt request.	• Waits for a falling edge from the reception pin, and detects a start bit. • Sets the reception status flag.
(3)		Receives data.
(4)		Receives a parity bit when parity is used.
(5)		Receives only 1 stop bit.
(6)		• Generates a successful RLIN3n reception interrupt request. • Clears the reception status flag.
(7)	Checks the RLN3nLST register, and clears flags	Waits for a falling edge from the reception pin.

13.8.2.1 Data Reception

Data reception is performed by using the synchronized RLIN3nRX (an internal signal) that is the input from the RLIN3nRX pin synchronized with the prescaler clock.

The byte field is synchronized at the falling edge of the start bit for the synchronized RLIN3nRX signal. After the falling edge is detected, resampling is performed 0.5 Tbits later when the number of the sampling per 1 Tbit is even and {(the number of the sampling + 1) / 2} / (the number of sampling) Tbits later when the number is odd. If the synchronized RLIN3nRX signal is low level, the bit is recognized as a start bit. The bit is not recognized as a start bit if the RLIN3nRX signal is fixed at low level after the reset is cleared or if a high level is detected during the resampling.

After the start bit is detected, 1 bit is sampled per Tbit. However, when the BERE bit in the RLN3nLEDE register is 1, the sampling point is the same as the bit error detection timing.

The LIN/UART interface has a noise filter function with respect to received data. If the LRDNFS bit in the RLN3nLMD register is 0, the noise filter is used. For a sampling value, the value determined by a 3-sampling majority rule by the prescaler clock is used. If the LRDNFS bit in the RLN3nLMD register is 1, the noise filter is not used. In this case, for a sampling value, the synchronized RLIN3nRX value at the sampling position is used as is.

Figure 13.28 shows an example of data reception timing.

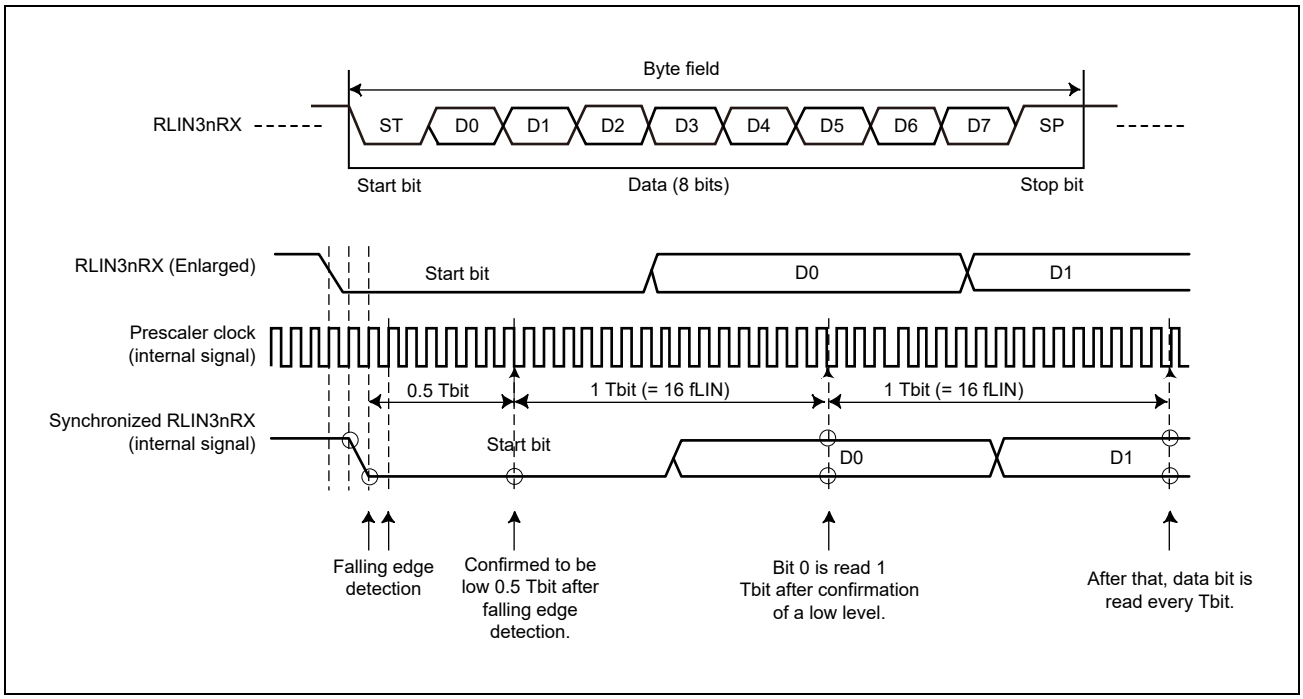


Figure 13.28 Example of Data Reception Timing (When Sampling Count is 16 in 1 Tbit)

13.8.3 Expansion Bits

The LIN/UART interface (in UART mode) can transmit and receive 9-bit long data by setting the UEBE bit in the RLN3nLUOR1 register to 1.

13.8.3.1 Expansion Bit Transmission

The LIN/UART interface (in UART mode) can transmit 9-bit long data when the expansion bit enable bit (UEBE) in the UART option register 1 (RLN3nLUOR1) is 1 and by writing the 9-bit data to either the UART transmission data register (RLN3nLUTDR) or the UART wait transmission data register (RLN3nLUWTDR).

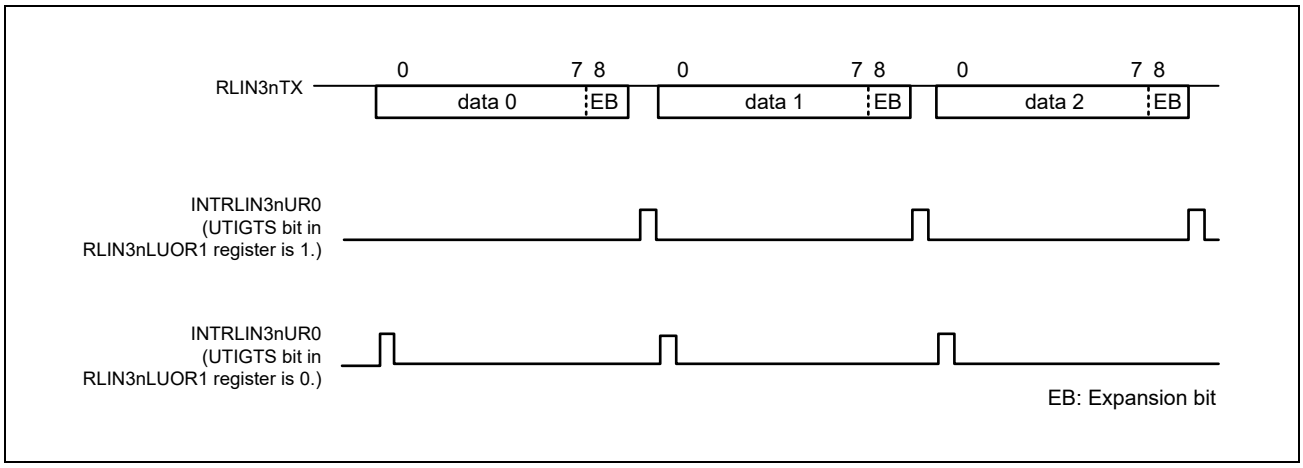


Figure 13.29 Transmission Example When Expansion Bit is Enabled (LSB First)

13.8.3.2 Expansion Bit Reception

With the LIN/UART interface (in UART mode), 9-bit data can always be received without requiring a comparison of expansion bits, provided that the expansion bit enable bit (UEBE) in the UART option register 1 (RLN3nLUOR1) is 1, the expansion bit comparison disable bit (UECD) is 1, and the expansion bit data comparison enable bit (UEBDCE) is 0. Irrespective of the particular setting of the expansion bit detection level select bit (UEBDL) in the UART option register 1 (RLN3nLUOR1), a successful RLIN3n reception interrupt is generated when 9-bit data is received.

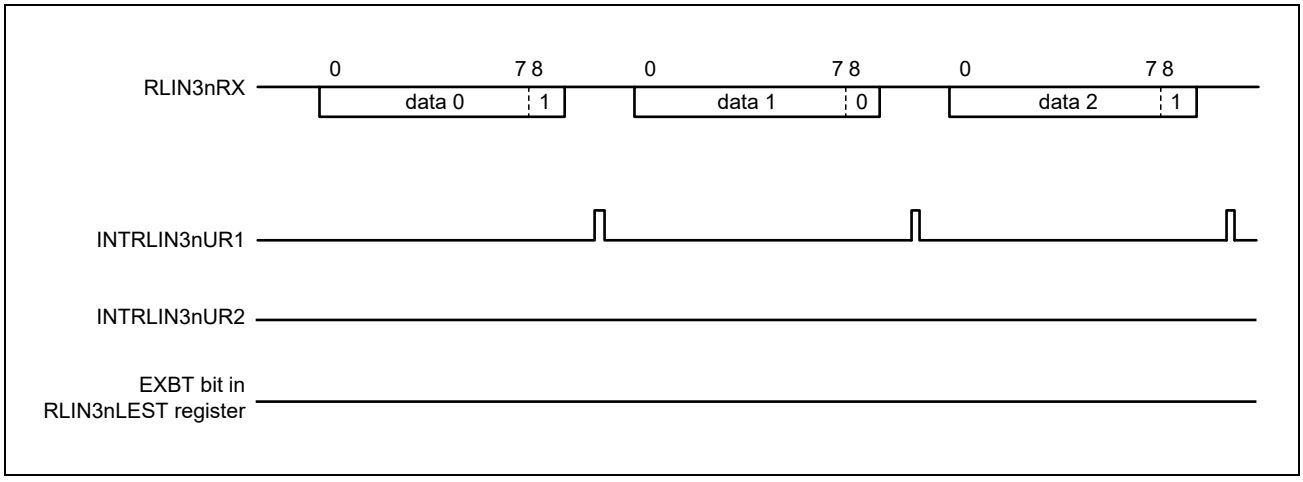


Figure 13.30 Expansion Bit Reception Example (LSB First)

13.8.3.3 Expansion Bit Reception (with Expansion Bit Comparison)

The LIN/UART interface (in UART mode) can compare received expansion bits and the UEBDL bits when the expansion bit enable bit (UEBE) in the UART option register 1 (RLN3nLUOR1) is 1 and the expansion bit comparison disable bit (UECD) is 0 and the expansion bit/data comparison enable bit (UEBDCE) is 0.

If the level that was set in the expansion bit detection level select bit (UEBDL) is detected, an RLIN3n status interrupt request is generated upon completion of data reception, and the expansion bit detection flag (EXBT) in the LIN error status register (RLN3nLEST) is set. If the reversed value of an expansion bit detection level is detected, a successful RLIN3n reception interrupt request is generated. In either case, the received data is stored in the UART reception data register (RLN3nLURDR), unless there was an overrun error.

Figure 13.31 shows an example when the expansion bit detection level select bit (UEBDL) is set to 0.

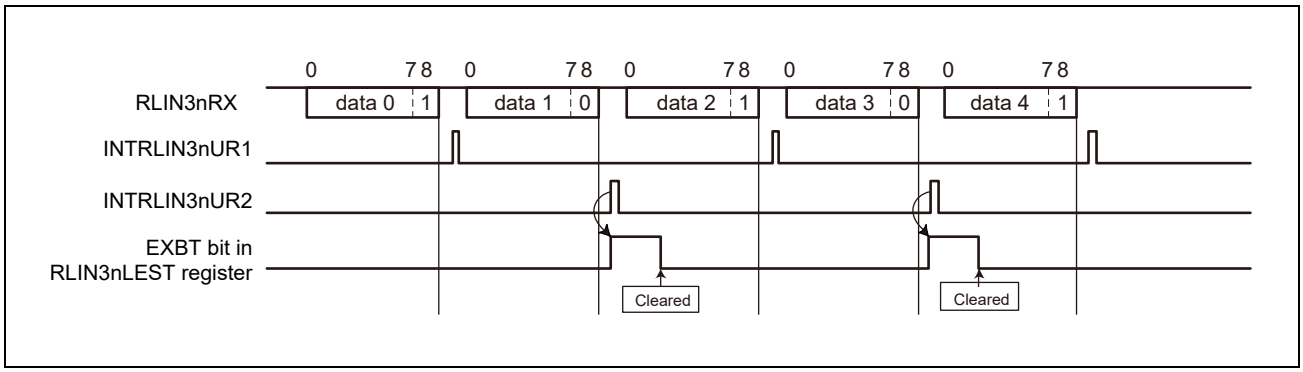


Figure 13.31 Expansion Bit Reception Example (with Expansion Bit Comparison)(LSB First, UEBDL = 0)

NOTE

- If a reception error (parity error, framing error, or overrun error) occurs in received data 0, 2, or 4 (if a reversed value of an expansion bit detection level is detected), an RLIN3n status interrupt is generated, and the error flag is updated. In this case, a successful RLIN3n reception interrupt is not generated.
- If a reception error (parity error, framing error, or overrun error) occurs in received data 1 or 3 (if an expansion bit detection level is detected), an RLIN3n status interrupt is generated, and the error flag is updated. If the overrun error occurs, the expansion bit detection flag (EXBT) is also set.

13.8.3.4 Expansion Bit Reception (with Data Comparison)

If the expansion bit enable bit (UEBE) in the UART option register 1 (RLN3nLUOR1) is 1 and the expansion bit comparison disable bit (UECD) is 0 and the expansion bit/data comparison enable bit (UEBDCE) is 1, and if the level that was set by the expansion bit detection level select bit (RLN3nUEBDL) is detected, the LIN/UART interface compares the 8 bits, exclusive of the expansion bit in the received data, with the a pre-set RLN3nLIDB register value.

If the result of the comparison is a match, the LIN/UART interface performs the following operations:

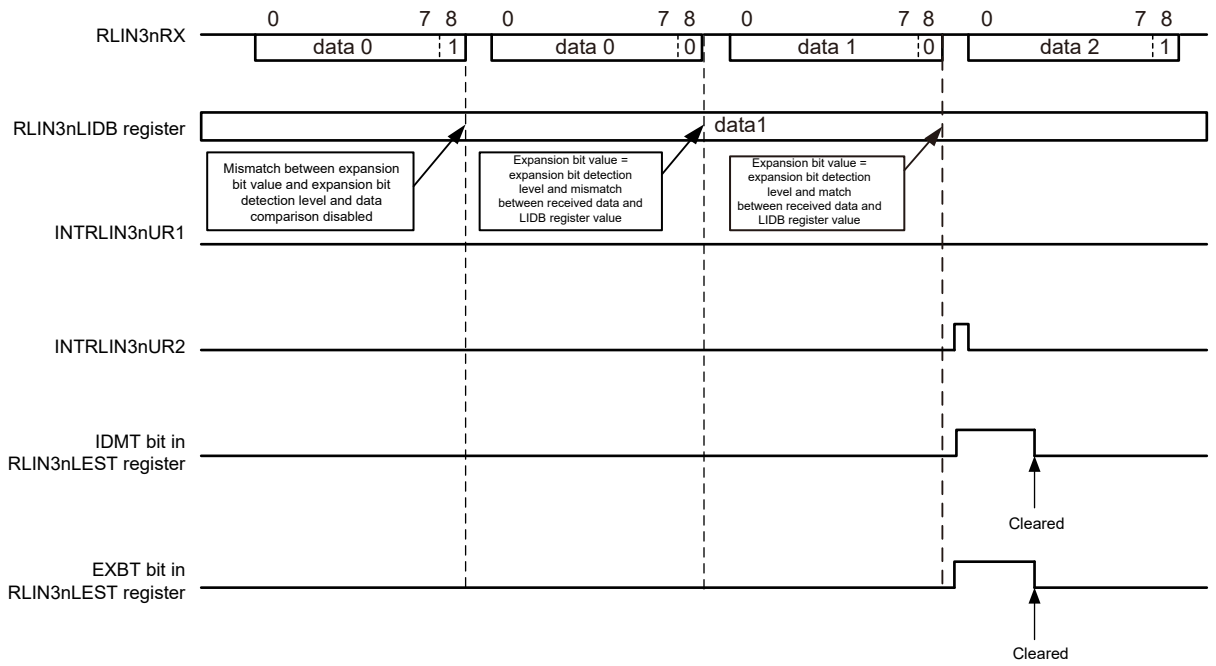
- Generates an RLIN3n status interrupt
- Sets an expansion bit detection flag (EXBT)
- Sets an ID match flag (IDMT)
- Stores the received data in the UART reception data register (RLN3nLURDR)

Even when the result of the comparison is a match, a successful RLIN3n reception interrupt is not generated.

If the result of the comparison is not a match, no successful RLIN3n reception interrupt or RLIN3n status interrupt is generated, and the EXBT and IDMT flags are not set to 1. The received data is stored in the UART reception data register (RLN3nLURDR).

When changing the UEBDCE bit to 0, make the change before the reception of another set of data is finished.

Figure 13.32 shows an example when the expansion bit detection level select bit (UEBDL) is set to 0.



Note: If a reception error (parity error, framing error, or overrun error) occurs, an RLIN3n status interrupt is generated, and the error flag is updated. If the overrun error occurs and the result of comparison is a match, the EXBT and IDMT are also set to 1.

Figure 13.32 Expansion Bit Reception Example (with Data Comparison) (LSB First, UEBDL = 0)

13.8.4 Status

In UART mode, the LIN/UART interface can detect five types of statuses.

Two statuses, successful UART buffer transmission and error detection, can generate interrupt requests.

Table 13.92 shows the types of statuses available in UART mode.

Table 13.92 Types of Statuses in UART Mode

Status	Status Set Condition	Status Clear Condition	Corresponding Bit	Interrupt
Reset	After the OM0 bit in the RLN3nLCUC register is set to not-LIN–reset-mode, if actually the LIN/UART interface is cleared from LIN reset mode.	After the OM0 bit in the RLN3nLCUC register is set to LIN reset mode, if actually the LIN/UART interface enters LIN reset mode.	OMM0 bit in RLN3nLMST register	—
Successful UART buffer transmission	• When the UTIGTS bit in the RLN3nLUOR1 register is 0 (transmission interrupt request is generated upon start of transmission), the transmission of the last data of the data length set by the MDL bit in the RLN3nLDFC register is started. • When the UTIGTS bit in the RLN3nLUOR1 register is 1 (transmission interrupt request is generated upon end of transmission), the transmission of the data length set by the MDL bit in the RLN3nLDFC register is ended.	• When cleared by software • After transition to LIN reset mode	FTC flag in RLN3nLST register	√
Error detection	If any of the UPER flag, IDMT flag, EXBT flag, FER flag, OER flag, and BER flags in the RLN3nLEST register turns 1 (error detected).	• When cleared by software*1 • After transition to LIN reset mode	ERR flag in RLN3nLST register	√
Transmission status	• When data is written to the RLN3nLUTDR or RLN3nLUWTDR register. • When a 1 is written to the RTS bit in the RLN3nLTRC register.	• The transmission of the data set in the RLN3nLUTDR or RLN3nLUWTDR register is complete, but another transmission data item is not set • The transmission of the data in the UART buffer is complete, and the RTS bit in the RLN3nLTRC register is cleared • After transition to LIN reset mode	UTS flag in RLN3nLST register	—
Reception status	• When a start bit is detected.	• When a sampling point for stop bits is detected • After transition to LIN reset mode	URS flag in RLN3nLST register	—

Note 1. Writing a 0 to the UPER, IDMT, EXBT, FER, OER, and BER flags in the RLN3nLEST register when the LIN reset mode is being canceled turns the ERR flag in the RLN3nLST register to 0.

13.8.5 Error Status

Types of Error Statuses

In UART mode, the LIN/UART interface can detect four types of errors and two types of statuses. The condition of these statuses can be checked by means of the corresponding bits in the RLN3nLEST register.

Table 13.93 lists applicable status types.

Table 13.93 Types of Statuses in UART Mode

Status	Error Detection Condition	Communication	Enable/ Disable Detection	Corresponding Bit
Bit error	The transmitted data and the data monitored on the receive pin do not match*1	Continues until the transmission of the set transmission data is finished.	√	BER flag in RLN3nLEST register
Overrun error	After received data is stored in the RLN3nLURDR register, another data item is received before the data is read. (In this case, no data is stored in the RLN3nLURDR register).	— (Reception is finished by the time this error is detected)	√	OER flag in RLN3nLEST register
Framing error	When the first stop bit is low level in the reception processing.	— (Reception is finished by the time this error is detected)	√	FER flag in RLN3nLEST register
Parity error	The received parity value fails to match the parity value calculated from the received data	(Reception is finished by the time this error is detected)	×*2	UPER flag in RLN3nLEST register
Expansion bit detection	The value of the received expansion bit matches the value of the UEBDL bit in the RLN3nLUOR1 register.	—	√	EXBT flag in RLN3nLEST register
ID match detection	The value of the received expansion bit matches the value of the UEBDL bit in the RLN3nLUOR1 register and the 8-bit receive data excluding the expansion bit matches the value of the RLN3nLIDB register.	—	√	IDMT flag in RLN3nLEST register

Note 1. In the case of transmission from the UART buffer, bit errors are detected even in the space between UARTframes (inter-byte space).

Note 2. Setting the UPS[1:0] bits in the RLN3nLBFC register to 10_B (0 parity) disables the checking of parity bit values. In this case, no parity error is generated.

The error status is cleared by software or at a transition to LIN reset mode.

13.9 LIN Self-Test Mode

The LIN/UART interface has the LIN self-test mode. When the LIN/UART interface enters the LIN self-test mode, RLIN3nTX and RLIN3nRX are disconnected from external pins and RLIN3nTX and RLIN3nRX are connected in the LIN/UART interface. Therefore, the frame transmitted from RLIN3nTX is looped back to RLIN3nRX. The LIN self-test mode can perform tests exclusively in LIN mode.

The self-test can be performed in the following four types.

- LIN master self-test mode (transmission): Header transmission and response transmission
- LIN master self-test mode (reception): Header transmission and response reception
- LIN slave self-test mode (transmission): Header reception and response transmission
- LIN slave self-test mode (reception): Header reception and response reception

In LIN self-test mode, the operate is at the fastest baud rate, regardless of the setting of the baud rate generator.

Regardless of the setting of the baud rate related registers, the baud rate operates at the LIN communication clock source/16 [bps]. (The NSPB bits in the RLN3nLWBR register should be set to 0000_B or 1111_B.)

In addition, in LIN self-test mode, the following functions are not supported.

- LIN wake-up mode
- Frame separate mode
- Multi-byte response transmission/reception
- LIN slave mode (Auto baud rate)
- Frame/response timeout error

Do not use these functions.

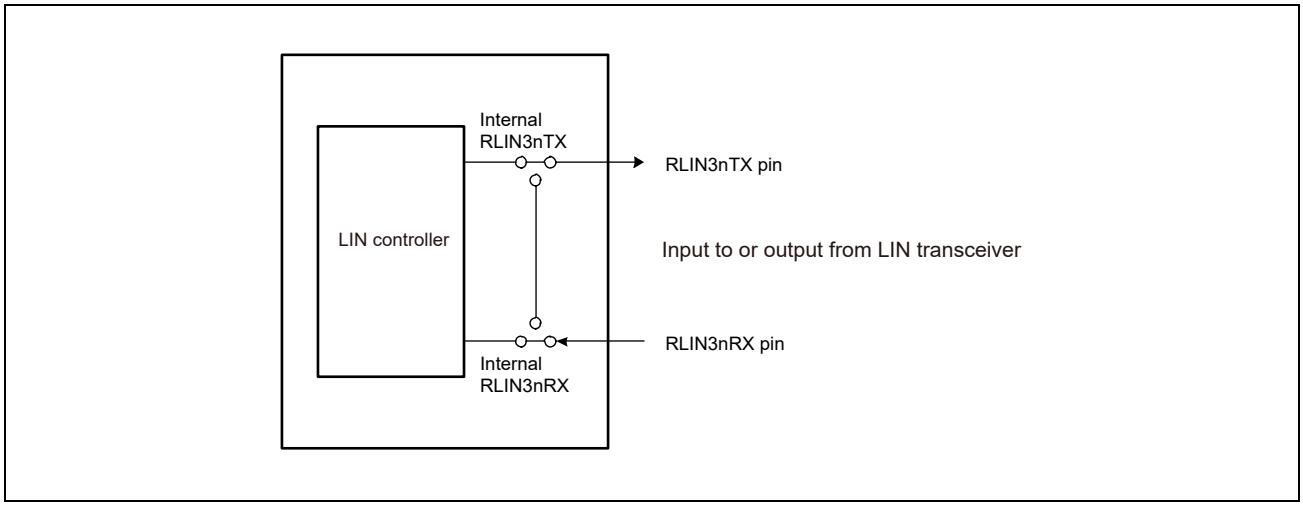


Figure 13.33 Connection in LIN Reset Mode, LIN Mode, and UART Mode

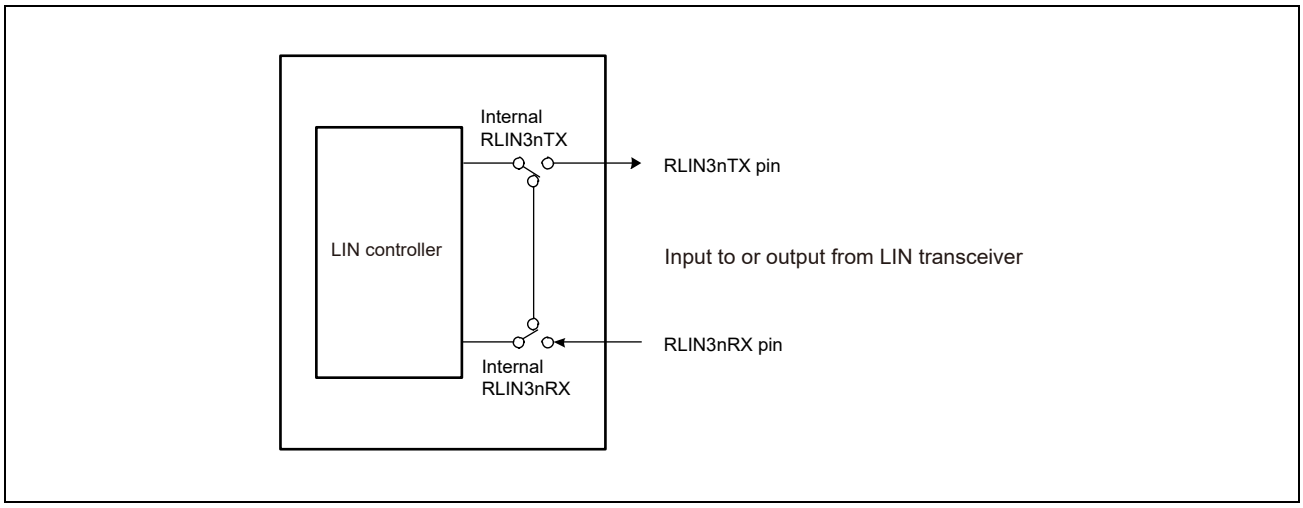


Figure 13.34 Connection in LIN Self-Test Mode

13.9.1 Change to LIN Self-Test Mode

Writing to the RLN3nLSTC register makes a transition to the LIN self-test mode.

When the LSTM bit in the RLN3nLSTC register is set to 1, the shift to the LIN self-test mode is checked. When changing to LIN self-test mode, be sure to execute a specific sequence. In that sequence, information must be written three times consecutively to the LIN self-test control register, as follows:

- Change to LIN reset mode
  Set the OM0 bit in the RLN3nLCUC register to 0 (LIN reset mode).
  Read the OMM0 bit in the RLN3nLMST register; verify that it is 0 (LIN reset mode).
- Select a LIN mode
  LMD bits in RLN3nLMD = 00_B (LIN master mode) or 11_B (LIN slave mode [fixed baud rate])
- 1st write: RLN3nLSTC register = $1010\ 0111_B$ ($A7_H$)
- 2nd write: RLN3nLSTC register = $0101\ 1000_B$ (58_H)
- 3rd write: RLN3nLSTC register = $0000\ 0001_B$ (01_H)
- Verify the transition to LIN self-test mode
  Read the LSTM bit in the RLN3nLSTC register; verify that it is 1 (LIN self-test mode).

If the key of the first write ($A7_H$) is written twice by mistake, the transition to LIN self-test mode is canceled. The above sequence should be retried from the step of first write. In addition, if a write to another LIN-related register is performed during transition to LIN self-test mode (three consecutive write operations to the RLN3nLSTC register), the transition is also canceled.

13.9.2 Transmission in LIN Master Self-Test Mode

To execute a self-test on LIN master transmission, perform the procedure below:

- Set the baud rate, noise filter, and interrupt output related registers.
  RLN3nLWBR register = 0000 $xxxx_B$*1
  RLN3nLBRP0 register = xxxx $xxxx_B$*1
  RLN3nLBRP1 register = xxxx $xxxx_B$*1
  RLN3nLMD register = 00xx $xx00_B$*1
- Set the interrupt enable and error enable related registers.
  RLN3nLIE register = 0000 $xxxx_B$*2
  RLN3nLEDE register = x000 $x0xx_B$
- Set the break field and space related registers.
  RLN3nLBFC register = 00xx $xxxx_B$
  RLN3nLSC register = 00xx $0xxx_B$
- Release from the LIN reset mode.
  Write 11_B to the OM1 and OM0 bits in the RLN3nLCUC register, and check that the OMM1 and OMM0 bits in the RLN3nLMST register are 11_B.
- Set the transmit frame related registers.
  RLN3nLDFC register = 00x1 $xxxx_B$
  RLN3nLIDB register = xxxx $xxxx_B$
  RLN3nLDBR1 to RLN3nLDBR8 register = xxxx $xxxx_B$
- Header transmission → response transmission started
  Set the FTS bit in the RLN3nLTRC register to 1 (frame transmission or wake-up transmission/reception started). The LIN master self-test mode (transmission) is executed. In this mode, interrupt are generated, and status and error status are also updated. The checksum is automatically calculated by the LIN/UART interface. To suspend the LIN master self-test mode (transmission) being executed, write 0 (LIN reset mode) to the OM0 bit in the RLN3nLCUC register for transition to LIN reset mode.
- When the transmission is completed, the reversed value of the looped-back frame data is stored in the RLN3nLIDB, RLN3nLDBRb (b = 1 to 8), and RLN3nLCBR registers (the data is reversed before being stored because the transmitted value should be compared with the looped-back value). Then, the FTS bit in the RLN3nLTRC register is cleared.
- If the transmission fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN3nLTRC register is cleared.

Note: x: Don't care

Note 1. The following register settings are not reflected to the operation of the LIN self-test mode. The LPRS bit in the RLN3nLWBR register, the RLN3nLBRP0 register, the RLN3nLBRP1 register and the LCKS bit in the RLN3nLMD register. Therefore, those settings are not necessary.

Note 2. If necessary, set the related registers described in **Section 6, Interrupt**.

Note 3. interrupt are used in the same interrupt processing, if the software processing of the successful header transmission interrupt is not completed before the generation of the successful frame transmission interrupt, the SHIE bit in the RLN3nLIE register should not be set to 1 (successful header transmission interrupt enabled). The time required from the set of the successful header transmission flag to the set of the successful frame/wake-up transmission flag is calculated by the following formula.

10 × (number of data bytes + 1) [Tbit]

1 Tbit = 1/frequency of the LIN communication clock source × 16

13.9.3 Reception in LIN Master Self-Test Mode

To execute a self-test on LIN master reception, perform the procedure below:

- Set the baud rate, noise filter, and interrupt output related registers.
  RLN3nLWBR register = 0000 $xxxx_B$*1
  RLN3nLBRP0 register = xxxx $xxxx_B$*1
  RLN3nLBRP1 register = xxxx $xxxx_B$*1
  RLN3nLMD register = 00xx $xx00_B$*1
- Set the interrupt enable and error enable related registers.
  RLN3nLIE register = 0000 $xxxx_B$*2
  RLN3nLEDE register = x000 $x0xx_B$
- Set the break field and space related registers.
  RLN3nLBFC register = 00xx $xxxx_B$
  RLN3nLSC register = 00xx $0xxx_B$*1
- Release from the LIN reset mode.
  Write 11_B to the OM1 and OM0 bits in the RLN3nLCUC register, and check that the OMM1 and OMM0 bits in the RLN3nLMST register are 11_B.
- Set the reception frame related registers.
  RLN3nLDFC register = 00x0 $xxxx_B$
  RLN3nLIDB register = xxxx $xxxx_B$
  RLN3nLDBR1 to RLN3nLDBR8 register = xxxx $xxxx_B$
  RLN3nLCBR register = xxxx $xxxx_B$
  Since the checksum value to be transmitted is not automatically calculated, set the calculation value to the RLN3nLCBR register. Here, by setting a wrong value, the functionality of checksum error can be tested.
- Header transmission → response reception started
  Set the FTS bit in the RLN3nLTRC register to 1 (frame transmission or wake-up transmission/reception started). The LIN master self-test mode (reception) is executed. In this mode, interrupt are generated, and status and error status are also updated. To suspend the LIN master self-test mode (transmission) being executed, write 0 (LIN reset mode) to the OM0 bit in the RLN3nLCUC register for transition to LIN reset mode.
- When the reception is completed, the reversed value of the looped-back frame data is stored in the RLN3nLIDB, RLN3nLDBRb (b = 1 to 8), and RLN3nLCBR registers (the data is reversed before being stored because the set value should be compared with the looped-back value). Then, the FTS bit in the RLN3nLTRC register is cleared.
- If the reception fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN3nLTRC register is cleared.

Note: x: Don't care

Note 1. The following register settings are not reflected to the operation of the LIN self-test mode.
The LPRS bit in the RLN3nLWBR register, the RLN3nLBRP0 register, the RLN3nLBRP1 register, the LCKS bit in the RLN3nLMD register, and the IBS bit in the RLN3nLSC register. Therefore, those settings are not necessary.

Note 2. If necessary, set the related registers described in **Section 6, Interrupt**.

Note 3. When the successful header transmission interrupt and the successful frame reception interrupt are used in the same interrupt processing, if the software processing of the successful header transmission interrupt is not completed before the generation of the successful frame reception interrupt, the SHIE bit in the RLN3nLIE register should not be set to 1 (successful header transmission interrupt enabled).
The time required from the set of the successful header transmission flag to the set of the successful frame/wake-up reception flag is calculated by the following formula.

10 × (number of data bytes + 1) [Tbit]

1 Tbit = 1/frequency of the LIN communication clock source × 16

13.9.4 Transmission in LIN Slave Self-Test Mode

To execute a self-test on LIN slave transmission, perform the procedure below:

- Set the baud rate, noise filter, and interrupt output related registers.
  RLN3nLWBR register = 0000 xxx0_B*1
  RLN3nLBRP0 register = xxxx xxxx$_B$*1
  RLN3nLBRP1 register = xxxx xxxx$_B$*1
  RLN3nLMD register = 00xx 0011_B
- Set the interrupt enable and error enable related registers.
  RLN3nLIE register = 0000 xxxx$_B$*2
  RLN3nLEDE register = xx0x x00x$_B$
- Set the break field and space related registers.
  RLN3nLBFC register = 0000 000x$_B$*3
  RLN3nLSC register = 00xx 0001_B
- Release from the LIN reset mode.
  Write 11_B to the OM1 and OM0 bits in the RLN3nLCUC register, and check that the OMM1 and OMM0 bits in the RLN3nLMST register are 11_B.
- Set the transmit frame related registers.
  RLN3nLDFC register= 00x1 xxxx$_B$
  RLN3nLIDB register = xxxx xxxx$_B$
  RLN3nLDBR1～RLN3nLDBR8 register = xxxx xxxx$_B$
- Header reception → response transmission started
  Set the FTS bit in the RLN3nLTRC register to 1 (header reception or wake-up transmission/reception started). (Without any setting of the RTS bit in the RLN3nLTRC register, the header reception and the response transmission are executed in this order.)
  The LIN slave self-test mode (transmission) is executed. In this mode, interrupt are generated, and status and error status are also updated.
  The checksum is automatically calculated by the LIN/UART interface. To suspend the LIN master self-test mode (transmission) being executed, write 0 (LIN reset mode) to the OM0 bit in the RLN3nLCUC register for transition to LIN reset mode.
- When the transmission is completed, the reversed value of the looped-back frame data is stored in the RLN3nLIDB, RLN3nLDBRb (b = 1 to 8), and RLN3nLCBR registers (the data is reversed before being stored because the transmitted value should be compared with the looped-back value). Then, the FTS bit in the RLN3nLTRC register is cleared.
- If the transmission fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN3nLTRC register is cleared.

Note: x: Don't care

Note 1. The following register settings are not reflected to the operation of the LIN self-test mode. The LPRS bit in the RLN3nLWBR register, the RLN3nLBRP0 register, and the RLN3nLBRP1 register. Therefore, those settings are not necessary.

Note 2. If necessary, set the related registers described in **Section 6, Interrupt**.

Note 3. According to the setting of this register, 9.5-Tbit or 10.5-Tbit width break is output from the internal RLIN3nTX.

Note 4. When the successful header reception interrupt and the successful response transmission interrupt are used in the same interrupt processing, if the software processing of the successful header reception interrupt is not completed before the generation of the successful response transmission interrupt, the SHIE bit in the RLN3nLIE register should not be set to 1 (successful header reception interrupt enabled). The time from setting of the successful header reception flag to setting of the successful response/wake-up transmission flag is calculated by using the following formula.

$$10 \times (\text{number of data bytes} + 1)\ [\text{Tbit}]$$

$$1\ \text{Tbit} = 1/\text{frequency of the LIN communication clock source} \times 16$$

13.9.5 Reception in LIN Slave Self-Test Mode

To execute a self-test on LIN slave reception, perform the procedure below:。

- Set the baud rate, noise filter, and interrupt output related registers.
  RLN3nLWBR register = 0000 $xxx0_B$*1
  RLN3nLBRP0 register = xxxx $xxxx_B$*1
  RLN3nLBRP1 register = xxxx $xxxx_B$*1
  RLN3nLMD register = 00xx 0011_B
- Set the interrupt enable and error enable related registers.
  RLN3nLIE register = 0000 $xxxx_B$*2
  RLN3nLEDE register = xx0x $x00x_B$
- Set the break field and space related registers.
  RLN3nLBFC register = 0000 $000x_B$*3
  RLN3nLSC register = 00xx 0001_B*1
- Release from the LIN reset mode.
  Write 11_B to the OM1 and OM0 bits in the RLN3nLCUC register, and check that the OMM1 and OMM0 bits in the RLN3nLMST register are 11_B.
- Set the reception frame related registers.
  RLN3nLDFC register = 00x0 $xxxx_B$
  RLN3nLIDB register = xxxx $xxxx_B$
  RLN3nLDBR1 to RLN3nLDBR8 register = xxxx $xxxx_B$
  RLN3nLCBR register = xxxx $xxxx_B$
  Since the checksum value to be transmitted is not automatically calculated, set the calculation value to the RLN3nLCBR register. Here, by setting the wrong value, the functionality of checksum error can be tested.
- Header reception → response reception started
  Set the FTS bit in the RLN3nLTRC register to 1 (header reception or wake-up transmission/reception started). (Without any setting of the RTS bit in the RLN3nLTRC register, the header reception and the response reception are executed in this order.)
  The LIN slave self-test mode (reception) is executed. In this mode, interrupt are generated, and status and error status are also updated. To suspend the LIN master self-test mode (transmission) being executed, write 0 (LIN reset mode) to the OM0 bit in the RLN3nLCUC register for transition to LIN reset mode.

- When the reception is completed, the reversed value of the looped-back frame data is stored in the RLN3nLIDB, RLN3nLDBRb (b = 1 to 8), and RLN3nLCBR registers (the data is reversed before being stored because the set value should be compared with the looped-back value). Then, the FTS bit in the RLN3nLTRC register is cleared.
- If the reception fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN3nLTRC register is cleared.

Note: x: Don't care

Note 1. The following register settings are not reflected to the operation of the LIN self-test mode.
The LPRS bit in the RLN3nLWBR register, the RLN3nLBRP0 register, the RLN3nLBRP1 register, and the IBS bit in the RLN3nLSC register.
Therefore, those settings are not necessary.

Note 2. If necessary, set the related registers described in **Section 6, Interrupt**.

Note 3. According to the setting of this register, 9.5-Tbit or 10.5-Tbit width break is output from the internal RLIN3nTX.

Note 4. When the successful header reception interrupt and the successful response reception interrupt are used in the same interrupt processing, if the software processing of the successful header reception interrupt is not completed before the generation of the successful response reception interrupt, the SHIE bit in the RLN3nLIE register should not be set to 1 (successful header reception interrupt enabled).
The time required from the set of the successful header reception flag to the set of the successful response/wake-up reception flag is calculated by the following formula.

10 × (number of data bytes + 1) [Tbit]

1 Tbit = 1/frequency of the LIN communication clock source × 16

13.9.6 Terminating LIN Self-Test Mode

To terminate LIN self-test mode, perform the procedure below:

- Write 0 (LIN reset mode) to the OM0 bit in the RLN3nLCUC register.
If the OMM1 and OMM0 bits in the RLN3nLMST register are not 11_B, write 11_B to the OM1 and OM0 bits in the RLN3nLCUC register. After confirming that the OMM1 and OMM0 bits in the RLN3nLMST register have turned 11_B, change to LIN reset mode.
- Verify the cancelation of LIN self-test mode.
Read the LSTM bit in the RLN3nLSTC register; confirm that it is not 0 (not in LIN self-test mode)
- Verify the transition to LIN reset mode.
Read the OMM0 bit in the RLN3nLMST register; verify that it is 0 (LIN reset mode).

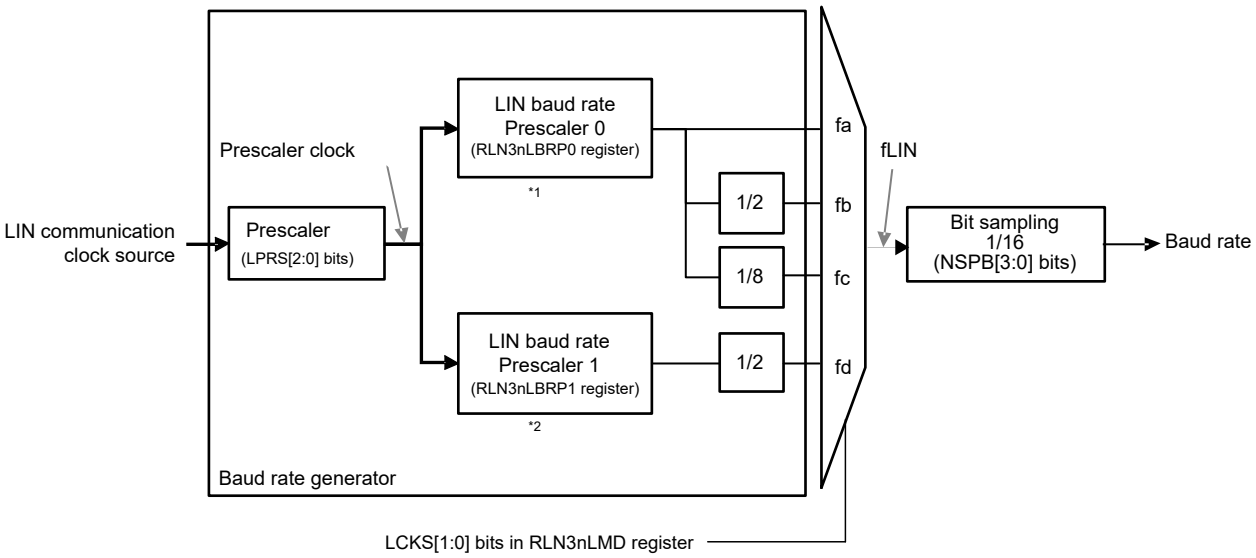
13.10 Baud Rate Generator

The prescaler clock is obtained by frequency-dividing the LIN communication clock source by the prescaler, and the LIN system clock (fLIN) is obtained by frequency-dividing the prescaler clock by the baud rate generator. The clock obtained by frequency-dividing the LIN system clock (fLIN) by the number of samples is the baud rate. The reciprocal of this baud rate is called the bit time (Tbit).

The LIN/UART interface has two kinds of baud rate generators. The baud rate generators switch over according to the mode used.

13.10.1 LIN Master Mode

Figure 13.35 shows a block diagram of baud rate generation in LIN master mode.



Note 1. When the value in RLN3nLBRP0 register is N (N = 0 to 255), the clock frequency is divided by N + 1.

Note 2. When the value in RLN3nLBRP1 register is M (M = 0 to 255), the clock frequency is divided by M + 1.

Figure 13.35 Block Diagram of Baud Rate Generation in LIN Master Mode

By setting the RLN3nLBRP0 register so that fa is 307200 Hz (= 19200 × 16), the resulting bit rates are fa = 19200 × 16, fb = 9600 × 16, and fc = 2400 × 16. These bit rates are frequency-divided by 16 in the bit timing generator, enabling bit rates of 19200 bps, 9600 bps and 2400 bps, to be generated. Also, by setting the RLN3nLBRP1 register so that fd is 166672 Hz (= 10417 × 16), the resulting bit rate is fd = 10417 × 16. This bit rate is frequency-divided by 16 in the bit timing generator, enabling 10417 bps to be generated.

The formula for baud rate is described below.

Baud rate of LIN slave

= {Frequency of LIN communication clock source} × (RLN3nLWBR.LPRS[2:0] selection clock) / (RLN3nLBRP0 + 1) /16 [bps] (When fa is selected for f_{LIN})

= {Frequency of LIN communication clock source} × (RLN3nLWBR.LPRS[2:0] selection clock) / (RLN3nLBRP0 + 1) /2/16 [bps] (When fb is selected for f_{LIN})

= {Frequency of LIN communication clock source} × (RLN3nLWBR.LPRS[2:0] selection clock)
/ (RLN3nLBRP0 + 1) /8/16 [bps] (When fc is selected for f_{LIN})

= {Frequency of LIN communication clock source} × (RLN3nLWBR.LPRS[2:0] selection clock)
/ (RLN3nLBRP1 + 1) /2/16 [bps] (When fd is selected for f_{LIN})

13.10.2 LIN Slave Mode

Figure 13.36 shows a block diagram of baud rate generation in LIN slave mode.

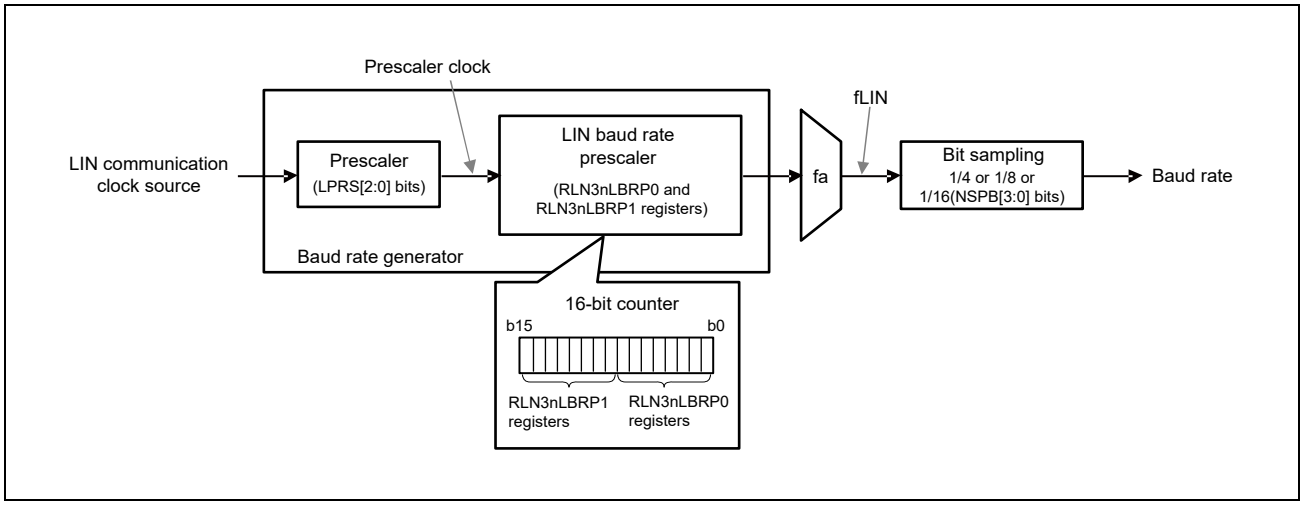


Figure 13.36 Block Diagram of Baud Rate Generation in LIN Slave Mode

In LIN slave mode (auto baud rate), the baud rate can be set to the range from 1 kbps to 20 kbps. Set the prescaler clock as follows according to the target baud rate:

[Target baud rate]	[Prescaler clock]
1 kbps to 20 kbps	: 4 MHz*1
1 kbps to 2.4 kbps (excluding 2.4 kbps)	: 4 MHz
2.4 kbps to 20 kbps	: 8 MHz to 12 MHz

Note 1. Use the clock with NSPB[3:0] bits in the RLN3nLWBR register set to "0011_B" (four samplings).

The formula for baud rate is described below.

Baud rate of LIN slave

= {Frequency of LIN communication clock source} × (RLN3nLWBR.LPRS[2:0] selection clock)
/ (RLN3nLBRP01 + 1) /16 [bps] ([Fixed baud rate])

= {Frequency of LIN communication clock source} × (RLN3nLWBR.LPRS[2:0] selection clock)
/ (RLN3nLBRP01 + 1) /4 or 8 [bps] ([Auto baud rate])

NOTE

For a LIN slave with fixed baud rate, set the NSPB[3:0] bit to “0000_B” (16 samples) or “1111_B” (16 samples). For a LIN slave with auto baud rate, set the NSPB[3:0] bits to “0011_B” (4 samples) or “0100_B” (8 samples).

13.10.3 UART Mode

Figure 13.37 shows a block diagram of baud rate generation in UART mode.

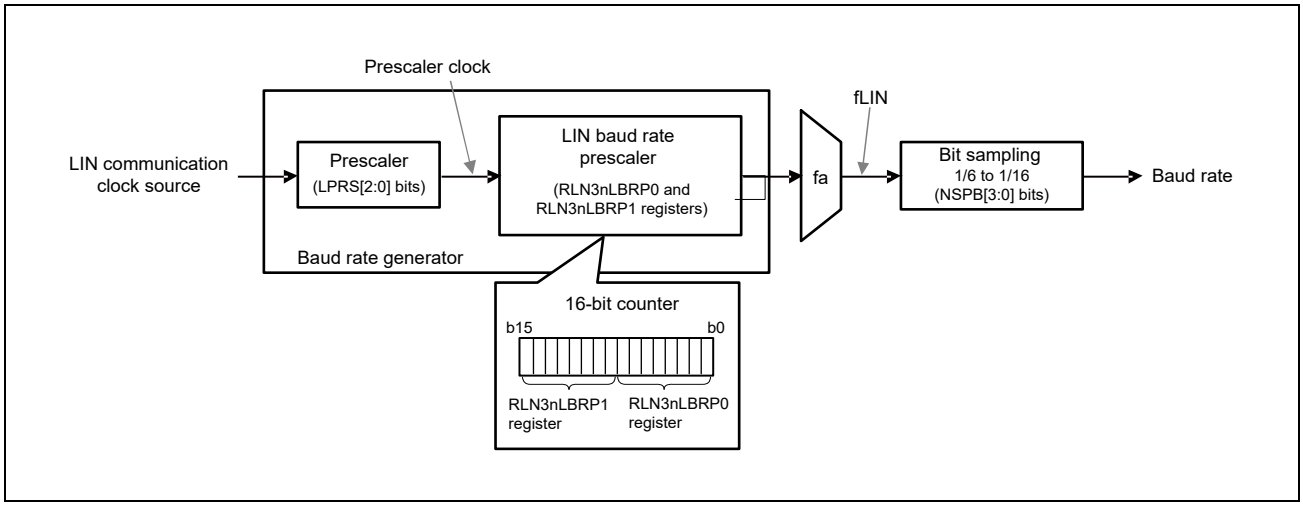


Figure 13.37 Block Diagram of Baud Rate Generation in UART Mode

UART baud rate is calculated with the following formula:

UART baud rate

= {LIN communication clock source frequency} × (RLN3nLWBR.LPRS[2:0] select clock) / (RLN3nLBRP01 + 1) / {RLN3nLWBR.NSPB[3:0] select count} [bps]

13.11 Noise Filter

The LIN/UART interface has a noise filter for reducing erroneous receiving of data due to noise. By setting the LRDNFS bit in the RLN3nLMD register to 0 (to use the noise filter), the noise filter is activated. The noise filter samples the level of the synchronized RLIN3nRX with the prescaler clock, and outputs the sampling value determined by a 3-sampling majority rule. The value of each bit of the receive data is determined based on the noise filter output.

Figure 13.38 shows the configuration of the noise filter, **Figure 13.39** shows an example of a noise filter circuit, and **Figure 13.40** shows the determination of the received data when the noise filter is used.

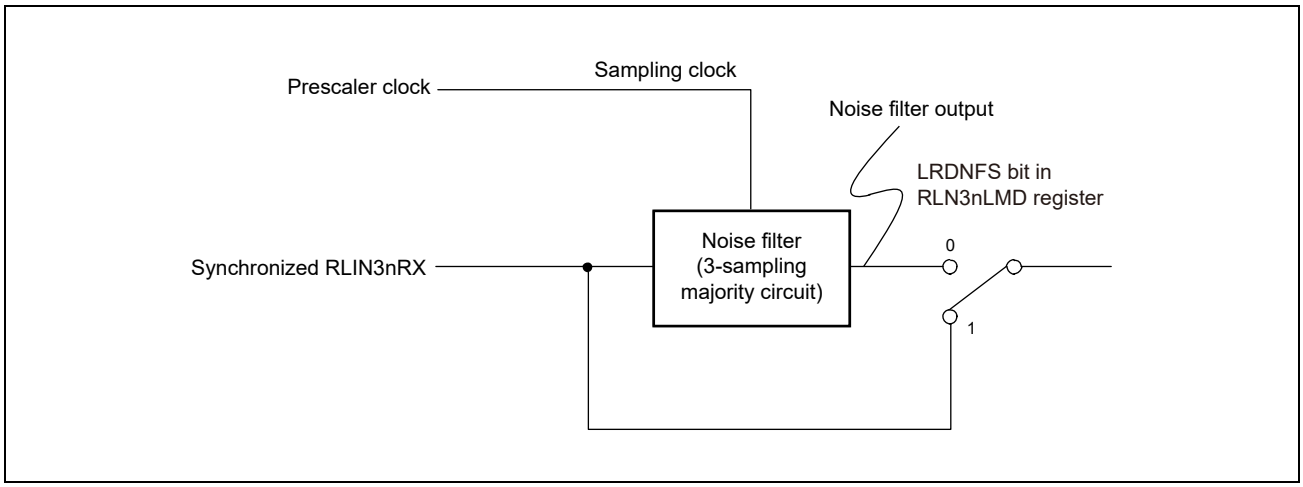


Figure 13.38 Configuration of Noise Filter

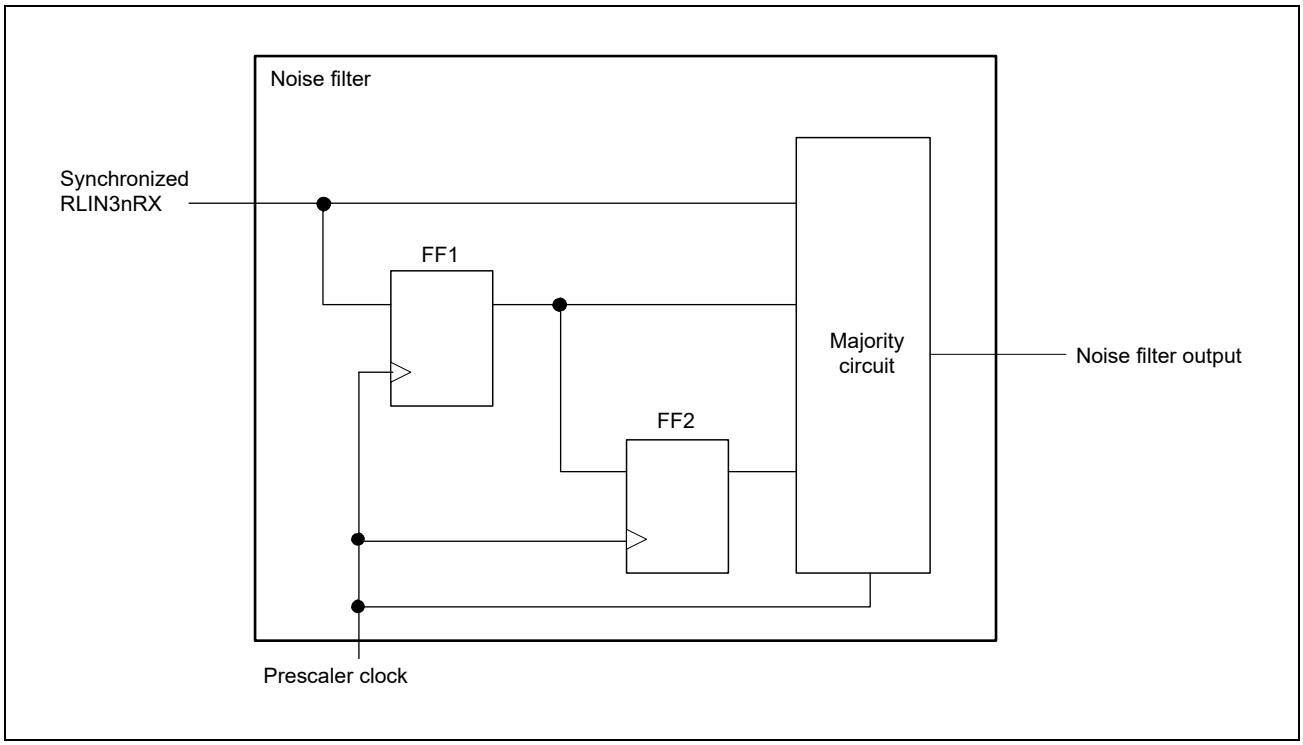


Figure 13.39 Example of Noise Filter Circuit

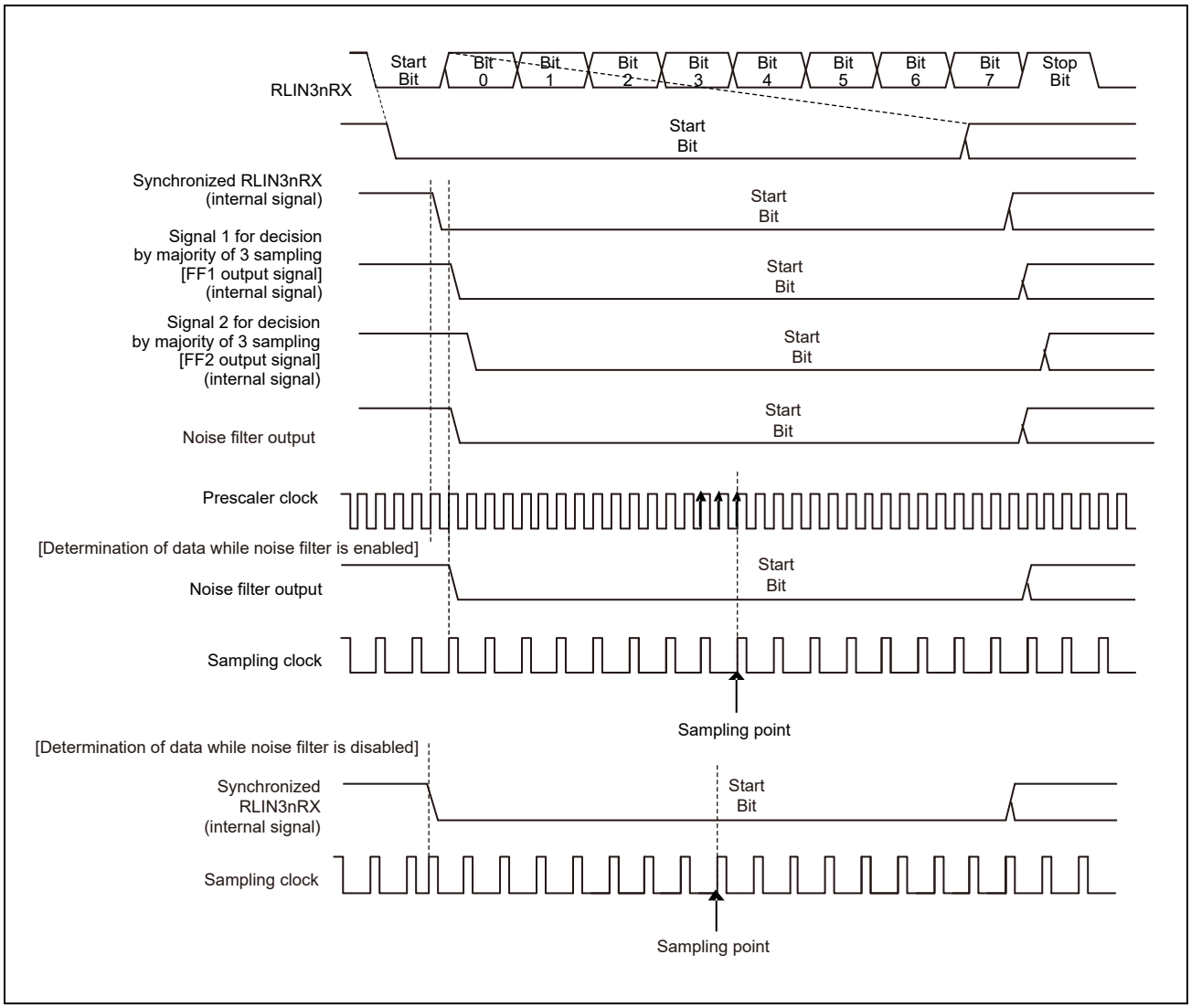


Figure 13.40 Determination of Received Data when Noise Filter is Used

13.12 Usage Notes

13.12.1 Notes on the LIN Master Mode

No response transmission is started if the initiation of a response transmission (writing a 1 into RLN3nLTRC.RTS) and the generation of an error*[1] occur at the same time in frame separate mode (RLN3nLDFC.FSM = 1).

It is likely that unexpected data is transmitted during the response transmission processing after the header for the next frame is received even though RLN3nLTRC.RTS is not set to 1. Once an error occurred, the LIN/UART module (RLIN3) is reinitialized by transiting to the LIN reset mode.

Note 1. Bit error, physical bus error, frame/response timeout error, framing error, or response preparation error.

13.12.2 Notes on the LIN Slave Mode (Fixed Baud Rate)

If the low level width which is specified by the receive brake (low level) detection width configuration bit (RLN3nLBFC.LBLT) is detected during a response transmission (RLN3nLDFC.RCDS = 1 and RLN3nLTRC.RTS = 1), the response transmission is suspended and the RLIN3nTX pin keeps the same state that was established when the low level width was detected.

In fixed baud rate LIN slave mode, RLN3nLEDE.BERE must be set to 1 (enable bit error detection) and RLN3nLSC.IBHS[2:0] to 001_B to 111_B (set the response space to a value other than 0 Tbit).

If the response space is set to 0 Tbit (RLN3nLSC.IBHS[2:0] = 000_B), the LIN/UART module (RLIN3) is reinitialized by setting RLN3nLEDE.BERE = 1 and RLN3nLEDE.TERE = 1 (enable timeout error detection) and transiting to the LIN reset mode after a timeout error occurs.

13.12.3 Notes on the LIN Slave Mode (Auto Baud Rate)

The header may not be received if the conditions (1) and (2) listed below are satisfied in succession, in which case the ID field is erroneously recognized. The next header is received normally, however.

1. A falling edge is detected within the period from the sampling point at which stop bit and response/interbyte space bit error occurred till the end of this bit.
2. The falling edge for the start bit of the ID field is detected within the period from the sampling point for the next sync filed till the end of this bit.

Section 14 CAN Interface (RS-CANFD)

This section contains a generic description of the CAN Interface (RS-CANFD).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of RS-CANFD.

14.1 Features of RH850/C1M-A RS-CANFD

14.1.1 Number of Units and Channels

This microcontroller has the following number of RS-CANFD units.

Table 14.1 Number of Units

Product Name	RH850/C1M-A2	RH850/C1M-A1
Number of Units	1	
Name	RSCFDn (n = 0)	

The individual products have the CAN Interface Channel listed below.

Table 14.2 Unit Configurations and Channels

Unit Name	Channel Name	RH850/C1M-A2	RH850/C1M-A1
RSCFD0	CAN0	√	√
	CAN1	√	√
	CAN2	√	√
	CAN3	√	√

The RS-CANFD has two interface modes (classical CAN mode and CAN FD mode) and uses different registers for each mode. There are two types of register names RSCANnXXX and RSCFDnCFDXXX (XXX: arbitrary) depending on interface modes. When explaining specifications common to two registers, register names are described as RSCFDn(CFD)XXX.

Table 14.3 Indices

Index	Meaning
n	Throughout this section, the individual RS-CANFD units are generically indicated by the index "n" (n = 0); for example, RSCFDn(CFD)GCTR is the global control register of the RSCFD unit.
m	Throughout this section, the individual channels of RS-CANFD units are generically indicated by the index "m" (m = 0 to 3); for example, RSCFDn(CFD)CmSTS is the channel m status register.
j	The individual registers associated with receive rule table are generically indicated by the index "j" (j = 0 to 15); for example, RSCFDn(CFD)GAFLIDj is the receive rule ID register.
k	The individual transmit/receive FIFO buffers are generically indicated by the index “k” (k = 0 to [channel m × 3 + 2]); for example, RSCFDn(CFD)CFCCk is the transmit/receive FIFO buffer configuration/control register.
x	The individual receive FIFO buffers are generically identified by the index “x” (x = 0 to 7); for example, RSCFDn(CFD)RFSTSx is the receive FIFO buffer status register.
d	Data field registers of transmit/receive FIFO buffers and receive FIFO buffers are identified by "d" (classical CAN mode: d = 0 to 1, CAN FD mode: d = 0 to 15). For example, the transmit/receive FIFO buffer data field register is described as RSCFDn(CFD)CFDFd_k.
q	The individual receive buffers are generically indicated by the index “q” (q = 0 to [channel m × 16 + 15]); for example, RSCFDn(CFD)RMIDq is the receive buffer ID register.
p	The individual transmit buffers are generically indicated by the index “p” (q = 0 to [channel m × 16 + 15]; for example, RSCFDn(CFD)TMCp is the transmit buffer control register.
b	Data field registers of receive buffers and transmit buffers are identified by “b” (classical CAN mode: b = 0 to 1, CAN FD mode: b = 0 to 4). For example, the receive buffer data field register is described as RSCFDn(CFD)RMDFb_q.
r	The individual RAM tests for CAN are generically indicated by the index “r” (r = 0 to 63); for example, RSCFDn(CFD)RPGACCr is the RAM test page access register.
y	The registers not covered above are indicated by the letter “y” (y = 0, 1); for example, RSCFDn(CFD)RMNDy is a receive buffer new data register.

14.1.2 Register Base Address

RSCFDn base addresses are listed in the following table.

RSCFDn register addresses are given as offsets from the base addresses in general.

Table 14.4 Register Base Address

Base Address Name	Base Address
<RSCFD0_base>	FFD0 0000_H

14.1.3 Clock Supply

The RSCFDn clock supply is shown in the following table.

Table 14.5 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
RSCFDn	clk_xincan	MOSC (Main OSC)
	clkc	CLKC_LSB (unmodulated low-speed peripheral clock)
	pclk	CLK_HSB (high-speed peripheral clock)

14.1.4 Interrupt Request

RSCFDn interrupt requests are listed in the following table.

Table 14.6 Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt Number
RSCFD0		
INTRCANGERR	CAN global error interrupt	264
INTRCANGRECC	CAN receive FIFO interrupt	263
CAN0		
INTRCAN0ERR	CAN0 error interrupt	266
INTRCAN0REC	CAN0 transmit/receive FIFO receive completion interrupt	265
INTRCAN0TRX	CAN0 transmit interrupt	267
CAN1		
INTRCAN1ERR	CAN1 error interrupt	269
INTRCAN1REC	CAN1 transmit/receive FIFO receive completion interrupt	268
INTRCAN1TRX	CAN1 transmit interrupt	270
CAN2		
INTRCAN2ERR	CAN2 error interrupt	272
INTRCAN2REC	CAN2 transmit/receive FIFO receive completion interrupt	271
INTRCAN2TRX	CAN2 transmit interrupt	273
CAN3		
INTRCAN3ERR	CAN3 error interrupt	275
INTRCAN3REC	CAN3 transmit/receive FIFO receive completion interrupt	274
INTRCAN3TRX	CAN3 transmit interrupt	276

• DMA Request

RSCFDn DMA requests are listed in the following table.

Table 14.7 DMA Request

DMA/DTS Source Signal	Outline	DMA Trigger Number*1		DTS Trigger Number*1	
		1st	2nd	1st	2nd
RSCFD0					
CAN RX FIFO 0	CAN RX FIFO DMA request #0	103	—	103	—
CAN RX FIFO 1	CAN RX FIFO DMA request #1	104	—	104	—
CAN RX FIFO 2	CAN RX FIFO DMA request #2	105	—	105	—
CAN RX FIFO 3	CAN RX FIFO DMA request #3	106	—	106	—
CAN RX FIFO 4	CAN RX FIFO DMA request #4	107	—	107	—
CAN RX FIFO 5	CAN RX FIFO DMA request #5	108	—	108	—
CAN RX FIFO 6	CAN RX FIFO DMA request #6	109	—	109	—
CAN RX FIFO 7	CAN RX FIFO DMA request #7	110	—	110	—
CAN0					
CAN ch.0 FIFO	CAN0 COM FIFO DMA request	99	—	99	—
CAN1					
CAN ch.1 FIFO	CAN1 COM FIFO DMA request	100	—	100	—
CAN2					
CAN ch.2 FIFO	CAN2 COM FIFO DMA request	101	—	101	—
CAN3					
CAN ch.3 FIFO	CAN3 COM FIFO DMA request	102	—	102	—

—: Not assigned

Note 1. 1st: Primary channel, 2nd: Secondary Channel

14.1.5 Reset Source

The reset source of RSCFDn is shown below. The reset source initializes RSCFDn.

Table 14.8 Reset Source

Unit Name	Reset Source
RSCFDn	All reset sources

14.1.6 External Input/Output Signals

External input/output signals of RSCFDn are listed below.

Table 14.9 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
RSCFD0		
CANmRX (m = 0 to 3)	CANm receive data input	CANmRX (m = 0 to 3)
CANmTX (m = 0 to 3)	CANm transmit data output	CANmTX (m = 0 to 3)

14.2 Overview

14.2.1 Functional Overview

Table 14.10 shows the RS-CANFD module specifications. **Figure 14.1** shows the RS-CANFD module block diagram.

Table 14.10 RS-CANFD Module Specifications (1/2)

Item	Specification
Number of channels	4
Protocol	ISO11898-1:2015 compliant Using CAN FD frames is selectable by switching interface modes.
Communication speed	Classical CAN mode: • Maximum 1 Mbps Communication speed (CANm bit time clock) = $\frac{1}{\text{CANm bit time}}$ CANm bit time = CANmTq × Tq count per bit CANmTq = $\frac{\text{(BRP[9:0] bits in the RSCANnCmCFG register + 1)}}{\text{fCAN}}$ fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCANnGCFG register) CAN FD mode: • Nominal bit rate: max.1 Mbps, data bit rate: max. 5 Mbps Transmission rate (CANm nominal bit time clock) = $\frac{1}{\text{CANm nominal bit time}}$ Transmission rate (CANm data bit time clock) = $\frac{1}{\text{CANm data bit time}}$ CANm nominal bit time = CANmTq(N) × Tq count per nominal bit CANm data bit time = CANmTq(D) × Tq count per data bit CANmTq(N) = $\frac{\text{(NBRP[9:0] bits in the RSCFDnCFDCmNCFG register + 1)}}{\text{fCAN}}$ CANmTq(D) = $\frac{\text{(DBRP[7:0] bits in the RSCFDnCFDCmDCFG register + 1)}}{\text{fCAN}}$ fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCFDnCFDGCFG register) m = 0 to 3 Tq: Time quantum
Buffer	320 buffers in total • Individual buffers: 64 buffers (16 buffers × 4 channels) Transmit buffer: 16 buffers per channel Transmit queue: Single queue per channel (shared with the transmit buffer; up to 16 buffers allocatable) • Shared buffers: 256 buffers for all channels Receive buffer: 0 to 64 buffers Receive FIFO buffer: 8 FIFO buffers (up to 128 buffers allocatable to each) Transmit/receive FIFO buffer: 3 FIFO buffers per channel (up to 128 buffers allocatable to each) • ECC included
Reception function	• Receives data frames and remote frames. • Selects ID format (standard ID, extended ID, or both IDs) to be received. • Sets interrupt enable/disable for each FIFO. • Mirror function (reception of messages transmitted from the own CAN node) • Timestamp function (to record message reception time as a 16-bit timer value)
Reception filter function	• Selects receive messages according to 256 receive rules. • Sets the number of receive rules (0 to 128) for each channel. • Acceptance filter processing: Sets ID and mask for each receive rule. • DLC filter processing: Enables DLC filter check for each acceptance rule.

Table 14.10 RS-CANFD Module Specifications (2/2)

Item	Specification
Receive message transfer function	• Routing function Transfers receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer • Label addition function Stores label information together with a message in a receive buffer and FIFO buffer.
Transmission function	• Transmits data frames and remote frames. • Selects ID format (standard ID, extended ID, or both IDs) to be transmitted. • Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer. • Selects ID priority transmission or transmit buffer number priority transmission. • Transmit request can be aborted (possible to confirm with a flag) • One-shot transmission function
Interval transmission function	Transmit messages at configurable intervals (transmit mode or gateway mode of transmit/receive FIFO buffers)
Transmit queue function	Transmits all stored messages according to the ID priority.
Transmit history function	Stores the history information of transmission-completed messages Adds timestamp (recording message transmission time as a 16-bit timer value) to the history information.
Gateway function	Transmits a received message automatically.
Bus off recovery mode selection	Selects the method for returning from bus off state • ISO11898-1:2015 compliant • Automatic entry to channel halt mode at bus-off entry • Automatic entry to channel halt mode at bus-off end • Transition to channel halt mode by program request • Transition to the error-active state by program request (forcible return from the bus off state)
Error status monitoring	• Monitors CAN protocol errors (stuff error, form error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock). • Detects error status transitions (error warning, error passive, bus off entry, and bus off recovery) • Reads the error counter. • Monitors DLC errors.
Interrupt source	14 sources • Global Interrupts (2 sources) Receive FIFO interrupt Global error interrupt • Channel interrupts (3 sources/channel) CANm transmit interrupt (m = 0 to 3) - CANm transmit complete interrupt - CANm transmit abort interrupt - CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode) - CANm transmit history interrupt - CANm transmit queue interrupt CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode) CANm error interrupt
CAN stop mode	Reduces power consumption by stopping clock supply to the RS-CANFD module.
CAN clock source	Selects the clkc or the clk_xincan.
Test function	Test function for user evaluation • Listen-only mode • Self-test mode 0 (external loopback) • Self-test mode 1 (internal loopback) • Restricted operation mode • RAM test (read/write test) • Inter-channel communication test [CRC error test enabled]

14.2.2 Interface Modes

The RS-CANFD has two interface modes.

- Classical CAN mode: Handles only classical CAN frames.
- CAN FD mode: Handles classical CAN frames and CAN FD frames.

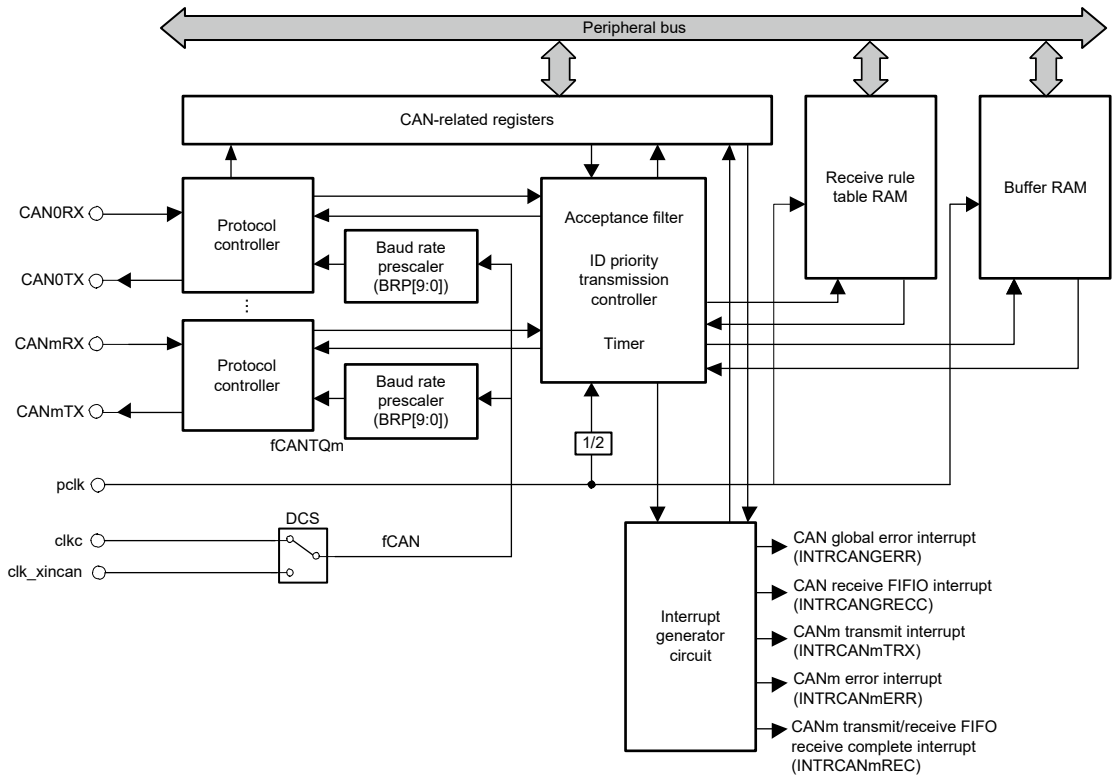
These two modes use different register maps with the same base address. Register maps change by switching interface modes.

Interface modes are switched by the RCMC bit in the RSCFDn(CFD)GRMCFG register.

14.2.3 CAN FD Protocol Switching

This product supports ISO 11898-1:2015 compatible CAN FD with a new CRC field containing a stuff counter. It can also support the CRC field that conforms to ISO/CD 11898-1 (2014-08-12 version) when the NIE bit of the RSCFDnCFDGCRCCFG register is set to 1.

14.2.4 Block Diagram



Note: m = 0 to 3
BRP[9:0]: Bits in the RSCANnCmCFG register
DCS: Bits in the RSCANnGCFG register
fCANTQm: CANmTq clock
fCAN: CAN clock

Figure 14.1 RS-CANFD Module Block Diagram (Classical CAN Mode)

In CAN FD mode, different clock signals are input to the baud rate prescaler and the protocol controller respectively. See **Section 14.11.1.3, Communication Speed Setting**.

14.3 Registers (Classical CAN Mode)

14.3.1 List of Registers

The following tables list RS-CANFD registers to be used in classical CAN mode.

For details about <RSCFDn_base>, see **Section 14.1.2, Register Base Address.**

Table 14.11 Registers (1/3)

Module	Register	Symbol	Address
Interface mode-related registers			
RSCANn	Global Interface Mode Select Register	RSCANnGRMCFG	<RSCFDn_base> + $04FC_H$
Channel-related registers			
RSCANn	Channel m Configuration Register	RSCANnCmCFG	<RSCFDn_base> + 0000_H + (10_H × m)
RSCANn	Channel m Control Register	RSCANnCmCTR	<RSCFDn_base> + 0004_H + (10_H × m)
RSCANn	Channel m Status Register	RSCANnCmSTS	<RSCFDn_base> + 0008_H + (10_H × m)
RSCANn	Channel m Error Flag Register	RSCANnCmERFL	<RSCFDn_base> + $000C_H$ + (10_H × m)
Global-related registers			
RSCANn	Global Configuration Register	RSCANnGCFG	<RSCFDn_base> + 0084_H
RSCANn	Global Control Register	RSCANnGCTR	<RSCFDn_base> + 0088_H
RSCANn	Global Status Register	RSCANnGSTS	<RSCFDn_base> + $008C_H$
RSCANn	Global Error Flag Register	RSCANnGERFL	<RSCFDn_base> + 0090_H
RSCANn	Global Timestamp Counter Register	RSCANnGTSC	<RSCFDn_base> + 0094_H
RSCANn	Global TX Interrupt Status Register 0	RSCANnGTINTSTS0	<RSCFDn_base> + 0460_H
RSCANn	Global FD Configuration Register	RSCANnGFDCFG	<RSCFDn_base> + 0474_H
Receive rule-related registers			
RSCANn	Receive Rule Entry Control Register	RSCANnGAFLECTR	<RSCFDn_base> + 0098_H
RSCANn	Receive Rule Configuration Register 0	RSCANnGAFLCFG0	<RSCFDn_base> + $009C_H$
RSCANn	Receive Rule ID Register j	RSCANnGAFLIDj	<RSCFDn_base> + 0500_H + (10_H × j)
RSCANn	Receive Rule Mask Register j	RSCANnGAFLMj	<RSCFDn_base> + 0504_H + (10_H × j)
RSCANn	Receive Rule Pointer 0 Register j	RSCANnGAFLP0_j	<RSCFDn_base> + 0508_H + (10_H × j)
RSCANn	Receive Rule Pointer 1 Register j	RSCANnGAFLP1_j	<RSCFDn_base> + $050C_H$ + (10_H × j)
Receive buffer-related registers			
RSCANn	Receive Buffer Number Register	RSCANnRMNB	<RSCFDn_base> + $00A4_H$
RSCANn	Receive Buffer New Data Register y	RSCANnRMNDy	<RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCANn	Receive Buffer ID Register q	RSCANnRMIDq	<RSCFDn_base> + 0600_H + (10_H × q)
RSCANn	Receive Buffer Pointer Register q	RSCANnRMPTRq	<RSCFDn_base> + 0604_H + (10_H × q)
RSCANn	Receive Buffer Data Field 0 Register q	RSCANnRMDF0_q	<RSCFDn_base> + 0608_H + (10_H × q)
RSCANn	Receive Buffer Data Field 1 Register q	RSCANnRMDF1_q	<RSCFDn_base> + $060C_H$ + (10_H × q)

Table 14.11 Registers (2/3)

Module	Register	Symbol	Address
Receive FIFO buffer-related registers			
RSCANn	Receive FIFO Buffer Configuration and Control Register x	RSCANnRFCCx	<RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCANn	Receive FIFO Buffer Status Register x	RSCANnRFSTSx	<RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCANn	Receive FIFO Buffer Pointer Control Register x	RSCANnRFPCTRx	<RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCANn	Receive FIFO Buffer Access ID Register x	RSCANnRFIDx	<RSCFDn_base> + $0E00_H$ + (10_H × x)
RSCANn	Receive FIFO Buffer Access Pointer Register x	RSCANnRFPTRx	<RSCFDn_base> + $0E04_H$ + (10_H × x)
RSCANn	Receive FIFO Buffer Access Data Field 0 Register x	RSCANnRFDF0_x	<RSCFDn_base> + $0E08_H$ + (10_H × x)
RSCANn	Receive FIFO Buffer Access Data Field 1 Register x	RSCANnRFDF1_x	<RSCFDn_base> + $0E0C_H$ + (10_H × x)
Transmit/receive FIFO buffer related registers			
RSCANn	Transmit/receive FIFO Buffer Configuration and Control Register k	RSCANnCFCCk	<RSCFDn_base> + 0118_H + (04_H × k)
RSCANn	Transmit/receive FIFO Buffer Status Register k	RSCANnCFSTSk	<RSCFDn_base> + 0178_H + (04_H × k)
RSCANn	Transmit/receive FIFO Buffer Pointer Control Register k	RSCANnCFPCTRk	<RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCANn	Transmit/receive FIFO Buffer Access ID Register k	RSCANnCFIDk	<RSCFDn_base> + $0E80_H$ + (10_H × k)
RSCANn	Transmit/receive FIFO Buffer Access Pointer Register k	RSCANnCFPTRk	<RSCFDn_base> + $0E84_H$ + (10_H × k)
RSCANn	Transmit/receive FIFO Buffer Access Data Field 0 Register k	RSCANnCFDF0_k	<RSCFDn_base> + $0E88_H$ + (10_H × k)
RSCANn	Transmit/receive FIFO Buffer Access Data Field 1 Register k	RSCANnCFDF1_k	<RSCFDn_base> + $0E8C_H$ + (10_H × k)
FIFO status-related registers			
RSCANn	FIFO Empty Status Register	RSCANnFESTS	<RSCFDn_base> + 0238_H
RSCANn	FIFO Full Status Register	RSCANnFFSTS	<RSCFDn_base> + $023C_H$
RSCANn	FIFO Message Lost Status Register	RSCANnFMSTS	<RSCFDn_base> + 0240_H
RSCANn	Receive FIFO Buffer Interrupt Flag Status Register	RSCANnRFISTS	<RSCFDn_base> + 0244_H
RSCANn	Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register	RSCANnCFRISTS	<RSCFDn_base> + 0248_H
RSCANn	Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register	RSCANnCFTISTS	<RSCFDn_base> + $024C_H$

Table 14.11 Registers (3/3)

Module	Register	Symbol	Address
Transmit buffer-related registers			
RSCANn	Transmit Buffer Control Register p	RSCANnTMCp	<RSCFDn_base> + 0250_H + (01_H × p)
RSCANn	Transmit Buffer Status Register p	RSCANnTMSTSp	<RSCFDn_base> + $02D0_H$ + (01_H × p)
RSCANn	Transmit Buffer ID Register p	RSCANnTMIDp	<RSCFDn_base> + 1000_H + (10_H × p)
RSCANn	Transmit Buffer Pointer Register p	RSCANnTMPTRp	<RSCFDn_base> + 1004_H + (10_H × p)
RSCANn	Transmit Buffer Data Field 0 Register p	RSCANnTMDF0_p	<RSCFDn_base> + 1008_H + (10_H × p)
RSCANn	Transmit Buffer Data Field 1 Register p	RSCANnTMDF1_p	<RSCFDn_base> + $100C_H$ + (10_H × p)
RSCANn	Transmit Buffer Interrupt Enable Configuration Register y	RSCANnTMIECy	<RSCFDn_base> + 0390_H + (04_H × y)
Transmit buffer status-related registers			
RSCANn	Transmit Buffer Transmit Request Status Register y	RSCANnTMTRSTSy	<RSCFDn_base> + 0350_H + (04_H × y)
RSCANn	Transmit Buffer Transmit Abort Request Status Register y	RSCANnTMTARSTSy	<RSCFDn_base> + 0360_H + (04_H × y)
RSCANn	Transmit Buffer Transmit Complete Status Register y	RSCANnTMTCSTSy	<RSCFDn_base> + 0370_H + (04_H × y)
RSCANn	Transmit Buffer Transmit Abort Status Register y	RSCANnTMTASTSy	<RSCFDn_base> + 0380_H + (04_H × y)
Transmit queue-related registers			
RSCANn	Transmit Queue Configuration and Control Register m	RSCANnTXQCCm	<RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCANn	Transmit Queue Status Register m	RSCANnTXQSTSm	<RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCANn	Transmit Queue Pointer Control Register m	RSCANnTXQPCTRm	<RSCFDn_base> + $03E0_H$ + (04_H × m)
Transmit history-related registers			
RSCANn	Transmit History Configuration and Control Register m	RSCANnTHLCCm	<RSCFDn_base> + 0400_H + (04_H × m)
RSCANn	Transmit History Status Register m	RSCANnTHLSTSm	<RSCFDn_base> + 0420_H + (04_H × m)
RSCANn	Transmit History Pointer Control Register m	RSCANnTHLPCTRm	<RSCFDn_base> + 0440_H + (04_H × m)
RSCANn	Transmit History Access Register m	RSCANnTHLACCm	<RSCFDn_base> + 1800_H + (04_H × m)
Test-related registers			
RSCANn	Global Test Configuration Register	RSCANnGTSTCFG	<RSCFDn_base> + 0468_H
RSCANn	Global Test Control Register	RSCANnGTSTCTR	<RSCFDn_base> + $046C_H$
RSCANn	Global Lock Key Register	RSCANnGLOCKK	<RSCFDn_base> + $047C_H$
RSCANn	RAM Test Page Access Register r	RSCANnRPGACCr	<RSCFDn_base> + 1900_H + (04_H × r)

Table 14.12 Transmit Buffer p Allocated to Each Channel

	CANm
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
	Transmit buffer 16 × m + 15

Table 14.13 Transmit/Receive FIFO Buffer k Allocated to Each Channel

	CANm
Transmit/receive FIFO buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

Table 14.14 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
0000_B	Transmit buffer 16 × m + 0
0001_B	Transmit buffer 16 × m + 1
0010_B	Transmit buffer 16 × m + 2
0011_B	Transmit buffer 16 × m + 3
0100_B	Transmit buffer 16 × m + 4
0101_B	Transmit buffer 16 × m + 5
0110_B	Transmit buffer 16 × m + 6
0111_B	Transmit buffer 16 × m + 7
1000_B	Transmit buffer 16 × m + 8
1001_B	Transmit buffer 16 × m + 9
1010_B	Transmit buffer 16 × m + 10
1011_B	Transmit buffer 16 × m + 11
1100_B	Transmit buffer 16 × m + 12
1101_B	Transmit buffer 16 × m + 13
1110_B	Transmit buffer 16 × m + 14
1111_B	Transmit buffer 16 × m + 15

Table 14.15 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

Setting of Bits TXQDC [3:0]	Transmit Buffer p Allocated to the Transmit Queue
0000_B	Setting prohibited
0001_B	Setting prohibited
0010_B	Transmit buffer 16 × m + 15 to 16 × m + 13
0011_B	Transmit buffer 16 × m + 15 to 16 × m + 12
0100_B	Transmit buffer 16 × m + 15 to 16 × m + 11
0101_B	Transmit buffer 16 × m + 15 to 16 × m + 10
0110_B	Transmit buffer 16 × m + 15 to 16 × m + 9
0111_B	Transmit buffer 16 × m + 15 to 16 × m + 8
1000_B	Transmit buffer 16 × m + 15 to 16 × m + 7
1001_B	Transmit buffer 16 × m + 15 to 16 × m + 6
1010_B	Transmit buffer 16 × m + 15 to 16 × m + 5
1011_B	Transmit buffer 16 × m + 15 to 16 × m + 4
1100_B	Transmit buffer 16 × m + 15 to 16 × m + 3
1101_B	Transmit buffer 16 × m + 15 to 16 × m + 2
1110_B	Transmit buffer 16 × m + 15 to 16 × m + 1
1111_B	Transmit buffer 16 × m + 15 to 16 × m + 0

14.3.2 Details of Interface Mode-Related Registers

14.3.2.1 RSCANnGRMCFG – Global Interface Mode Select Register

Access: RSCANnGRMCFG register can be read/written in 32-bit units
RSCANnGRMCFGL, RSCANnGRMCFGH registers can be read/written in 16-bit units.
RSCANnGRMCFGLL, RSCANnGRMCFGLH, RSCANnGRMCFGHL, RSCANnGRMCFGHH registers can be read/written in 8-bit units.

Address: RSCANnGRMCFG: <RSCFDn_base> + $04FC_H$
RSCANnGRMCFGL: <RSCFDn_base> + $04FC_H$,
RSCANnGRMCFGH: <RSCFDn_base> + $04FE_H$
RSCANnGRMCFGLL: <RSCFDn_base> + $04FC_H$,
RSCANnGRMCFGLH: <RSCFDn_base> + $04FD_H$,
RSCANnGRMCFGHL: <RSCFDn_base> + $04FE_H$,
RSCANnGRMCFGHH: <RSCFDn_base> + $04FF_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RCMC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 14.16 RSCANnGRMCFG Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
0	RCMC	Interface Mode Select 0: Classical CAN mode 1: CAN FD mode

Note: RSCANnGRMCFG and RSCFDnCFDGRMCFG are the same register. Set either of these registers.

Modify the RSCANnGRMCFG register only in global reset mode. Before setting other RS-CANFD registers, set this register.

RCMC Bit

Setting this bit to 0 makes classical CAN mode available. To switch CAN FD mode to classical CAN mode, set the value after reset to all registers and bits allocated to the register map of CAN FD mode and then modify the RSCANnGRMCFG register.

14.3.3 Details of Channel-Related Registers

14.3.3.1 RSCANnCmCFG ─ Channel Configuration Register (m = 0 to 3)

Access: RSCANnCmCFG register can be read/written in 32-bit units
RSCANnCmCFGL, RSCANnCmCFGH registers can be read/written in 16-bit units
RSCANnCmCFGLL, RSCANnCmCFGLH, RSCANnCmCFGHL, RSCANnCmCFGHH registers can be read/written in 8-bit units.

Address: RSCANnCmCFG: <RSCFDn_base> + 0000_H + (10_H × m)
RSCANnCmCFGL: <RSCFDn_base> + 0000_H + (10_H × m),
RSCANnCmCFGH: <RSCFDn_base> + 0002_H + (10_H × m)
RSCANnCmCFGLL: <RSCFDn_base> + 0000_H + (10_H × m),
RSCANnCmCFGLH: <RSCFDn_base> + 0001_H + (10_H × m),
RSCANnCmCFGHL: <RSCFDn_base> + 0002_H + (10_H × m),
RSCANnCmCFGHH: <RSCFDn_base> + 0003_H + (10_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	SJW [1:0]		—	TSEG2[2:0]			TSEG1[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BRP[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.17 RSCANnCmCFG Register Contents

Bit Position	Bit Name	Function
31 to 26	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
25, 24	SJW[1:0]	Resynchronization Jump Width Control b25 b24 0 0: 1 Tq 0 1: 2 Tq 1 0: 3 Tq 1 1: 4 Tq
23	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
22 to 20	TSEG2[2:0]	Time Segment 2 Control b22 b21 b20 0 0 0: Setting prohibited 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq
19 to 16	TSEG1[3:0]	Time Segment 1 Control b19 b18 b17 b16 0 0 0 0: Setting prohibited 0 0 0 1: Setting prohibited 0 0 1 0: Setting prohibited 0 0 1 1: 4 Tq 0 1 0 0: 5 Tq 0 1 0 1: 6 Tq 0 1 1 0: 7 Tq 0 1 1 1: 8 Tq 1 0 0 0: 9 Tq 1 0 0 1: 10 Tq 1 0 1 0: 11 Tq 1 0 1 1: 12 Tq 1 1 0 0: 13 Tq 1 1 0 1: 14 Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to10	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
9 to 0	BRP[9:0]	Prescaler Division Ratio Set When these bits are set to P (0 to 1023), the baud rate prescaler divides fCAN by P + 1.

Modify the RSCANnCmCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode before shifting to channel communication mode or channel halt mode. For a description of the bit timing parameters and settings, see **Section 14.11.1, Initial Settings.**

SJW[1:0] Bits

These bits are used to specify a Tq value for the resynchronization jump width. Allowed values are 1 Tq to 4 Tq, inclusive. Set a value less than or equal to the value of the TSEG2 bits.

TSEG2[2:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2).

Allowed values are 2 Tq to 8 Tq, inclusive.

Set a value smaller than the value of the TSEG1 bits.

TSEG1[3:0] Bits

These bits are used to specify a Tq value for the total length of the propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1).

Allowed values are 4 Tq to 16 Tq, inclusive.

BRP[9:0] Bits

The CANmTq clock (fCANTQm) is calculated by dividing the CAN clock (fCAN) by the baud rate prescaler, ((BRP[9:0]) + 1). One clock cycle of the CANmTq clock is 1 Time Quantum (Tq).

14.3.3.2 RSCANnCmCTR ─ Channel Control Register (m = 0 to 3)

Access: RSCANnCmCTR register can be read/written in 32-bit units
RSCANnCmCTRL, RSCANnCmCTRH registers can be read/written in 16-bit units
RSCANnCmCTRLL, RSCANnCmCTRLH, RSCANnCmCTRHL, RSCANnCmCTRHH registers can be read/written in 8-bit units.

Address: RSCANnCmCTR: <RSCFDn_base> + 0004_H + (10_H × m)
RSCANnCmCTRL: <RSCFDn_base> + 0004_H + (10_H × m),
RSCANnCmCTRH: <RSCFDn_base> + 0006_H + (10_H × m)
RSCANnCmCTRLL: <RSCFDn_base> + 0004_H + (10_H × m),
RSCANnCmCTRLH: <RSCFDn_base> + 0005_H + (10_H × m),
RSCANnCmCTRHL: <RSCFDn_base> + 0006_H + (10_H × m),
RSCANnCmCTRHH: <RSCFDn_base> + 0007_H + (10_H × m)

Value after reset: $0000\ 0005_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCT	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	—	—	—	TAIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 14.18 RSCANnCmCTR Register Contents (1/2)

Bit Position	Bit Name	Function
31	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
30	CRCT	CRC Error Test Enable 0: The first bit of the reception ID field is not inverted. 1: The first bit of the reception ID field is inverted.
29 to 27	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
26, 25	CTMS[1:0]	Communication Test Mode Select b26 b25 0 0: Standard test mode 0 1: Listen-only mode 1 0: Self-test mode 0 (external loopback mode) 1 1: Self-test mode 1 (internal loopback mode)
24	CTME	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	Error Display Mode Select 0: Error flags are displayed only for the first error information after bits 14 to 8 in the RSCANnCmERFL register are all cleared. 1: Error flags for all error information are displayed.

Table 14.18 RSCANnCmCTR Register Contents (2/2)

Bit Position	Bit Name	Function
22, 21	BOM[1:0]	Bus Off Recovery Mode Select b22 b21 0 0 : ISO 11898-1:2015 compliant 0 1 :Entry to channel halt mode automatically at bus-off entry 1 0 :Entry to channel halt mode automatically at bus-off end 1 1 :Entry to channel halt mode (in bus-off state) by program request
20 to 17	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
16	TAIE	Transmit Abort Interrupt Enable 0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.
15	ALIE	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.
14	BLIE	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.
13	OLIE	Overload Frame Transmit Interrupt Enable 0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.
12	BORIE	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.
11	BOEIE	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
3	RTBO	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.
2	CSLPR	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1, 0	CHMDC[1:0]	Mode Select b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

CRCT Bit

This bit is used to test the CRC generation circuit in the RS-CANFD module. Setting this bit to 1 inverts the first bit of the ID field when a message is received. With this inversion of bit, the CRC calculation result does not match the normal CRC value of the received frame, which can detect a CRC error (the CERR bit in the RSCANnCmERFL register is 1). When using this function, note the following.

- This function is available while the CTME bit in the RSCANnCmCTR register is 1 (communication test mode enabled).
- This function cannot communicate with other CAN nodes. Use this function for inter-channel communication test (the CmICBCE bit in the RSCANnGTSTCFG register is 1)
- Bit inversion in the ID field may cause bit stuffing rule violation. In that case, no CRC error is detected but a stuff error is detected.

Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CTMS[1:0] Bits

These bits are used to select a communication test mode. Modify these bits in channel halt mode only. These bits are set to 0 in channel reset mode.

CTME Bit

Setting this bit to 1 enables communication test mode. Modify these bits in channel halt mode. This bit is set to 0 in channel reset mode.

ERRD Bit

This bit is used to control the display mode of bits 14 to 8 in the RSCANnCmERFL register.

When this bit is clear to 0, if any error is detected while the flags of bits 14-8 in the RSCANnCmERFL register are all 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1.

When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order.

Modify this bit only in channel reset mode or channel halt mode.

BOM[1:0] Bits

These bits are used to select the bus off recovery mode of the RS-CANFD module. When the BOM[1:0] bits are set to 00_B, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CANFD module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to 10_B (channel halt mode) before recessive bits are detected 128 times, the RS-CANFD module does not transition to channel halt mode until recessive bits are detected 128 times.

When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 01_B, the CHMDC[1:0] bits in the RSCANnCmCTR register (m = 0 to 3) are set to 10_B and the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCANnCmSTS register are cleared to 00_H.

When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 10_B, the CHMDC[1:0] bits are set to 10_B and the RS-CANFD module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H.

When the BOM[1:0] bits are set to 11_B and the CHMDC[1:0] bits are set to 10_B while the RS-CANFD module is in the bus off state, the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H. However, if 11 consecutive recessive bits are detected 128 times and the RS-CANFD module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10_B, a bus off recovery interrupt request is generated.

If a program writes to the CHMDC[1:0] bit at the same time as the RS-CANFD module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01_B or at bus off end when the BOM[1:0] bits are 10_B), the program's writing takes precedence. Modify the BOM[1:0] bits only in channel reset mode.

TAIE Bit

When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.

ALIE Bit

When the ALF flag in the RSCANnCmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BLIE Bit

When the BLF flag in the RSCANnCmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

OLIE Bit

When the OVLF flag in the RSCANnCmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BORIE Bit

When the BORF flag in the RSCANnCmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BOEIE Bit

When the BOEF flag in the RSCANnCmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EPIE Bit

When the EPF flag in the RSCANnCmERFLregister is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EWIE Bit

When the EWF flag in the RSCANnCmERFLregister is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BEIE Bit

When the BEF flag in the RSCANnCmERFLregister is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

RTBO Bit

Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCANnCmSTSregister to 00_H and also clears the BOSTS flag in the RSCANnCmSTSregister to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCANnCmCTRregister are 00_B (ISO 11898-1:2015 compliant).

A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RS-CANFD module transitions to the error active state. Set this bit to 1 in channel communication mode.

CSLPR Bit

Setting this bit to 1 places the channel into channel stop mode.

Clearing this bit to 0 makes the channel exit channel stop mode.

This bit should not be modified in channel communication mode or channel halt mode.

CHMDC[1:0] Bits

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see **Section 14.6.2, Channel Modes.** Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to 11_B. When the RS-CANFD module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become 10_B.

14.3.3.3 RSCANnCmSTS ─ Channel Status Register (m = 0 to 3)

Access: RSCANnCmSTS register can be read only in 32-bit units
RSCANnCmSTSL, RSCANnCmSTSH registers can be read only in 16-bit units
RSCANnCmSTSLL, RSCANnCmSTSLH, RSCANnCmSTSHL, RSCANnCmSTSHH registers can be read only in 8-bit units

Address: RSCANnCmSTS: <RSCFDn_base> + 0008_H + (10_H × m)
RSCANnCmSTSL: <RSCFDn_base> + 0008_H + (10_H × m),
RSCANnCmSTSH: <RSCFDn_base> + $000A_H$ + (10_H × m)
RSCANnCmSTSLL: <RSCFDn_base> + 0008_H + (10_H × m),
RSCANnCmSTSLH: <RSCFDn_base> + 0009_H + (10_H × m),
RSCANnCmSTSHL: <RSCFDn_base> + $000A_H$ + (10_H × m),
RSCANnCmSTSHH: <RSCFDn_base> + $000B_H$ + (10_H × m)

Value after reset: $0000\ 0005_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.19 RSCANnCmSTS Register Contents

Bit Position	Bit Name	Function
31 to 24	TEC[7:0]	The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	The receive error counter (REC) can be read.
15 to 8	Reserved	When read, the value after a reset is read.
7	COMSTS	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode
0	CRSTSTS	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

TEC[7:0] Bits

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1:2015).

These bits are cleared to 0 in channel reset mode.

REC[7:0] Bits

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1:2015).

These bits are cleared to 0 in channel reset mode.

COMSTS Flag

This bit indicates that communication is ready.

This flag becomes 1 when the RS-CANFD module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode.

This flag is cleared to 0 in channel reset mode or channel halt mode.

RECSTS Flag

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

TRMSTS Flag

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

BOSTS Flag

This flag is set to 1 when the bus off state (TEC[7:0] > 255) is entered. It is cleared to 0 when the RS-CANFD module has exited the bus off state.

EPSTS Flag

This flag is set to 1 when the RS-CANFD module has entered the error passive state ((128 ≤ TEC[7:0] ≤ 255) or (128 ≤ REC[7:0])), It is cleared to 0 when the RS-CANFD module has exited the error passive state or has entered channel reset mode.

CSLPSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to channel stop mode, and is cleared to 0 when the RS-CANFD module has returned from channel stop mode.

CHLTSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to channel halt mode, and is cleared to 0 when the RS-CANFD module has returned from channel halt mode.

CRSTSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to channel reset mode, and is cleared to 0 when the RS-CANFD module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the RS-CANFD module transitions from channel reset mode to channel stop mode.

14.3.3.4 RSCANnCmERFL ─ Channel Error Flag Register (m = 0 to 3)

Access: RSCANnCmERFL register can be read/written in 32-bit units
RSCANnCmERFLL, RSCANnCmERFLH registers can be read/written in 16-bit units
RSCANnCmERFLLL, RSCANnCmERFLLH, RSCANnCmERFLHL, RSCANnCmERFLHH registers can be read/written in 8-bit units

Address: RSCANnCmERFL: <RSCFDn_base> + $000C_H$ + (10_H × m)
RSCANnCmERFLL: <RSCFDn_base> + $000C_H$ + (10_H × m),
RSCANnCmERFLH: <RSCFDn_base> + $000E_H$ + (10_H × m)
RSCANnCmERFLLL: <RSCFDn_base> + $000C_H$ + (10_H × m),
RSCANnCmERFLLH: <RSCFDn_base> + $000D_H$ + (10_H × m),
RSCANnCmERFLHL: <RSCFDn_base> + $000E_H$ + (10_H × m),
RSCANnCmERFLHH: <RSCFDn_base> + $000F_H$ + (10_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.20 RSCANnCmERFL Register Contents (1/2)

Bit Position	Bit Name	Function
31	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
30 to 16	CRCREG[14:0]	CRC Calculation Data A CRC value calculated based on the transmit message or receive message is indicated.
15	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
14	ADERR	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.

Table 14.20 RSCANnCmERFL Register Contents (2/2)

Bit Position	Bit Name	Function
10	AERR	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.
9	FERR	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.
7	ALF	Arbitration-lost Flag 0: No arbitration-lost is detected. 1: Arbitration-lost is detected.
6	BLF	Bus Lock Flag 0: No channel bus is detected. 1: Channel bus is detected.
5	OVLF	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORF	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWF	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	Bus Error Flag 0: No channel bus error is detected. 1: Channel bus error is detected.

See the CAN specification (ISO11898-1:2015) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these flags is set to 0 at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0.

If the ERRD bit in the RSCANnCmCTR register is set to 0 (ie, only the flags for the first error event are displayed) and an error related to bits 14 to 8 of RSCANnCmERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the time when the error occurred.

CRCREG[14:0] Flags

When the CTME bit in the RSCANnCmCTR register is set to 1 (communication test mode is enabled), the CRC value calculated based on the transmit or receive message can be read. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0.

This bit is always 0 in channe reset mode.

ADERR Flag

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

B0ERR Flag

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

B1ERR Flag

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

CERR Flag

This flag is set to 1 when a CRC error has been detected.

AERR Flag

This flag is set to 1 when an ACK error has been detected.

FERR Flag

This flag is set to 1 when a form error has been detected.

SERR Flag

This flag is set to 1 when a stuff error has been detected.

ALF Flag

This flag is set to 1 when an arbitration-lost has been detected.

BLF Flag

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, detection of a dominant lock is restarted when either of the following conditions is met.

- A recessive bit is detected after the BLF bit has been cleared from 1 to 0.
- The RS-CANFD module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0.

OVLF Flag

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BORF Flag

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the RS-CANFD module returns from the bus off state. However, this flag is not set to 1 if the RS-CANFD module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCANnCmCTR register are set to 01_B (channel reset mode).
- The RTBO bit in the RSCANnCmCTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCANnCmCTR register are set to 01_B (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits are set to 10_B (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11_B (transition to channel halt mode upon a request from the program during bus off).

BOEF Flag

This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCANnCmCTR register (m = 0 to 3) set to 01_B (transition to channel halt mode at bus off entry).

EPF Flag

This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127). This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.

EWF Flag

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.

BEF Flag

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCANnCmERFL register is set to 1.

NOTE

To clear the flag of this register to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

14.3.4 Details of Global-Related Registers

14.3.4.1 RSCANnGCFG ─ Global Configuration Register

Access: RSCANnGCFG register can be read/written in 32-bit units
RSCANnGCFGL, RSCANnGCFGH registers can be read/written in 16-bit units
RSCANnGCFGLL, RSCANnGCFGLH, RSCANnGCFGHL, RSCANnGCFGHH registers can be read/written in 8-bit units

Address: RSCANnGCFG: <RSCFDn_base> + 0084_H
RSCANnGCFGL: <RSCFDn_base> + 0084_H, RSCANnGCFGH: <RSCFDn_base> + 0086_H
RSCANnGCFGLL: <RSCFDn_base> + 0084_H, RSCANnGCFGLH: <RSCFDn_base> + 0085_H,
RSCANnGCFGHL: <RSCFDn_base> + 0086_H, RSCANnGCFGHH: <RSCFDn_base> + 0087_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				TMTSCE	EEFE	—	DCS	MME	DRE	DCE	TPRI
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 14.21 RSCANnGCFG Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 16	ITRCP[15:0]	Interval Timer Prescaler Set When these bits are set to M, the pclk is divided by M. Setting 0000_H is prohibited when the interval timer is in use.
15 to 13	TSBTCS[2:0]	Timestamp Clock Source Select b15 b14 b13 0 0 0: Channel 0 bit time clock 0 0 1: Channel 1 bit time clock 0 1 0: Channel 2 bit time clock 0 1 1: Channel 3 bit time clock 1 0 0: Setting prohibited 1 0 1: Setting prohibited 1 1 0: Setting prohibited 1 1 1: Setting prohibited
12	TSSS	Timestamp Source Select 0: pclk/2*1 1: Bit time clock

Table 14.21 RSCANnGCFG Register Contents (2/2)

Bit Position	Bit Name	Function
11 to 8	TSP[3:0]	Timestamp Clock Source Division b11 b10 b9 b8 0 0 0 0: Not divided 0 0 0 1: Divided by 2 0 0 1 0: Divided by 4 0 0 1 1: Divided by 8 0 1 0 0: Divided by 16 0 1 0 1: Divided by 32 0 1 1 0: Divided by 64 0 1 1 1: Divided by 128 1 0 0 0: Divided by 256 1 0 0 1: Divided by 512 1 0 1 0: Divided by 1024 1 0 1 1: Divided by 2048 1 1 0 0: Divided by 4096 1 1 0 1: Divided by 8192 1 1 1 0: Divided by 16384 1 1 1 1: Divided by 32768
7	TMTSCE	Transmission Timestamp Enable 0: Transmission timestamp is disabled. 1: Transmission timestamp is enabled.
6	EEFE	ECC Error Flag Enable 0: The ECC error flag is disabled. 1: The ECC error flag is enabled.
5	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
4	DCS	CAN Clock Source Select 0: clkc 1: clk_xincan
3	MME	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000_B.

Modify the RSCANnGCFG register only in global reset mode.

ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. See **Section 14.8.3.1, Interval Transmission Function**.

TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the bit time clock that will be the clock source of the timestamp counter.

TSSS Bit

This bit is used to select a clock source of the timestamp counter.

TSP[3:0] Bits

A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.

TMTSCE Bit

Setting this bit to 1 makes it possible to store the timestamp of a transmitted message in the transmit history buffer. The timestamp is stored in TMTS[15:0] bits in the RSCANnTHLACCm register.

EEFE Bit

Setting this bit to 1 sets the EEFm bit in the RSCANnGERFL register to 1 when a 2-bit ECC error is detected during the transmission priority determination. At this time, the message in which a 2-bit ECC error was detected is not transmitted.

DCS Bit

When this bit is set to 0, clkc is used as the clock source of the CAN clock (fCAN).

When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (fCAN).

MME Bit

Setting this bit to 1 makes the mirror function available.

DRE Bit

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of 00_H is stored in each data byte beyond the DLC value of the receive rule.

The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

DCE Bit

Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCANnGAFLP0_j register to 0000_B before clearing the DCE bit in the RSCANnGCFG register to 0.

TPRI Bit

This bit is used to set the transmit priority.

When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1:2015 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those has the highest priority.

While the transmit queue is in use, this bit should be set to 0.

14.3.4.2 RSCANnGCTR ─ Global Control Register

Access: RSCANnGCTR register can be read/written in 32-bit units.
RSCANnGCTRL, RSCANnGCTRH registers can be read/written in 16-bit units.
RSCANnGCTRLL, RSCANnGCTRLH, RSCANnGCTRHL, RSCANnGCTRHH registers can be read/written in 8-bit units.

Address: RSCANnGCTR: <RSCFDn_base> + 0088_H
RSCANnGCTRL: <RSCFDn_base> + 0088_H, RSCANnGCTRH: <RSCFDn_base> + $008A_H$
RSCANnGCTRLL: <RSCFDn_base> + 0088_H, RSCANnGCTRLH: <RSCFDn_base> + 0089_H,
RSCANnGCTRHL: <RSCFDn_base> + $008A_H$, RSCANnGCTRHH: <RSCFDn_base> + $008B_H$

Value after reset: $0000\ 0005_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 14.22 RSCANnGCTR Register Contents

Bit Position	Bit Name	Function
31 to 17	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
16	TSRST	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 11	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
10	THLEIE	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	FIFO Message Lost Interrupt Enable 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	GSLPR	Global Stop Mode 0: Other than global stop mode 1: Global stop mode
1, 0	GMDC[1:0]	Global Mode Select b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCANnGTSC register is cleared to 0000_H.

THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in the RSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

MEIE Bit

When the MEIE bit is set to 1 and the MES flag in the RSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

DEIE Bit

When the DEIE bit is set to 1 and the DEF flag in the RSCANnGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

GSLPR Bit

Setting this bit to 1 places the RS-CANFD module into global stop mode.

Clearing this bit to 0 makes the RS-CANFD module leave from global stop mode.

This bit should not be modified in global operating mode or global test mode.

GMDC[1:0] Bits

These bits are used to select the mode of entire RS-CANFD module (global operating mode, global reset mode, or global test mode). For details, see **Section 14.6.1, Global Modes**. Setting the GSLPR bit to 1 when in global reset mode places the RS-CANFD module into global stop mode.

14.3.4.3 RSCANnGSTS ─ Global Status Register

Access: RSCANnGSTS register can be read only in 32-bit units.
RSCANnGSTSL, RSCANnGSTSH registers can be read only in 16-bit units.
RSCANnGSTSLL, RSCANnGSTSLH, RSCANnGSTSHL, RSCANnGSTSHH registers can be read only in 8-bit units.

Address: RSCANnGSTS: <RSCFDn_base> + $008C_H$
RSCANnGSTSL: <RSCFDn_base> + $008C_H$, RSCANnGSTSH: <RSCFDn_base> + $008E_H$
RSCANnGSTSLL: <RSCFDn_base> + $008C_H$, RSCANnGSTSLH: <RSCFDn_base> + $008D_H$,
RSCANnGSTSHL: <RSCFDn_base> + $008E_H$, RSCANnGSTSHH: <RSCFDn_base> + $008F_H$

Value after reset: 0000 $000D_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAM INIT	GSLP STS	GHLT STS	GRST STS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.23 RSCANnGSTS Register Contents

Bit Position	Bit Name	Function
31 to 4	Reserved	When read, the value after reset is returned.
3	GRAMINIT	CAN RAM Initialization Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.

This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.

GSLPSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to global stop mode, and is cleared to 0 when the RS-CANFD module has returned from global stop mode.

GHLTSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to global test mode, and is cleared to 0 when the RS-CANFD module has exited global test mode.

GRSTSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to global reset mode, and is cleared to 0 when the RS-CANFD module has exited global reset mode. This flag remains 1 even when the RS-CANFD module has transitioned from global reset mode to global stop mode.

14.3.4.4 RSCANnGERFL – Global Error Flag Register

Access: RSCANnGERFL register can be read/written in 32-bit units.
RSCANnGERFLL, RSCANnGERFLH registers can be read/written in 16-bit units.
RSCANnGERFLLL, RSCANnGERFLLH, RSCANnGERFLHL, RSCANnGERFLHH registers can be read/written in 8-bit units.

Address: RSCANnGERFL: <RSCFDn_base> + 0090_H
RSCANnGERFLL: <RSCFDn_base> + 0090_H, RSCANnGERFLH: <RSCFDn_base> + 0092_H
RSCANnGERFLLL: <RSCFDn_base> + 0090_H, RSCANnGERFLLH: <RSCFDn_base> + 0091_H,
RSCANnGERFLHL: <RSCFDn_base> + 0092_H, RSCANnGERFLHH: <RSCFDn_base> + 0093_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	EEF3	EEF2	EEF1	EEF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R/W *1	R/W *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	THLES	MES	DEF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.24 RSCANnGERFL Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
19	EEF3	ECC Error Flag for Channel 3 0: No 2-bit ECC error when deciding transmission priority 1: A 2-bit ECC error when deciding transmission priority
18	EEF2	ECC Error Flag for Channel 2 0: No 2-bit ECC error when deciding transmission priority 1: A 2-bit ECC error when deciding transmission priority
17	EEF1	ECC Error Flag for Channel 1 0: No 2-bit ECC error when deciding transmission priority 1: A 2-bit ECC error when deciding transmission priority
16	EEF0	ECC Error Flag for Channel 0 0: No 2-bit ECC error when deciding transmission priority 1: A 2-bit ECC error when deciding transmission priority
15 to 3	Reserved	When read, the undefined value is returned. When writing these bits, write the value after reset.
2	THLES	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.
1	MES	FIFO Message Lost Status Flag 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.
0	DEF	DLC Error Flag 0: No DLC error has occurred. 1: A DLC error has occurred.

All flags in the RSCANnGERFL register are cleared to 0 in global reset mode.

EEFm Flag

While the EEFE bit in the RSCANnGCFG register is 1, when a 2-bit ECC error is detected during the transmission priority determination of channel m (m = 0 to 3), the EEFm flag is set to 1, disabling message transmission. This flag can be cleared to 0 by writing 0 by the program.

THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCANnTHLSTSm register (m = 0 to 3) is set to 1.

This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in the RSCANnRFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCANnCFSTSk register (k = 0 to 11) is set to 1.

This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.

NOTE

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

14.3.4.5 RSCANnGTSC ─ Global Timestamp Counter Register

Access: RSCANnGTSC register can be read only in 32-bit units.
RSCANnGTSCL, RSCANnGTSCH registers can be read only in 16-bit units.

Address: RSCANnGTSC: <RSCFDn_base> + 0094_H
RSCANnGTSCL: <RSCFDn_base> + 0094_H, RSCANnGTSCH: <RSCFDn_base> + 0096_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.25 RSCANnGTSC Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned.
15 to 0	TS[15:0]	Timestamp Value The timestamp counter value can be read. Counter Value: 0000_H to $FFFF_H$

TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. Furthermore, while the TMTSCE bit in the RSCANnGCFG register is 1, the TS[15:0] value is stored in the transmit history buffer. The timestamp counter is initialized in global reset mode.

The timestamp counter start timing and stop timing depend on the count source.

- When the TSSS bit in the RSCANnGCFG register is 0 (pclk):
  The timestamp counter starts counting when the RS-CANFD module has transitioned to global operating mode.
  This counter stops counting when the RS-CANFD module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CANm bit time clock):
  The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode.
  This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

14.3.4.6 RSCANnGTINTSTS0 ─ Global TX Interrupt Status Register 0

Access: RSCANnGTINTSTS0 register can be read only in 32-bit units.
RSCANnGTINTSTS0L, RSCANnGTINTSTS0H registers can be read only in 16-bit units.
RSCANnGTINTSTS0LL, RSCANnGTINTSTS0LH, RSCANnGTINTSTS0HL, RSCANnGTINTSTS0HH registers can be read only in 8-bit units.

Address: RSCANnGTINTSTS0: <RSCFDn_base> + 0460_H
RSCANnGTINTSTS0L: <RSCFDn_base> + 0460_H, RSCANnGTINTSTS0H: <RSCFDn_base> + 0462_H
RSCANnGTINTSTS0LL: <RSCFDn_base> + 0460_H, RSCANnGTINTSTS0LH: <RSCFDn_base> + 0461_H,
RSCANnGTINTSTS0HL: <RSCFDn_base> + 0462_H, RSCANnGTINTSTS0HH: <RSCFDn_base> + 0463_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	THIF3	CFTIF3	TQIF3	TAIF3	TSIF3	—	—	—	THIF2	CFTIF2	TQIF2	TAIF2	TSIF2
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

Table 14.26 RSCANnGTINTSTS0 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 29	Reserved	When read, the value after reset is returned.
28	THIF3	Channel 3 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
27	CFTIF3	Channel 3 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
26	TQIF3	Channel 3 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
25	TAIF3	Channel 3 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
24	TSIF3	Channel 3 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
23 to 21	Reserved	When read, the value after reset is returned.
20	THIF2	Channel 2 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
19	CFTIF2	Channel 2 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.

Table 14.26 RSCANnGTINTSTS0 Register Contents (2/2)

Bit Position	Bit Name	Function
18	TQIF2	Channel 2 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
17	TAIF2	Channel 2 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
16	TSIF2	Channel 2 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
15 to 13	Reserved	When read, the value after a reset is read.
12	THIF1	Channel 1 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
11	CFTIF1	Channel 1 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
10	TQIF1	Channel 1 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
9	TAIF1	Channel 1 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
8	TSIF1	Channel 1 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
7 to 5	Reserved	When read, the value after a reset is read.
4	THIF0	Channel 0 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF0	Channel 0 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
2	TQIF0	Channel 0 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TAIF0	Channel 0 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
0	TSIF0	Channel 0 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

TSIFm Bits

The TSIFm bit is set to 1 when the TMIEp bit in the RSCANnTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCANnTMSTSp register are set to 10_B (transmit completed without abort request) or 11_B (transmit completed with abort request).

When the TMTRF[1:0] flags are cleared to 00_B under the condition that the TSIFm bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.

TAIFm Bits

The TAIFm bit is set to 1 when the TAIE bit in the RSCANnCmCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCANnTMSTSp register are set to 01_B (transmit abort completed).

This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00_B after the transmit abort is completed.

TQIFm Bits

When the TXQIE bit in the RSCANnTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCANnTXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFm bit is set to 1.

When the TXQIF bit (transmit queue interrupt request) in the RSCANnTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.

CFTIFm Bits

When the CFTXIE bit in the RSCANnCFCCk register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCANnCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFm bit is set to 1.

When the CFTXIF bit is cleared to 0 under the conditions that the CFTIFm bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.

THIFm Bits

When the THLIE bit in the RSCANnTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCANnTHLSTSm register is set to 1 (transmit history interrupt request), the THIFm bit is set to 1.

When the THLIF bit in the RSCANnTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.

14.3.4.7 RSCANnGFDCFG ─ Global FD Configuration Register

Access: RSCANnGFDCFG register can be read/written in 32-bit units.
RSCANnGFDCFG L, RSCANnGFDCFG H registers can be read/written in 16-bit units.
RSCANnGFDCFG LL, RSCANnGFDCFG LH, RSCANnGFDCFG HL, RSCANnGFDCFG HH registers can be read/written in 8-bit units.

Address: RSCANnGFDCFG: <RSCFDn_base> + 0474_H
RSCANnGFDCFG L: <RSCFDn_base> + 0474_H, RSCANnGFDCFG H: <RSCFDn_base> + 0476_H
RSCANnGFDCFG LL: <RSCFDn_base> + 0474_H, RSCANnGFDCFG LH: <RSCFDn_base> + 0475_H,
RSCANnGFDCFG HL: <RSCFDn_base> + 0476_H, RSCANnGFDCFG HH: <RSCFDn_base> + 0477_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSCCFG[1:0]		—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Table 14.27 RSCANnGFDCFG Register Contents

Bit Position	Bit Name	Function
31 to 10	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
9, 8	TSCCFG[1:0]	Time-stamp capture setting bit b 9 b8 0 0: Captured at a sample point in the SOF bit. 0 1: Captured when a valid frame has been transmitted/received. 1 0: Setting prohibited 1 1: Setting prohibited
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	Reserved	When read, the undefined value is returned. When writing this bit, write the value after reset.

TSCCFG bit

Select a point where a time-stamp value is caputured. Modify this bit only in global reset mode.

14.3.5 Details of Receive Rule-Related Registers

14.3.5.1 RSCANnGAFLECTR ─ Receive Rule Entry Control Register

Access: RSCANnGAFLECTR register can be read/written in 32-bit units.
RSCANnGAFLECTRL, RSCANnGAFLECTRH registers can be read/written in 16-bit units.
RSCANnGAFLECTRLL, RSCANnGAFLECTRLH, RSCANnGAFLECTRHL, RSCANnGAFLECTRHH registers can be read/written in 8-bit units.

Address: RSCANnGAFLECTR: <RSCFDn_base> + 0098_H
RSCANnGAFLECTRL: <RSCFDn_base> + 0098_H, RSCANnGAFLECTRH: <RSCFDn_base> + $009A_H$
RSCANnGAFLECTRLL: <RSCFDn_base> + 0098_H, RSCANnGAFLECTRLH: <RSCFDn_base> + 0099_H,
RSCANnGAFLECTRHL: <RSCFDn_base> + $009A_H$, RSCANnGAFLECTRHH: <RSCFDn_base> + $009B_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFLDAE	—	—	—	AFLPN[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 14.28 RSCANnGAFLECTR Register Contents

Bit Position	Bit Name	Function
31 to 9	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
8	AFLDAE	Receive Rule Table Write Enable 0: Receive rule table write is disabled. 1: Receive rule table write is enabled.
7 to 5	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
4 to 0	AFLPN[4:0]	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000_B) to page 15 (01111_B).

AFLDAE Bit

Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit.

Set the AFLDAE bit to 1 only in global reset mode.

AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page.

Set these bits to a value within the range of 00000_B to 01111_B.

14.3.5.2 RSCANnGAFLCFG0 ─ Receive Rule Configuration Register 0

Access: RSCANnGAFLCFG0 register can be read/written in 32-bit units.
RSCANnGAFLCFG0L, RSCANnGAFLCFG0H registers can be read/written in 16-bit units.
RSCANnGAFLCFG0LL, RSCANnGAFLCFG0LH, RSCANnGAFLCFG0HL, RSCANnGAFLCFG0HH registers can be read/written in 8-bit units.

Address: RSCANnGAFLCFG0: <RSCFDn_base> + $009C_H$
RSCANnGAFLCFG0L: <RSCFDn_base> + $009C_H$, RSCANnGAFLCFG0H: <RSCFDn_base> + $009E_H$
RSCANnGAFLCFG0LL: <RSCFDn_base> + $009C_H$, RSCANnGAFLCFG0LH: <RSCFDn_base> + $009D_H$,
RSCANnGAFLCFG0HL: <RSCFDn_base> + $009E_H$, RSCANnGAFLCFG0HH: <RSCFDn_base> + $009F_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RNC2[7:0]								RNC3[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.29 RSCANnGAFLCFG0 Register Contents

Bit Position	Bit Name	Function
31 to 24	RNC0[7:0]	Number of Rules for Channel 0 Set the number of receive rules exclusively used for channel 0.
23 to 16	RNC1[7:0]	Number of Rules for Channel 1 Set the number of receive rules exclusively used for channel 1.
15 to 8	RNC2[7:0]	Number of Rules for Channel 2 Set the number of receive rules exclusively used for channel 2.
7 to 0	RNC3[7:0]	Number of Rules for Channel 3 Set the number of receive rules exclusively used for channel 3.

Modify the RSCANnGAFLCFG0 register only in global reset mode.

Up to 64 × (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC2[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 2 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC3[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 3 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

14.3.5.3 RSCANnGAFLIDj ─ Receive Rule ID Register (j = 0 to 15)

Access: RSCANnGAFLIDj register can be read/written in 32-bit units.
RSCANnGAFLIDjL, RSCANnGAFLIDjH registers can be read/written in 16-bit units.
RSCANnGAFLIDjLL, RSCANnGAFLIDjLH, RSCANnGAFLIDjHL, RSCANnGAFLIDjHH registers can be read/written in 8-bit units.

Address: RSCANnGAFLIDj: <RSCFDn_base> + 0500_H + (10_H × j)
RSCANnGAFLIDjL: <RSCFDn_base> + 0500_H + (10_H × j),
RSCANnGAFLIDjH: <RSCFDn_base> + 0502_H + (10_H × j)
RSCANnGAFLIDjLL: <RSCFDn_base> + 0500_H + (10_H × j),
RSCANnGAFLIDjLH: <RSCFDn_base> + 0501_H + (10_H × j),
RSCANnGAFLIDjHL: <RSCFDn_base> + 0502_H + (10_H × j),
RSCANnGAFLIDjHH: <RSCFDn_base> + 0503_H + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.30 RSCANnGAFLIDj Register Contents

Bit Position	Bit Name	Function
31	GAFLIDE	IDE Select 0: Standard ID 1: Extended ID
30	GAFLRTR	RTR Select 0: Data frame 1: Remote frame
29	GAFLLB	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received
28 to 0	GAFLID[28:0]	ID Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCANnGAFLIDj register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLLB Bit

When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.

When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.

GAFLID[28:0] Bits

These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

14.3.5.4 RSCANnGAFLMj ─ Receive Rule Mask Register (j = 0 to 15)

Access: RSCANnGAFLMj register can be read/written in 32-bit units.
RSCANnGAFLMjL, RSCANnGAFLMjH registers can be read/written in 16-bit units.
RSCANnGAFLMjLL, RSCANnGAFLMjLH, RSCANnGAFLMjHL, RSCANnGAFLMjHH registers can be read/written in 8-bit units.

Address: RSCANnGAFLMj: <RSCFDn_base> + 0504_H + (10_H × j)
RSCANnGAFLMjL: <RSCFDn_base> + 0504_H + (10_H × j),
RSCANnGAFLMjH: <RSCFDn_base> + 0506_H + (10_H × j)
RSCANnGAFLMjLL: <RSCFDn_base> + 0504_H + (10_H × j),
RSCANnGAFLMjLH: <RSCFDn_base> + 0505_H + (10_H × j),
RSCANnGAFLMjHL: <RSCFDn_base> + 0506_H + (10_H × j),
RSCANnGAFLMjHH: <RSCFDn_base> + 0507_H + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDEM	GAFLRTRM	—	GAFLIDM[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.31 RSCANnGAFLMj Register Contents

Bit Position	Bit Name	Function
31	GAFLIDEM	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared.
29	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
28 to 0	GAFLIDM[28:0]	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCANnGAFLMj register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCANnGAFLIDj register.

When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bit of the receive rule.

14.3.5.5 RSCANnGAFLP0_j ─ Receive Rule Pointer 0 Register (j = 0 to 15)

Access: RSCANnGAFLP0_j register can be read/written in 32-bit units.
RSCANnGAFLP0_jL, RSCANnGAFLP0_jH registers can be read/written in 16-bit units.
RSCANnGAFLP0_jLL, RSCANnGAFLP0_jLH, RSCANnGAFLP0_jHL, RSCANnGAFLP0_jHH registers can be read/written in 8-bit units.

Address: RSCANnGAFLP0_j: <RSCFDn_base> + 0508_H + (10_H × j)
RSCANnGAFLP0_jL: <RSCFDn_base> + 0508_H + (10_H × j),
RSCANnGAFLP0_jH: <RSCFDn_base> + $050A_H$ + (10_H × j)
RSCANnGAFLP0_jLL: <RSCFDn_base> + 0508_H + (10_H × j),
RSCANnGAFLP0_jLH: <RSCFDn_base> + 0509_H + (10_H × j),
RSCANnGAFLP0_jHL: <RSCFDn_base> + $050A_H$ + (10_H × j),
RSCANnGAFLP0_jHH: <RSCFDn_base> + $050B_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFL RMV	GAFLRMDP[6:0]							—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 14.32 RSCANnGAFLP0_j Register Contents

Bit Position	Bit Name	Function
31 to 28	GAFLDLC[3:0]	Receive Rule DLC b31 b30 b29 b28 0 0 0 0: DLC check is disabled. 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	GAFLPTR[11:0]	Receive Rule Label Set the 12-bit label information.
15	GAFLRMV	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP[6:0]	Receive Buffer Number Select Set the receive buffer number to store receive messages.
7 to 0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Modify the RSCANnGAFLP0_j register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to 0000_B disables the DLC check function allowing messages with any data length to pass the DLC check.

GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

GAFLRMV Bit

When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLRMDP[6:0] Bits

These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCANnRMNB register.

14.3.5.6 RSCANnGAFLP1_j ─ Receive Rule Pointer 1 Register (j = 0 to 15)

Access: RSCANnGAFLP1_j register can be read/written in 32-bit units.
RSCANnGAFLP1_jL, RSCANnGAFLP1_jH registers can be read/written in 16-bit units.
RSCANnGAFLP1_jLL, RSCANnGAFLP1_jLH, RSCANnGAFLP1_jHL, RSCANnGAFLP1_jHH registers can be read/written in 8-bit units.

Address: RSCANnGAFLP1_j: <RSCFDn_base> + $050C_H$ + (10_H × j)
RSCANnGAFLP1_jL: <RSCFDn_base> + $050C_H$ + (10_H × j),
RSCANnGAFLP1_jH: <RSCFDn_base> + $050E_H$ + (10_H × j)
RSCANnGAFLP1_jLL: <RSCFDn_base> + $050C_H$ + (10_H × j),
RSCANnGAFLP1_jLH: <RSCFDn_base> + $050D_H$ + (10_H × j),
RSCANnGAFLP1_jHL: <RSCFDn_base> + $050E_H$ + (10_H × j),
RSCANnGAFLP1_jHH: <RSCFDn_base> + $050F_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	GAFLFDP[19:16]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLFDP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.33 RSCANnGAFLP1_j Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
19 to 8	GAFLFDP[19:8]	Transmit/Receive FIFO Buffer k Select (Bit position –8 = target transmit/receive FIFO buffer number k) 0: Transmit/receive FIFO buffer is not selected. 1: Transmit/receive FIFO buffer is selected.
7 to 0	GAFLFDP[7:0]	Receive FIFO Buffer x Select (Bit position = target receive FIFO buffer number x) 0: Receive FIFO buffer is not selected. 1: Receive FIFO buffer is selected.

Modify the RSCANnGAFLP1_j register when the AFLDAE bit in the RSCANnGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP [19:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCANnGAFLP0_j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected. Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCANnCFCCk register are set to 00_B (receive mode) or 10_B (gateway mode) are selectable.

14.3.6 Details of Receive Buffer-Related Registers

14.3.6.1 RSCANnRMNB ─ Receive Buffer Number Register

Access: RSCANnRMNB register can be read/written in 32-bit units.
RSCANnRMNBL, RSCANnRMNBH registers can be read/written in 16-bit units.
RSCANnRMNBLL, RSCANnRMNBLH, RSCANnRMNBHL, RSCANnRMNBHH registers can be read/written in 8-bit units.

Address: RSCANnRMNB: <RSCFDn_base> + $00A4_H$
RSCANnRMNBL: <RSCFDn_base> + $00A4_H$, RSCANnRMNBH: <RSCFDn_base> + $00A6_H$
RSCANnRMNBLL: <RSCFDn_base> + $00A4_H$, RSCANnRMNBLH: <RSCFDn_base> + $00A5_H$,
RSCANnRMNBHL: <RSCFDn_base> + $00A6_H$, RSCANnRMNBHH: <RSCFDn_base> + $00A7_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	NRXMB[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.34 RSCANnRMNB Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
7 to 0	NRXMB[7:0]	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 64.

Modify the RSCANnRMNB register only in global reset mode.

NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the RS-CANFD module. The maximum value is 16 × (number of channels).

Setting these bits all to 0 makes receive buffers unavailable.

14.3.6.2 RSCANnRMNDy ─ Receive Buffer New Data Register (y = 0, 1)

Access: RSCANnRMNDy register can be read/written in 32-bit units.
RSCANnRMNDyL, RSCANnRMNDyH registers can be read/written in 16-bit units.
RSCANnRMNDyLL, RSCANnRMNDyLH, RSCANnRMNDyHL, RSCANnRMNDyHH registers can be read/written in 8-bit units.

Address: RSCANnRMNDy: <RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCANnRMNDyL: <RSCFDn_base> + $00A8_H$ + (04_H × y),
RSCANnRMNDyH: <RSCFDn_base> + $00AA_H$ + (04_H × y)
RSCANnRMNDyLL: <RSCFDn_base> + $00A8_H$ + (04_H × y),
RSCANnRMNDyLH: <RSCFDn_base> + $00A9_H$ + (04_H × y),
RSCANnRMNDyHL: <RSCFDn_base> + $00AA_H$ + (04_H × y),
RSCANnRMNDyHH: <RSCFDn_base> + $00AB_H$ + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.35 RSCANnRMNDy Register Contents

Bit Position	Bit Name	Function
31 to 16	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.
15 to 0	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.

Write 0 to the RSCANnRMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 63)

Each RMNSq flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.

To clear a flag to 0, the program must write 0 to the flag. Use a store instruction to write “0” to the flag and “1” to other flags. These bits cannot be set to 0 while a message is being stored. It takes ten clock cycles of pclk to store a message.

These flags are cleared to 0 in global reset mode.

14.3.6.3 RSCANnRMIDq — Receive Buffer ID Register (q = 0 to 63)

Access: RSCANnRMIDq register can be read only in 32-bit units.
RSCANnRMIDqL, RSCANnRMIDqH registers can be read only in 16-bit units.
RSCANnRMIDqLL, RSCANnRMIDqLH, RSCANnRMIDqHL, RSCANnRMIDqHH registers can be read only in 8-bit units.

Address: RSCANnRMIDq: <RSCFDn_base> + 0600_H + (10_H × q)
RSCANnRMIDqL: <RSCFDn_base> + 0600_H + (10_H × q),
RSCANnRMIDqH: <RSCFDn_base> + 0602_H + (10_H × q)
RSCANnRMIDqLL: <RSCFDn_base> + 0600_H + (10_H × q),
RSCANnRMIDqLH: <RSCFDn_base> + 0601_H + (10_H × q),
RSCANnRMIDqHL: <RSCFDn_base> + 0602_H + (10_H × q),
RSCANnRMIDqHH: <RSCFDn_base> + 0603_H + (10_H × q)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.36 RSCANnRMIDq Register Contents

Bit Position	Bit Name	Function
31	RMIDE	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	Receive Buffer RTR 0: Data frame 1: Remote frame
29	Reserved	When read, the value after reset is returned.
28 to 0	RMID[28:0]	Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RMIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer.

RMRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer.

RMID[28:0] Bits

These bits contain the ID of the message stored in the receive buffer.

14.3.6.4 RSCANnRMPTRq — Receive Buffer Pointer Register (q = 0 to 63)

Access: RSCANnRMPTRq register can be read only in 32-bit units.
RSCANnRMPTRqL, RSCANnRMPTRqH registers can be read only in 16-bit units.
RSCANnRMPTRqLL, RSCANnRMPTRqLH, RSCANnRMPTRqHL, RSCANnRMPTRqHH registers can be read only in 8-bit units.

Address: RSCANnRMPTRq: <RSCFDn_base> + 0604_H + (10_H × q)
RSCANnRMPTRqL: <RSCFDn_base> + 0604_H + (10_H × q),
RSCANnRMPTRqH: <RSCFDn_base> + 0606_H + (10_H × q)
RSCANnRMPTRqLL: <RSCFDn_base> + 0604_H + (10_H × q),
RSCANnRMPTRqLH: <RSCFDn_base> + 0605_H + (10_H × q),
RSCANnRMPTRqHL: <RSCFDn_base> + 0606_H + (10_H × q),
RSCANnRMPTRqHH: <RSCFDn_base> + 0607_H + (10_H × q)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.37 RSCANnRMPTRq Register Contents

Bit Position	Bit Name	Function
31 to 28	RMDLC[3:0]	Receive Buffer DLC Data b31 b30 b29 b28 0 0 0 0: No data byte 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RMPTR[11:0]	Receive Buffer Label Data Label information of the received message.
15 to 0	RMTS[15:0]	Receive Buffer Timestamp Data Timestamp value of the received message.

RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer.

RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

14.3.6.5 RSCANnRMDF0_q — Receive Buffer Data Field 0 Register (q = 0 to 63)

Access: RSCANnRMDF0_q register can be read only in 32-bit units.
RSCANnRMDF0_qL, RSCANnRMDF0_qH registers can be read only in 16-bit units.
RSCANnRMDF0_qLL, RSCANnRMDF0_qLH, RSCANnRMDF0_qHL, RSCANnRMDF0_qHH registers can be read only in 8-bit units.

Address: RSCANnRMDF0_q: <RSCFDn_base> + 0608_H + (10_H × q)
RSCANnRMDF0_qL: <RSCFDn_base> + 0608_H + (10_H × q),
RSCANnRMDF0_qH: <RSCFDn_base> + $060A_H$ + (10_H × q)
RSCANnRMDF0_qLL: <RSCFDn_base> + 0608_H + (10_H × q),
RSCANnRMDF0_qLH: <RSCFDn_base> + 0609_H + (10_H × q),
RSCANnRMDF0_qHL: <RSCFDn_base> + $060A_H$ + (10_H × q),
RSCANnRMDF0_qHH: <RSCFDn_base> + $060B_H$ + (10_H × q)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB3[7:0]								RMDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB1[7:0]								RMDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.38 RSCANnRMDF0_q Register Contents

Bit Position	Bit Name	Function
31 to 24	RMDB3[7:0]	Receive Buffer Data Byte 3
23 to 16	RMDB2[7:0]	Receive Buffer Data Byte 2
15 to 8	RMDB1[7:0]	Receive Buffer Data Byte 1
7 to 0	RMDB0[7:0]	Receive Buffer Data Byte 0 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCANnRMPTRq register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

14.3.6.6 RSCANnRMDF1_q — Receive Buffer Data Field 1 Register (q = 0 to 63)

Access: RSCANnRMDF1_q register can be read only in 32-bit units.
RSCANnRMDF1_qL, RSCANnRMDF1_qH registers can be read only in 16-bit units.
RSCANnRMDF1_qLL, RSCANnRMDF1_qLH, RSCANnRMDF1_qHL, RSCANnRMDF1_qHH registers can be read only in 8-bit units.

Address: RSCANnRMDF1_q: <RSCFDn_base> + $060C_H$ + (10_H × q)
RSCANnRMDF1_qL: <RSCFDn_base> + $060C_H$ + (10_H × q),
RSCANnRMDF1_qH: <RSCFDn_base> + $060E_H$ + (10_H × q)
RSCANnRMDF1_qLL: <RSCFDn_base> + $060C_H$ + (10_H × q),
RSCANnRMDF1_qLH: <RSCFDn_base> + $060D_H$ + (10_H × q),
RSCANnRMDF1_qHL: <RSCFDn_base> + $060E_H$ + (10_H × q),
RSCANnRMDF1_qHH: <RSCFDn_base> + $060F_H$ + (10_H × q)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB7[7:0]								RMDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB5[7:0]								RMDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.39 RSCANnRMDF1_q Register Contents

Bit Position	Bit Name	Function
31 to 24	RMDB7[7:0]	Receive Buffer Data Byte 7
23 to 16	RMDB6[7:0]	Receive Buffer Data Byte 6
15 to 8	RMDB5[7:0]	Receive Buffer Data Byte 5
7 to 0	RMDB4[7:0]	Receive Buffer Data Byte 4 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCANnRMPTRq register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

14.3.7 Details of Receive FIFO Buffer-Related Registers

14.3.7.1 RSCANnRFCCx ─ Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

Access: RSCANnRFCCx register can be read/written in 32-bit units.
RSCANnRFCCxL, RSCANnRFCCxH registers can be read/written in 16-bit units.
RSCANnRFCCxLL, RSCANnRFCCxLH, RSCANnRFCCxHL, RSCANnRFCCxHH registers can be read/written in 8-bit units.

Address: RSCANnRFCCx: <RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCANnRFCCxL: <RSCFDn_base> + $00B8_H$ + (04_H × x),
RSCANnRFCCxH: <RSCFDn_base> + $00BA_H$ + (04_H × x)
RSCANnRFCCxLL: <RSCFDn_base> + $00B8_H$ + (04_H × x),
RSCANnRFCCxLH: <RSCFDn_base> + $00B9_H$ + (04_H × x),
RSCANnRFCCxHL: <RSCFDn_base> + $00BA_H$ + (04_H × x),
RSCANnRFCCxHH: <RSCFDn_base> + $00BB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	—	—	—	—	—	RFIE	RFE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 14.40 RSCANnRFCCx Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
15 to 13	RFIGCV[2:0]	Receive FIFO Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
10 to 8	RFDC[2:0]	Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	RFIE	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.
0	RFE	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

RFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]).

When the RFDC[2:0] bits are set to 001_B (4 messages), set the RFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B. Modify these bits only in global reset mode.

RFIM Bit

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

RFDC[2:0] Bits

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. When these bits are set to 000_B, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

RFIE Bit

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).

RFE Bit

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCANnRFSTSx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode.

Set this bit to 1 with another instruction after the settings to all bits in the RSCANnRFCCx register have been done.

This bit is cleared to 0 in global reset mode.

14.3.7.2 RSCANnRFSTSx – Receive FIFO Buffer Status Register (x = 0 to 7)

Access: RSCANnRFSTSx register can be read/written in 32-bit units.
RSCANnRFSTSxL, RSCANnRFSTSxH registers can be read/written in 16-bit units.
RSCANnRFSTSxLL, RSCANnRFSTSxLH, RSCANnRFSTSxHL, RSCANnRFSTSxHH registers can be read/written in 8-bit units.

Address: RSCANnRFSTSx: <RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCANnRFSTSxL: <RSCFDn_base> + $00D8_H$ + (04_H × x),
RSCANnRFSTSxH: <RSCFDn_base> + $00DA_H$ + (04_H × x)
RSCANnRFSTSxLL: <RSCFDn_base> + $00D8_H$ + (04_H × x),
RSCANnRFSTSxLH: <RSCFDn_base> + $00D9_H$ + (04_H × x),
RSCANnRFSTSxHL: <RSCFDn_base> + $00DA_H$ + (04_H × x),
RSCANnRFSTSxHH: <RSCFDn_base> + $00DB_H$ + (04_H × x)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.41 RSCANnRFSTSx Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
15 to 8	RFMC[7:0]	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	RFIF	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).

RFMC[7:0] Flags

This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes 00_H when the RFE bit in the RSCANnRFCCx register is set to 0.

RFIF Flag

This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCANnRFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFMLT Flag

This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.

This flag is cleared to 0 in global reset mode or by writing 0 to this flag.

Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFFLL Flag

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCANnRFCCx register.

If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCANnRFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFEMP Flag

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCANnRFCCx register is 0 or in global reset mode.

This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

NOTE

To clear the RFMLT or RFIF flag to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

14.3.7.3 RSCANnRFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

Access: RSCANnRFPCTRx register can only be written in 32-bit units.
RSCANnRFPCTRxL, RSCANnRFPCTRxH registers can only be written in 16-bit units.
RSCANnRFPCTRxLL, RSCANnRFPCTRxLH, RSCANnRFPCTRxHL, RSCANnRFPCTRxHH registers can only be written in 8-bit units.

Address: RSCANnRFPCTRx: <RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCANnRFPCTRxL: <RSCFDn_base> + $00F8_H$ + (04_H × x),
RSCANnRFPCTRxH: <RSCFDn_base> + $00FA_H$ + (04_H × x)
RSCANnRFPCTRxLL: <RSCFDn_base> + $00F8_H$ + (04_H × x),
RSCANnRFPCTRxLH: <RSCFDn_base> + $00F9_H$ + (04_H × x),
RSCANnRFPCTRxHL: <RSCFDn_base> + $00FA_H$ + (04_H × x),
RSCANnRFPCTRxHH: <RSCFDn_base> + $00FB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 14.42 RSCANnRFPCTRx Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	RFPC[7:0]	Receive FIFO Pointer Control When these bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer.

RFPC[7:0] Bits

When the RFPC[7:0] bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCANnRFSTSx register is decremented. Read the RSCANnRFIDx, RSCANnRFPTRx, RSCANnRFDF0_x, and RSCANnRFDF1_x registers to read messages in the receive FIFO buffer, and then write FF_H to the RFPC[7:0] bits.

When writing FF_H to these bits, make sure that the RFE bit in the RSCANnRFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCANnRFSTSx register is 0 (the receive FIFO buffer contains unread messages).

14.3.7.4 RSCANnRFIDx ─ Receive FIFO Buffer Access ID Register (x = 0 to 7)

Access: RSCANnRFIDx register can be read only in 32-bit units.
RSCANnRFIDxL, RSCANnRFIDxH registers can be read only in 16-bit units.
RSCANnRFIDxLL, RSCANnRFIDxLH, RSCANnRFIDxHL, RSCANnRFIDxHH registers can be read only in 8-bit units.

Address: RSCANnRFIDx: <RSCFDn_base> + $0E00_H$ + (10_H × x)
RSCANnRFIDxL: <RSCFDn_base> + $0E00_H$ + (10_H × x),
RSCANnRFIDxH: <RSCFDn_base> + $0E02_H$ + (10_H × x)
RSCANnRFIDxLL: <RSCFDn_base> + $0E00_H$ + (10_H × x),
RSCANnRFIDxLH: <RSCFDn_base> + $0E01_H$ + (10_H × x),
RSCANnRFIDxHL: <RSCFDn_base> + $0E02_H$ + (10_H × x),
RSCANnRFIDxHH: <RSCFDn_base> + $0E03_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.43 RSCANnRFIDx Register Contents

Bit Position	Bit Name	Function
31	RFIDE	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	Reserved	When read, the value after reset is returned.
28 to 0	RFID[28:0]	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

RFRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer.

RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

14.3.7.5 RSCANnRFPTRx ─ Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

Access: RSCANnRFPTRx register can be read only in 32-bit units.
RSCANnRFPTRxL, RSCANnRFPTRxH registers can be read only in 16-bit units.
RSCANnRFPTRxLL, RSCANnRFPTRxLH, RSCANnRFPTRxHL, RSCANnRFPTRxHH registers can be read only in 8-bit units.

Address: RSCANnRFPTRx: <RSCFDn_base> + $0E04_H$ + (10_H × x)
RSCANnRFPTRxL: <RSCFDn_base> + $0E04_H$ + (10_H × x),
RSCANnRFPTRxH: <RSCFDn_base> + $0E06_H$ + (10_H × x)
RSCANnRFPTRxLL: <RSCFDn_base> + $0E04_H$ + (10_H × x),
RSCANnRFPTRxLH: <RSCFDn_base> + $0E05_H$ + (10_H × x),
RSCANnRFPTRxHL: <RSCFDn_base> + $0E06_H$ + (10_H × x),
RSCANnRFPTRxHH: <RSCFDn_base> + $0E07_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.44 RSCANnRFPTRx Register Contents

Bit Position	Bit Name	Function
31 to 28	RFDLC[3:0]	Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RFPTR[11:0]	Receive FIFO Buffer Label Data Label information of the received message can be read.
15 to 0	RFTS[15:0]	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.

RFDLC[3:0] Bits

These bits contain the data length of the message stored in the receive FIFO buffer.

RFPTR[11:0] Bits

These bits contain the label information of the message stored in the receive FIFO buffer.

RFTS[15:0] Bits

These bits contain the timestamp value of the message stored in the receive FIFO buffer.

14.3.7.6 RSCANnRFDF0_x – Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7)

Access: RSCANnRFDF0_x register can be read only in 32-bit units.
RSCANnRFDF0_xL, RSCANnRFDF0_xH registers can be read only in 16-bit units.
RSCANnRFDF0_xLL, RSCANnRFDF0_xLH, RSCANnRFDF0_xHL, RSCANnRFDF0_xHH registers can be read only in 8-bit units.

Address: RSCANnRFDF0_x: <RSCFDn_base> + $0E08_H$ + (10_H × x)
RSCANnRFDF0_xL: <RSCFDn_base> + $0E08_H$ + (10_H × x),
RSCANnRFDF0_xH: <RSCFDn_base> + $0E0A_H$ + (10_H × x)
RSCANnRFDF0_xLL: <RSCFDn_base> + $0E08_H$ + (10_H × x),
RSCANnRFDF0_xLH: <RSCFDn_base> + $0E09_H$ + (10_H × x),
RSCANnRFDF0_xHL: <RSCFDn_base> + $0E0A_H$ + (10_H × x),
RSCANnRFDF0_xHH: <RSCFDn_base> + $0E0B_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB3[7:0]								RFDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB1[7:0]								RFDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.45 RSCANnRFDF0_x Register Contents

Bit Position	Bit Name	Function
31 to 24	RFDB3[7:0]	Receive FIFO Buffer Data Byte 3
23 to 16	RFDB2[7:0]	Receive FIFO Buffer Data Byte 2
15 to 8	RFDB1[7:0]	Receive FIFO Buffer Data Byte 1
7 to 0	RFDB0[7:0]	Receive FIFO Buffer Data Byte 0 Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCANnRFPTRx register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

14.3.7.7 RSCANnRFDF1_x – Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7)

Access: RSCANnRFDF1_x register can be read only in 32-bit units.
RSCANnRFDF1_xL, RSCANnRFDF1_xH registers can be read only in 16-bit units.
RSCANnRFDF1_xLL, RSCANnRFDF1_xLH, RSCANnRFDF1_xHL, RSCANnRFDF1_xHH registers can be read only in 8-bit units.

Address: RSCANnRFDF1_x: <RSCFDn_base> + $0E0C_H$ + (10_H × x)
RSCANnRFDF1_xL: <RSCFDn_base> + $0E0C_H$ + (10_H × x),
RSCANnRFDF1_xH: <RSCFDn_base> + $0E0E_H$ + (10_H × x)
RSCANnRFDF1_xLL: <RSCFDn_base> + $0E0C_H$ + (10_H × x),
RSCANnRFDF1_xLH: <RSCFDn_base> + $0E0D_H$ + (10_H × x),
RSCANnRFDF1_xHL: <RSCFDn_base> + $0E0E_H$ + (10_H × x),
RSCANnRFDF1_xHH: <RSCFDn_base> + $0E0F_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB7[7:0]								RFDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB5[7:0]								RFDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.46 RSCANnRFDF1_x Register Contents

Bit Position	Bit Name	Function
31 to 24	RFDB7[7:0]	Receive FIFO Buffer Data Byte 7
23 to 16	RFDB6[7:0]	Receive FIFO Buffer Data Byte 6
15 to 8	RFDB5[7:0]	Receive FIFO Buffer Data Byte 5
7 to 0	RFDB4[7:0]	Receive FIFO Buffer Data Byte 4 Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCANnRFPTRx register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

14.3.8 Details of Transmit/Receive FIFO Buffer-Related Registers

14.3.8.1 RSCANnCFCCk ─ Transmit/receive FIFO Buffer Configuration and Control Register (k = 0 to 11)

Access: RSCANnCFCCk register can be read/written in 32-bit units.
RSCANnCFCCkL, RSCANnCFCCkH registers can be read/written in 16-bit units.
RSCANnCFCCkLL, RSCANnCFCCkLH, RSCANnCFCCkHL, RSCANnCFCCkHH registers can be read/written in 8-bit units.

Address: RSCANnCFCCk: <RSCFDn_base> + 0118_H + (04_H × k)
RSCANnCFCCkL: <RSCFDn_base> + 0118_H + (04_H × k),
RSCANnCFCCkH: <RSCFDn_base> + $011A_H$ + (04_H × k)
RSCANnCFCCkLL: <RSCFDn_base> + 0118_H + (04_H × k),
RSCANnCFCCkLH: <RSCFDn_base> + 0119_H + (04_H × k),
RSCANnCFCCkHL: <RSCFDn_base> + $011A_H$ + (04_H × k),
RSCANnCFCCkHH: <RSCFDn_base> + $011B_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFTML[3:0]				CFITR	CFITSS	CFM[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	—	CFDC[2:0]			—	—	—	—	—	CFTXIE	CFRXIE	CFE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 14.47 RSCANnCFCCk Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 24	CFITT[7:0]	Set a message transmission interval. Set Value: 00_H to FF_H
23 to 20	CFTML[3:0]	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	Transmit/Receive FIFO Interval Timer Resolution 0: Clock dividing pclk by (ITRCP[15:0] bits) 1: Clock dividing pclk by (ITRCP[15:0] bits × 10)
18	CFITSS	Transmit/Receive FIFO Interval Timer Clock Source Select 0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the bit time clock for the channel to which the FIFO is linked.
17, 16	CFM[1:0]	Transmit/Receive FIFO Mode Select b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited

Table 14.47 RSCANnCFCCk Register Contents (2/2)

Bit Position	Bit Name	Function
15 to 13	CFIGCV[2:0]	Transmit/Receive FIFO Receive Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	CFIM	Transmit/Receive FIFO Interrupt Source Select 0: • Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. • Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. 1: • Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. • Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted.
11	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
10 to 8	CFDC[2:0]	Transmit/Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	CFTXIE	Transmit/Receive FIFO Transmit Interrupt Enable 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.
1	CFRXIE	Transmit/Receive FIFO Receive Interrupt Enable 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.
0	CFE	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

CFITT[7:0] Bits

These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode).

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

CFTML[3:0] Bits

These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode). There are three transmit/receive FIFO buffers per channel, so channel number m of FIFO buffer k is calculated as m = k/3 (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be ((16 × m) + CFTML[3:0]).

See **Table 14.12** and **Table 14.13** , as for the relationship between transmit/receive FIFO buffer k and transmit buffer p.

Setting the CFDC[2:0] bits to 001_B or more enables the setting of the CFTML[3:0] bits.

Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFITR Bit

This bit is enabled when the CFITSS bit is 0.

When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCANnGCFG register.

When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCANnGCFG register × 10).

Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFITSS Bit

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.

When this bit is 1, the bit time clock of the channel to which the FIFO is linked is the count source of the interval timer.

Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFM[1:0] Bits

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

CFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to 00_B (receive mode) or 10_B (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).

When the CFDC[2:0] bits are set to 001_B (4 messages), set the CFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B.

Modify these bits only in global reset mode.

CFIM Bit

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

CFDC[2:0] Bits

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to 000_B, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFTXIE Bit

When this bit is set to 1 and the CFTXIF flag in the RSCANnCFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFRXIE Bit

When this bit is set to 1 and the CFRXIF flag in the RSCANnCFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.

Modify this bit with the CFE bit set to 0.

CFE Bit

Setting this bit to 1 makes transmit/receive FIFO buffers available.

When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately.

This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode

Modify this bit in the following mode.

- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

After all other bits in the RSCANnCFCCk register have been set, set this bit to 1 by using another instruction.

14.3.8.2 RSCANnCFSTSk ─ Transmit/receive FIFO Buffer Status Register (k = 0 to 11)

Access: RSCANnCFSTSk register can be read/written in 32-bit units.
RSCANnCFSTSkL, RSCANnCFSTSkH registers can be read/written in 16-bit units.
RSCANnCFSTSkLL, RSCANnCFSTSkLH, RSCANnCFSTSkHL, RSCANnCFSTSkHH registers can be read/written in 8-bit units.

Address: RSCANnCFSTSk: <RSCFDn_base> + 0178_H + (04_H × k)
RSCANnCFSTSkL: <RSCFDn_base> + 0178_H + (04_H × k),
RSCANnCFSTSkH: <RSCFDn_base> + $017A_H$ + (04_H × k)
RSCANnCFSTSkLL: <RSCFDn_base> + 0178_H + (04_H × k),
RSCANnCFSTSkLH: <RSCFDn_base> + 0179_H + (04_H × k),
RSCANnCFSTSkHL: <RSCFDn_base> + $017A_H$ + (04_H × k),
RSCANnCFSTSkHH: <RSCFDn_base> + $017B_H$ + (04_H × k)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFT XIF	CFR XIF	CFMLT	CFFLL	CFEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.48 RSCANnCFSTSk Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
15 to 8	CFMC[7:0]	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer.
7 to 5	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
4	CFTXIF	Transmit/Receive FIFO Transmit Interrupt Request Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
3	CFRXIF	Transmit/Receive FIFO Receive Interrupt Request Flag 0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.
2	CFMLT	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFFLL	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCANnCFCCk register.

- When CFM[1:0] value is 01_B (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00_B (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10_B (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00_B: In global reset mode
- When CFM[1:0] value is 01_B or 10_B: In channel reset mode
- When the CFE bit in the RSCANnCFCCk register is cleared to 0.

CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01_B or 10_B, and the factor selected by the CFIM bit in the RSCANnCFCCk register occurs

The CFTXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFTXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B or 10_B, and the factor selected by the CFIM bit in the RSCANnCFCCk register occurs

The CFRXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFRXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFMLT flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFFLL Flag

The CFFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCANnCFCCk register.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits.
- When the CFE bit in the RSCANnCFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmit abort.
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: All messages have been transmitted, or in channel reset mode
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B or 10_B: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01_B: A value of FF_H has been written to the RSCANnCFPCTRk register after data was written to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers.

NOTE

To clear CFTXIF, CFRXIF, or CFMLT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

14.3.8.3 RSCANnCFPCTRk — Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 11)

Access: RSCANnCFPCTRk register can only be written in 32-bit units.
RSCANnCFPCTRkL, RSCANnCFPCTRkH registers can only be written in 16-bit units.
RSCANnCFPCTRkLL, RSCANnCFPCTRkLH, RSCANnCFPCTRkHL, RSCANnCFPCTRkHH registers can only be written in 8-bit units.

Address: RSCANnCFPCTRk: <RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCANnCFPCTRkL: <RSCFDn_base> + $01D8_H$ + (04_H × k),
RSCANnCFPCTRkH: <RSCFDn_base> + $01DA_H$ + (04_H × k)
RSCANnCFPCTRkLL: <RSCFDn_base> + $01D8_H$ + (04_H × k),
RSCANnCFPCTRkLH: <RSCFDn_base> + $01D9_H$ + (04_H × k),
RSCANnCFPCTRkHL: <RSCFDn_base> + $01DA_H$ + (04_H × k),
RSCANnCFPCTRkHH: <RSCFDn_base> + $01DB_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 14.49 RSCANnCFPCTRk Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	CFPC[7:0]	Transmit/Receive FIFO Pointer Control • Receive mode: Writing FF_H to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. • Transmit mode: Writing FF_H to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. • Gateway mode: Setting prohibited

CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCANnCFCCk register is 00_B):
  Writing FF_H to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCANnCFSTSk register is decremented. Read the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers to read messages from the transmit/receive FIFO buffer, and then write FF_H to the CFPC[7:0] bits.
  When writing FF_H to these bits, make sure that the CFE bit in the RSCANnCFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCANnCFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).
- Transmit mode (CFM[1:0] value in the RSCANnCFCCk register is 01_B):
  Writing FF_H to the CFPC[7:0] bits stores the data written to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCANnCFIDk, RSCANnCFPTRk, RSCANnCFDF0_k, and RSCANnCFDF1_k registers before writing FF_H to the CFPC[7:0] bits.
  When writing FF_H to these bits, make sure that the CFE bit in the RSCANnCFCCk register is set to 1 and the CFFLL flag in the RSCANnCFSTSk register is 0 (the transmit/receive FIFO buffer is not full).
- Gateway mode (CFM[1:0] value in the RSCANnCFCCk register is 10_B):
  Setting prohibited

14.3.8.4 RSCANnCFIDk — Transmit/Receive FIFO Buffer Access ID Register (k = 0 to 11)

Access: RSCANnCFIDk register can be read/written in 32-bit units.
RSCANnCFIDkL, RSCANnCFIDkH registers can be read/written in 16-bit units.
RSCANnCFIDkLL, RSCANnCFIDkLH, RSCANnCFIDkHL, RSCANnCFIDkHH registers can be read/written in 8-bit units.

Address: RSCANnCFIDk: <RSCFDn_base> + $0E80_H$ + (10_H × k)
RSCANnCFIDkL: <RSCFDn_base> + $0E80_H$ + (10_H × k),
RSCANnCFIDkH: <RSCFDn_base> + $0E82_H$ + (10_H × k)
RSCANnCFIDkLL: <RSCFDn_base> + $0E80_H$ + (10_H × k),
RSCANnCFIDkLH: <RSCFDn_base> + $0E81_H$ + (10_H × k),
RSCANnCFIDkHL: <RSCFDn_base> + $0E82_H$ + (10_H × k),
RSCANnCFIDkHH: <RSCFDn_base> + $0E83_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.50 RSCANnCFIDk Register Contents

Bit Position	Bit Name	Function
31	CFIDE	Transmit/Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	CFRTR	Transmit/Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable This bit is valid only when the CFM[1:0] value is 01_B (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	CFID[28:0]	Transmit/Receive FIFO Buffer ID Data • When CFM[1:0] value is 01_B (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. • When CFM[1:0] value is 00_B (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits 10 to 0. Bits 28 to 11 are read as 0.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, these bits are used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

CFRTR Bit

This bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, and buffer type) of transmit messages is stored in the transmit history buffer after transmission is completed.

This bit is enabled when the CFM[1:0] value is 01_B (transmit mode).

CFID[28:0] Bits

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B.

When the CFM[1:0] value is 01_B, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

14.3.8.5 RSCANnCFPTRk — Transmit/receive FIFO Buffer Access Pointer Register (k = 0 to 11)

Access: RSCANnCFPTRk register can be read/written in 32-bit units.
RSCANnCFPTRkL, RSCANnCFPTRkH registers can be read/written in 16-bit units.
RSCANnCFPTRkLL, RSCANnCFPTRkLH, RSCANnCFPTRkHL, RSCANnCFPTRkHH registers can be read/written in 8-bit units.

Address: RSCANnCFPTRk: <RSCFDn_base> + $0E84_H$ + (10_H × k)
RSCANnCFPTRkL: <RSCFDn_base> + $0E84_H$ + (10_H × k),
RSCANnCFPTRkH: <RSCFDn_base> + $0E86_H$ + (10_H × k)
RSCANnCFPTRkLL: <RSCFDn_base> + $0E84_H$ + (10_H × k),
RSCANnCFPTRkLH: <RSCFDn_base> + $0E85_H$ + (10_H × k),
RSCANnCFPTRkHL: <RSCFDn_base> + $0E86_H$ + (10_H × k),
RSCANnCFPTRkHH: <RSCFDn_base> + $0E87_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.51 RSCANnCFPTRk Register Contents

Bit Position	Bit Name	Function
31 to 28	CFDLC[3:0]	Transmit/Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	CFPTR[11:0]	Transmit/Receive FIFO Buffer Label Data • When CFM[1:0] value is 01_B (transmit mode): Set the label information to be stored in the transmit history buffer. Only bits CFPTR[7:0] are valid. • When CFM[1:0] value is 00_B (receive mode): The label information of the received message can be read.
15 to 0	CFTS[15:0]	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is 00_B (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFDLC[3:0] Bits

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer. If the data length is set to 1001_B or more, the actual transmit data defaults to 8 bytes.

CFPTR[11:0] Bits

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

CFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer.

These bits are valid when the CFM[1:0] value is 00_B.

14.3.8.6 RSCANnCFDF0_k — Transmit/receive FIFO Buffer Access Data Field 0 Register (k = 0 to 11)

Access: RSCANnCFDF0_k register can be read/written in 32-bit units.
RSCANnCFDF0_kL, RSCANnCFDF0_kH registers can be read/written in 16-bit units.
RSCANnCFDF0_kLL, RSCANnCFDF0_kLH, RSCANnCFDF0_kHL, RSCANnCFDF0_kHH registers can be read/written in 8-bit units.

Address: RSCANnCFDF0_k: <RSCFDn_base> + $0E88_H$ + (10_H × k)
RSCANnCFDF0_kL: <RSCFDn_base> + $0E88_H$ + (10_H × k),
RSCANnCFDF0_kH: <RSCFDn_base> + $0E8A_H$ + (10_H × k)
RSCANnCFDF0_kLL: <RSCFDn_base> + $0E88_H$ + (10_H × k),
RSCANnCFDF0_kLH: <RSCFDn_base> + $0E89_H$ + (10_H × k),
RSCANnCFDF0_kHL: <RSCFDn_base> + $0E8A_H$ + (10_H × k),
RSCANnCFDF0_kHH: <RSCFDn_base> + $0E8B_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB3[7:0]								CFDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB1[7:0]								CFDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.52 RSCANnCFDF0_k Register Contents

Bit Position	Bit Name	Function
31 to 24	CFDB3[7:0]	Transmit/Receive FIFO Buffer Data Byte 3
23 to 16	CFDB2[7:0]	Transmit/Receive FIFO Buffer Data Byte 2
15 to 8	CFDB1[7:0]	Transmit/Receive FIFO Buffer Data Byte 1
7 to 0	CFDB0[7:0]	Transmit/Receive FIFO Buffer Data Byte 0 • When CFM[1:0] value is 01_B (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00_B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01_B (transmit mode).

This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCANnCFPTRk register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

14.3.8.7 RSCANnCFDF1_k — Transmit/receive FIFO Buffer Access Data Field 1 Register (k = 0 to 11)

Access: RSCANnCFDF1_k register can be read/written in 32-bit units.
RSCANnCFDF1_kL, RSCANnCFDF1_kH registers can be read/written in 16-bit units.
RSCANnCFDF1_kLL, RSCANnCFDF1_kLH, RSCANnCFDF1_kHL, RSCANnCFDF1_kHH registers can be read/written in 8-bit units.

Address: RSCANnCFDF1_k: <RSCFDn_base> + $0E8C_H$ + (10_H × k),
RSCANnCFDF1_kL: <RSCFDn_base> + $0E8C_H$ + (10_H × k),
RSCANnCFDF1_kH: <RSCFDn_base> + $0E8E_H$ + (10_H × k),
RSCANnCFDF1_kLL: <RSCFDn_base> + $0E8C_H$ + (10_H × k),
RSCANnCFDF1_kLH: <RSCFDn_base> + $0E8D_H$ + (10_H × k),
RSCANnCFDF1_kHL: <RSCFDn_base> + $0E8E_H$ + (10_H × k),
RSCANnCFDF1_kHH: <RSCFDn_base> + $0E8F_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB7[7:0]								CFDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB5[7:0]								CFDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.53 RSCANnCFDF1_k Register Contents

Bit Position	Bit Name	Function
31 to 24	CFDB7[7:0]	Transmit/Receive FIFO Buffer Data Byte 7
23 to 16	CFDB6[7:0]	Transmit/Receive FIFO Buffer Data Byte 6
15 to 8	CFDB5[7:0]	Transmit/Receive FIFO Buffer Data Byte 5
7 to 0	CFDB4[7:0]	Transmit/Receive FIFO Buffer Data Byte 4 • When CFM[1:0] value is 01_B (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00_B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value in the RSCANnCFCCk register is 01_B (transmit mode).

This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCANnCFPTRk register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

14.3.9 Details of FIFO Status-Related Registers

14.3.9.1 RSCANnFESTS ─ FIFO Empty Status Register

Access: RSCANnFESTS register can be read only in 32-bit units.
RSCANnFESTSL, RSCANnFESTSH registers can be read only in 16-bit units.
RSCANnFESTSLL, RSCANnFESTSLH, RSCANnFESTSHL, RSCANnFESTSHH registers can be read only in 8-bit units.

Address: RSCANnFESTS: <RSCFDn_base> + 0238_H
RSCANnFESTSL: <RSCFDn_base> + 0238_H, RSCANnFESTSH: <RSCFDn_base> + $023A_H$
RSCANnFESTSLL: <RSCFDn_base> + 0238_H, RSCANnFESTSLH: <RSCFDn_base> + 0239_H,
RSCANnFESTSHL: <RSCFDn_base> + $023A_H$, RSCANnFESTSHH: <RSCFDn_base> + $023B_H$

Value after reset: $03FF\ FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11 EMP	CF10 EMP	CF9 EMP	CF8 EMP
Value after reset	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7 EMP	CF6 EMP	CF5 EMP	CF4 EMP	CF3 EMP	CF2 EMP	CF1 EMP	CF0 EMP	RF7 EMP	RF6 EMP	RF5 EMP	RF4 EMP	RF3 EMP	RF2 EMP	RF1 EMP	RF0 EMP
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.54 RSCANnFESTS Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after a reset is read.
19	CF11EMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 11)
18	CF10EMP	
17	CF9EMP	
16	CF8EMP	
15	CF7EMP	
14	CF6EMP	
13	CF5EMP	
12	CF4EMP	
11	CF3EMP	
10	CF2EMP	
9	CF1EMP	
8	CF0EMP	
7	RF7EMP	Receive FIFO Buffer Empty Status Flag 0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message (buffer empty). (x = 0 to 7)
6	RF6EMP	
5	RF5EMP	
4	RF4EMP	
3	RF3EMP	
2	RF2EMP	
1	RF1EMP	
0	RF0EMP	

The RSCANnFESTS register is set to 03FF FFFF$_H$ in global reset mode.

CFkEMP Flag (k = 0 to 11)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCANnCFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCANnRFSTSx register is set to 1 (the receive FIFO buffer contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

14.3.9.2 RSCANnFFSTS ─ FIFO Full Status Register

Access: RSCANnFFSTS register can be read only in 32-bit units.
RSCANnFFSTSL, RSCANnFFSTSH registers can be read only in 16-bit units.
RSCANnFFSTSLL, RSCANnFFSTSLH, RSCANnFFSTSHL, RSCANnFFSTSHH registers can be read only in 8-bit units.

Address: RSCANnFFSTS: <RSCFDn_base> + $023C_H$
RSCANnFFSTSL: <RSCFDn_base> + $023C_H$, RSCANnFFSTSH: <RSCFDn_base> + $023E_H$
RSCANnFFSTSLL: <RSCFDn_base> + $023C_H$, RSCANnFFSTSLH: <RSCFDn_base> + $023D_H$,
RSCANnFFSTSHL: <RSCFDn_base> + $023E_H$, RSCANnFFSTSHH: <RSCFDn_base> + $023F_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11FLL	CF10FLL	CF9FLL	CF8FLL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7FLL	CF6FLL	CF5FLL	CF4FLL	CF3FLL	CF2FLL	CF1FLL	CF0FLL	RF7FLL	RF6FLL	RF5FLL	RF4FLL	RF3FLL	RF2FLL	RF1FLL	RF0FLL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.55 RSCANnFFSTS Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after a reset is read.
19	CF11FLL	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full. (k = 0 to 11)
18	CF10FLL	
17	CF9FLL	
16	CF8FLL	
15	CF7FLL	
14	CF6FLL	
13	CF5FLL	
12	CF4FLL	
11	CF3FLL	
10	CF2FLL	
9	CF1FLL	
8	CF0FLL	
7	RF7FLL	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7)
6	RF6FLL	
5	RF5FLL	
4	RF4FLL	
3	RF3FLL	
2	RF2FLL	
1	RF1FLL	
0	RF0FLL	

The RSCANnFFSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkFLL Flag (k = 0 to 11)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCANnCFSTSk register is set to 1 (the transmit/receive FIFO buffer is full).

When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0.

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCANnRFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

14.3.9.3 RSCANnFMSTS – FIFO Message Lost Status Register

Access: RSCANnFMSTS register can be read only in 32-bit units.
RSCANnFMSTSL, RSCANnFMSTSH registers can be read only in 16-bit units.
RSCANnFMSTSLL, RSCANnFMSTSLH, RSCANnFMSTSHL, RSCANnFMSTSHH registers can be read only in 8-bit units.

Address: RSCANnFMSTS: <RSCFDn_base> + 0240_H
RSCANnFMSTSL: <RSCFDn_base> + 0240_H, RSCANnFMSTSH: <RSCFDn_base> + 0242_H
RSCANnFMSTSLL: <RSCFDn_base> + 0240_H, RSCANnFMSTSLH: <RSCFDn_base> + 0241_H,
RSCANnFMSTSHL: <RSCFDn_base> + 0242_H, RSCANnFMSTSHH: <RSCFDn_base> + 0243_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11MLT	CF10MLT	CF9MLT	CF8MLT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7MLT	CF6MLT	CF5MLT	CF4MLT	CF3MLT	CF2MLT	CF1MLT	CF0MLT	RF7MLT	RF6MLT	RF5MLT	RF4MLT	RF3MLT	RF2MLT	RF1MLT	RF0MLT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.56 RSCANnFMSTS Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after a reset is read.
19	CF11MLT	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 11)
18	CF10MLT	
17	CF9MLT	
16	CF8MLT	
15	CF7MLT	
14	CF6MLT	
13	CF5MLT	
12	CF4MLT	
11	CF3MLT	
10	CF2MLT	
9	CF1MLT	
8	CF0MLT	
7	RF7MLT	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	
5	RF5MLT	
4	RF4MLT	
3	RF3MLT	
2	RF2MLT	
1	RF1MLT	
0	RF0MLT	

The RSCANnFMSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkMLT Flag (k = 0 to 11)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO message is lost).

When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCANnRFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

14.3.9.4 RSCANnRFISTS ─ Receive FIFO Buffer Interrupt Flag Status Register

Access: RSCANnRFISTS register can be read only in 32-bit units.
RSCANnRFISTSL, RSCANnRFISTSH registers can be read only in 16-bit units.
RSCANnRFISTSLL, RSCANnRFISTSLH, RSCANnRFISTSHL, RSCANnRFISTSHH registers can be read only in 8-bit units.

Address: RSCANnRFISTS: <RSCFDn_base> + 0244_H
RSCANnRFISTSL: <RSCFDn_base> + 0244_H, RSCANnRFISTSH: <RSCFDn_base> + 0246_H
RSCANnRFISTSLL: <RSCFDn_base> + 0244_H, RSCANnRFISTSLH: <RSCFDn_base> + 0245_H,
RSCANnRFISTSHL: <RSCFDn_base> + 0246_H, RSCANnRFISTSHH: <RSCFDn_base> + 0247_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.57 RSCANnRFISTS Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When read, the value after reset is returned.
7	RF7IF	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	
5	RF5IF	
4	RF4IF	
3	RF3IF	
2	RF2IF	
1	RF1IF	
0	RF0IF	

The RSCANnRFISTS register is cleared to $0000\ 0000_H$ in global reset mode.

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCANnRFSTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

14.3.9.5 RSCANnCFRISTS ─ Transmit/Receive FIFO Buffer Receive Interrupt Flag Status Register

Access: RSCANnCFRISTS register can be read only in 32-bit units.
RSCANnCFRISTSL, RSCANnCFRISTSH registers can be read only in 16-bit units.
RSCANnCFRISTSLL, RSCANnCFRISTSLH, RSCANnCFRISTSHL, RSCANnCFRISTSHH registers can be read only in 8-bit units.

Address: RSCANnCFRISTS: <RSCFDn_base> + 0248_H
RSCANnCFRISTSL: <RSCFDn_base> + 0248_H, RSCANnCFRISTSH: <RSCFDn_base> + $024A_H$
RSCANnCFRISTSLL: <RSCFDn_base> + 0248_H, RSCANnCFRISTSLH: <RSCFDn_base> + 0249_H,
RSCANnCFRISTSHL: <RSCFDn_base> + $024A_H$, RSCANnCFRISTSHH: <RSCFDn_base> + $024B_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CF11 RXIF	CF10 RXIF	CF9 RXIF	CF8 RXIF	CF7 RXIF	CF6 RXIF	CF5 RXIF	CF4 RXIF	CF3 RXIF	CF2 RXIF	CF1 RXIF	CF0 RXIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.58 RSCANnCFRISTS Register Contents

Bit Position	Bit Name	Function
31 to 12	Reserved	When read, the value after reset is returned.
11	CF11RXIF	Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k receive interrupt request is present. 1: A transmit/receive FIFO buffer k receive interrupt request is present. (k = 0 to 11)
10	CF10RXIF	
9	CF9RXIF	
8	CF8RXIF	
7	CF7RXIF	
6	CF6RXIF	
5	CF5RXIF	
4	CF4RXIF	
3	CF3RXIF	
2	CF2RXIF	
1	CF1RXIF	
0	CF0RXIF	

The RSCANnCFRISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkRXIF Flag (k = 0 to 11)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0.

14.3.9.6 RSCANnCFTISTS ─ Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register

Access: RSCANnCFTISTS register can be read only in 32-bit units.
RSCANnCFTISTSL, RSCANnCFTISTSH registers can be read only in 16-bit units.
RSCANnCFTISTSLL, RSCANnCFTISTSLH, RSCANnCFTISTSHL, RSCANnCFTISTSHH registers can be read only in 8-bit units.

Address: RSCANnCFTISTS: <RSCFDn_base> + $024C_H$
RSCANnCFTISTSL: <RSCFDn_base> + $024C_H$, RSCANnCFTISTSH: <RSCFDn_base> + $024E_H$
RSCANnCFTISTSLL: <RSCFDn_base> + $024C_H$, RSCANnCFTISTSLH: <RSCFDn_base> + $024D_H$,
RSCANnCFTISTSHL: <RSCFDn_base> + $024E_H$, RSCANnCFTISTSHH: <RSCFDn_base> + $024F_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CF11 TXIF	CF10 TXIF	CF9 TXIF	CF8 TXIF	CF7 TXIF	CF6 TXIF	CF5 TXIF	CF4 TXIF	CF3 TXIF	CF2 TXIF	CF1 TXIF	CF0 TXIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.59 RSCANnCFTISTS Register Contents

Bit Position	Bit Name	Function
31 to 12	Reserved	When read, the value after a reset is read.
11	CF11TXIF	Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k transmit interrupt request is present. 1: A transmit/receive FIFO buffer k transmit interrupt request is present. (k = 0 to 11)
10	CF10TXIF	
9	CF9TXIF	
8	CF8TXIF	
7	CF7TXIF	
6	CF6TXIF	
5	CF5TXIF	
4	CF4TXIF	
3	CF3TXIF	
2	CF2TXIF	
1	CF1TXIF	
0	CF0TXIF	

The RSCANnCFTISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkTXIF Flag (k = 0 to 11)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCANnCFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

14.3.10 Details of Transmit Buffer-Related Registers

14.3.10.1 RSCANnTMCp ─ Transmit Buffer Control Register (p = 0 to 63)

Access: RSCANnTMCp registers can be read/written in 8-bit units.

Address: RSCANnTMCp: <RSCFDn_base> + 0250_H + (01_H × p)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W*1	R/W*1

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

Table 14.60 RSCANnTMCp Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
2	TMOM	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	Transmit Abort Request 0: Transmit abort is not requested. 1: Transmit abort is requested.
0	TMTR	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

When the RSCANnTMCp register meets any of the following conditions, set it to 00_H.

- The RSCANnTMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCANnCFCCk register (p = m × 16 + the value of CFTML[3:0] bits).
- The RSCANnTMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCANnTXQCCm (m = 0 to 3) register (p = (m × 16 + 15) to (m × 16 + 15 – the value of TXQDC[3:0] bits)).

Bits in the RSCANnTMCp register are all cleared to 0 in channel reset mode. Modify the RSCANnTMCp register in channel communication mode or channel halt mode.

TMOM Bit

Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed.

Modify the TMOM bit when the TMTRM flag in the RSCANnTMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

TMTAR Bit

Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted.

The TMTAR bit can be set to 1 when TMTR bit is 1.

The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed.
- An error or arbitration loss has been detected.

If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0.

TMTR Bit

Setting this bit to 1 transmits the message stored in the transmit buffer.

The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed after the TMTAR bit was set to 1.
- An error or arbitration-lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCANnTMSTSp register is 00_B.

14.3.10.2 RSCANnTMSTSp ─ Transmit Buffer Status Register (p = 0 to 63)

Access: RSCANnTMSTSp registers can be read/written in 8-bit units.

Address: RSCANnTMSTSp: <RSCFDn_base> + $02D0_H$ + (01_H × p)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R

Table 14.61 RSCANnTMSTSp Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
4	TMTARM	Transmit Buffer Transmit Abort Request Status Flag 0: No transmit abort request is present. 1: A transmit abort request is present.
3	TMTRM	Transmit Buffer Transmit Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2, 1	TMTRF[1:0]	Transmit Buffer Transmit Result Status Flag b2 b1 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmit abort has been completed. 1 0: Transmission has been completed (without transmit abort request). 1 1: Transmission has been completed (with transmit abort request).
0	TMTSTS	Transmit Buffer Transmit Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCANnTMSTSp register is cleared to all 0 in channel reset mode.

TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCANnTMCp register is set to 1.

The TMTARM flag is set to 0 when the TMTAR bit in the RSCANnTMCp register is set to 0.

TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCANnTMCp register is set to 1.

The TMTRM flag is set to 0 when the TMTR bit in the RSCANnTMCp register is set to 0.

TMTRF[1:0] Flag

This flag indicates the result of transmission from the transmit buffer.

00_B: Transmission is in progress or no transmit request is present.

01_B: Transmission from the transmit buffer was aborted.

10_B: Transmission has been completed with the TMTAR bit in the RSCANnTMCp register set to 0 (transmit abort is not requested).

11_B: Transmission has been completed with the TMTAR bit in the RSCANnTMCp register set to 1 (transmit abort is requested).

Write 00_B to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00_B to this flag.

TMTSTS Flag

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.

14.3.10.3 RSCANnTMIDp — Transmit Buffer ID Register (p = 0 to 63)

Access: RSCANnTMIDp register can be read/written in 32-bit units.
RSCANnTMIDpL, RSCANnTMIDpH registers can be read/written in 16-bit units.
RSCANnTMIDpLL, RSCANnTMIDpLH, RSCANnTMIDpHL, RSCANnTMIDpHH registers can be read/written in 8-bit units.

Address: RSCANnTMIDp: <RSCFDn_base> + 1000_H + (10_H × p)
RSCANnTMIDpL: <RSCFDn_base> + 1000_H + (10_H × p),
RSCANnTMIDpH: <RSCFDn_base> + 1002_H + (10_H × p)
RSCANnTMIDpLL: <RSCFDn_base> + 1000_H + (10_H × p),
RSCANnTMIDpLH: <RSCFDn_base> + 1001_H + (10_H × p),
RSCANnTMIDpHL: <RSCFDn_base> + 1002_H + (10_H × p),
RSCANnTMIDpHH: <RSCFDn_base> + 1003_H + (10_H × p)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.62 RSCANnTMIDp Register Contents

Bit Position	Bit Name	Function
31	TMIDE	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	Transmit Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID[28:0]	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp (timestamp is included if the TMTSCE bit in the RSCANnGCFG register is 1)) of transmit messages is stored in the transmit history buffer after transmission is completed.

TMID[28:0] Bits

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

14.3.10.4 RSCANnTMPTRp — Transmit Buffer Pointer Register (p = 0 to 63)

Access: RSCANnTMPTRp register can be read/written in 32-bit units.
RSCANnTMPTRpL, RSCANnTMPTRpH registers can be read/written in 16-bit units.
RSCANnTMPTRpLL, RSCANnTMPTRpLH, RSCANnTMPTRpHL, RSCANnTMPTRpHH registers can be read/written in 8-bit units.

Address: RSCANnTMPTRp: <RSCFDn_base> + 1004_H + (10_H × p)
RSCANnTMPTRpL: <RSCFDn_base> + 1004_H + (10_H × p),
RSCANnTMPTRpH: <RSCFDn_base> + 1006_H + (10_H × p)
RSCANnTMPTRpLL: <RSCFDn_base> + 1004_H + (10_H × p),
RSCANnTMPTRpLH: <RSCFDn_base> + 1005_H + (10_H × p),
RSCANnTMPTRpHL: <RSCFDn_base> + 1006_H + (10_H × p),
RSCANnTMPTRpHH: <RSCFDn_base> + 1007_H + (10_H × p)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.63 RSCANnTMPTRp Register Contents

Bit Position	Bit Name	Function
31 to 28	TMDLC[3:0]	Transmit Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 24	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
23 to 16	TMPTR[7:0]	Transmit Buffer Label Data Set the label information to be stored in the transmit history buffer.
15 to 0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCANnTMIDp register is set to 0 (data frame). If the data length is set to 1001_B or more, the transmit data is 8 bytes long.

When the TMRTR bit is set to 1 (remote frame), set the data length of messages to be requested.

TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

14.3.10.5 RSCANnTMDF0_p — Transmit Buffer Data Field 0 Register (p = 0 to 63)

Access: RSCANnTMDF0_p register can be read/written in 32-bit units.
RSCANnTMDF0_pL, RSCANnTMDF0_pH registers can be read/written in 16-bit units.
RSCANnTMDF0_pLL, RSCANnTMDF0_pLH, RSCANnTMDF0_pHL, RSCANnTMDF0_pHH registers can be read/written in 8-bit units.

Address: RSCANnTMDF0_p: <RSCFDn_base> + 1008_H + (10_H × p)
RSCANnTMDF0_pL: <RSCFDn_base> + 1008_H + (10_H × p),
RSCANnTMDF0_pH: <RSCFDn_base> + $100A_H$ + (10_H × p)
RSCANnTMDF0_pLL: <RSCFDn_base> + 1008_H + (10_H × p),
RSCANnTMDF0_pLH: <RSCFDn_base> + 1009_H + (10_H × p),
RSCANnTMDF0_pHL: <RSCFDn_base> + $100A_H$ + (10_H × p),
RSCANnTMDF0_pHH: <RSCFDn_base> + $100B_H$ + (10_H × p)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB3[7:0]								TMDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB1[7:0]								TMDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.64 RSCANnTMDF0_p Register Contents

Bit Position	Bit Name	Function
31 to 24	TMDB3[7:0]	Transmit Buffer Data Byte 3
23 to 16	TMDB2[7:0]	Transmit Buffer Data Byte 2
15 to 8	TMDB1[7:0]	Transmit Buffer Data Byte 1
7 to 0	TMDB0[7:0]	Transmit Buffer Data Byte 0 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

14.3.10.6 RSCANnTMDF1_p ─ Transmit Buffer Data Field 1 Register (p = 0 to 63)

Access: RSCANnTMDF1_p register can be read/written in 32-bit units.
RSCANnTMDF1_pL, RSCANnTMDF1_pH registers can be read/written in 16-bit units.
RSCANnTMDF1_pLL, RSCANnTMDF1_pLH, RSCANnTMDF1_pHL, RSCANnTMDF1_pHH registers can be read/written in 8-bit units.

Address: RSCANnTMDF1_p: <RSCFDn_base> + $100C_H$ + (10_H × p)
RSCANnTMDF1_pL: <RSCFDn_base> + $100C_H$ + (10_H × p),
RSCANnTMDF1_pH: <RSCFDn_base> + $100E_H$ + (10_H × p)
RSCANnTMDF1_pLL: <RSCFDn_base> + $100C_H$ + (10_H × p),
RSCANnTMDF1_pLH: <RSCFDn_base> + $100D_H$ + (10_H × p),
RSCANnTMDF1_pHL: <RSCFDn_base> + $100E_H$ + (10_H × p),
RSCANnTMDF1_pHH: <RSCFDn_base> + $100F_H$ + (10_H × p)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB7[7:0]								TMDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB5[7:0]								TMDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.65 RSCANnTMDF1_p Register Contents

Bit Position	Bit Name	Function
31 to 24	TMDB7[7:0]	Transmit Buffer Data Byte 7
23 to 16	TMDB6[7:0]	Transmit Buffer Data Byte 6
15 to 8	TMDB5[7:0]	Transmit Buffer Data Byte 5
7 to 0	TMDB4[7:0]	Transmit Buffer Data Byte 4 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCANnTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

14.3.10.7 RSCANnTMIECy ─ Transmit Buffer Interrupt Enable Configuration Register (y = 0, 1)

Access: RSCANnTMIECy register can be read/written in 32-bit units.
RSCANnTMIECyL, RSCANnTMIECyH registers can be read/written in 16-bit units.
RSCANnTMIECyLL, RSCANnTMIECyLH, RSCANnTMIECyHL, RSCANnTMIECyHH registers can be read/written in 8-bit units.

Address: RSCANnTMIECy: <RSCFDn_base> + 0390_H + (04_H × y)
RSCANnTMIECyL: <RSCFDn_base> + 0390_H + (04_H × y),
RSCANnTMIECyH: <RSCFDn_base> + 0392_H + (04_H × y)
RSCANnTMIECyLL: <RSCFDn_base> + 0390_H + (04_H × y),
RSCANnTMIECyLH: <RSCFDn_base> + 0391_H + (04_H × y),
RSCANnTMIECyHL: <RSCFDn_base> + 0392_H + (04_H × y),
RSCANnTMIECyHH: <RSCFDn_base> + 0393_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.66 RSCANnTMIECy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMIEp	Transmit Buffer Interrupt Enable p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled
15 to 0	TMIEp	Transmit Buffer Interrupt Enable p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

TMIEp Bits (p = 0 to 63)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCANnTMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue.

Table 14.67 shows the bit assignment.

Table 14.67 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.11 Details of Transmit Buffer Status-Related Registers

14.3.11.1 RSCANnTMTRSTSy ─ Transmit Buffer Transmit Request Status Register (y = 0, 1)

Access: RSCANnTMTRSTSy register can be read only in 32-bit units.
RSCANnTMTRSTSyL, RSCANnTMTRSTSyH registers can be read only in 16-bit units.
RSCANnTMTRSTSyLL, RSCANnTMTRSTSyLH, RSCANnTMTRSTSyHL, RSCANnTMTRSTSyHH registers can be read only in 8-bit units.

Address: RSCANnTMTRSTSy: <RSCFDn_base> + 0350_H + (04_H × y)
RSCANnTMTRSTSyL: <RSCFDn_base> + 0350_H + (04_H × y),
RSCANnTMTRSTSyH: <RSCFDn_base> + 0352_H + (04_H × y)
RSCANnTMTRSTSyLL: <RSCFDn_base> + 0350_H + (04_H × y),
RSCANnTMTRSTSyLH: <RSCFDn_base> + 0351_H + (04_H × y),
RSCANnTMTRSTSyHL: <RSCFDn_base> + 0352_H + (04_H × y),
RSCANnTMTRSTSyHH: <RSCFDn_base> + 0353_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.68 RSCANnTMTRSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit request is present. 1: A transmit request is present.
15 to 0	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

TMTRSTSp Flags (p = 0 to 63)

These flags indicate the status of the TMTR bit in the RSCANnTMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 14.69 shows the bit assignment.

Table 14.69 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.11.2 RSCANnTMTARSTSy ─ Transmit Buffer Transmit Abort Request Status Register (y = 0, 1)

Access: RSCANnTMTARSTSy register can be read only in 32-bit units.
RSCANnTMTARSTSyL, RSCANnTMTARSTSyH registers can be read only in 16-bit units.
RSCANnTMTARSTSyLL, RSCANnTMTARSTSyLH, RSCANnTMTARSTSyHL, RSCANnTMTARSTSyHH registers can be read only in 8-bit units.

Address: RSCANnTMTARSTSy: <RSCFDn_base> + 0360_H + (04_H × y)
RSCANnTMTARSTSyL: <RSCFDn_base> + 0360_H + (04_H × y),
RSCANnTMTARSTSyH: <RSCFDn_base> + 0362_H + (04_H × y)
RSCANnTMTARSTSyLL: <RSCFDn_base> + 0360_H + (04_H × y),
RSCANnTMTARSTSyLH: <RSCFDn_base> + 0361_H + (04_H × y),
RSCANnTMTARSTSyHL: <RSCFDn_base> + 0362_H + (04_H × y),
RSCANnTMTARSTSyHH: <RSCFDn_base> + 0363_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.70 RSCANnTMTARSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit abort request is present. 1: A transmit abort request is present.
15 to 0	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present.

TMTARSTSp Flags (p = 0 to 63)

These flags indicate the status of the TMTAR bit in the RSCANnTMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 14.71 shows the bit assignment.

Table 14.71 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.11.3 RSCANnTMTCSTSy ─ Transmit Buffer Transmit Complete Status Register (y = 0, 1)

Access: RSCANnTMTCSTSy register can be read only in 32-bit units.
RSCANnTMTCSTSyL, RSCANnTMTCSTSyH registers can be read only in 16-bit units.
RSCANnTMTCSTSyLL, RSCANnTMTCSTSyLH, RSCANnTMTCSTSyHL, RSCANnTMTCSTSyHH registers can be read only in 8-bit units.

Address: RSCANnTMTCSTSy: <RSCFDn_base> + 0370_H + (04_H × y)
RSCANnTMTCSTSyL: <RSCFDn_base> + 0370_H + (04_H × y),
RSCANnTMTCSTSyH: <RSCFDn_base> + 0372_H + (04_H × y)
RSCANnTMTCSTSyLL: <RSCFDn_base> + 0370_H + (04_H × y),
RSCANnTMTCSTSyLH: <RSCFDn_base> + 0371_H + (04_H × y),
RSCANnTMTCSTSyHL: <RSCFDn_base> + 0372_H + (04_H × y),
RSCANnTMTCSTSyHH: <RSCFDn_base> + 0373_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.72 RSCANnTMTCSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission has not been completed. 1: Transmission has been completed.
15 to 0	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

TMTCSTSp Flags (p = 0 to 63)

When the TMTRF[1:0] flag in the RSCANnTMSTSp register is set to 10_B (transmission has been completed (without transmit abort request)) or 11_B (transmission has been completed (with transmit abort request)), the corresponding TMTCSTSp flag is set to 1.

A TMTCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 14.73 shows the bit assignment.

Table 14.73 TMTCSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.11.4 RSCANnTMTASTSy ─ Transmit Buffer Transmit Abort Status Register (y = 0, 1)

Access: RSCANnTMTASTSy register can be read only in 32-bit units.
RSCANnTMTASTSyL, RSCANnTMTASTSyH registers can be read only in 16-bit units.
RSCANnTMTASTSyLL, RSCANnTMTASTSyLH, RSCANnTMTASTSyHL, RSCANnTMTASTSyHH registers can be read only in 8-bit units.

Address: RSCANnTMTASTSy: <RSCFDn_base> + 0380_H + (04_H × y)
RSCANnTMTASTSyL: <RSCFDn_base> + 0380_H + (04_H × y),
RSCANnTMTASTSyH: <RSCFDn_base> + 0382_H + (04_H × y)
RSCANnTMTASTSyLL: <RSCFDn_base> + 0380_H + (04_H × y),
RSCANnTMTASTSyLH: <RSCFDn_base> + 0381_H + (04_H × y),
RSCANnTMTASTSyHL: <RSCFDn_base> + 0382_H + (04_H × y),
RSCANnTMTASTSyHH: <RSCFDn_base> + 0383_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTASTSp (p = y × 32 + 31 to y x 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.74 RSCANnTMTASTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted. 1: Transmission is aborted.
15 to 0	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

TMTASTSp Flags (p = 0 to 63)

When the TMTRF[1:0] flag in the RSCANnTMSTSp register is set to 01_B (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 14.75 shows the bit assignment.

Table 14.75 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.12 Details of Transmit Queue-Related Registers

14.3.12.1 RSCANnTXQCCm — Transmit Queue Configuration and Control Register (m = 0 to 3)

Access: RSCANnTXQCCm register can be read/written in 32-bit units.
RSCANnTXQCCmL, RSCANnTXQCCmH registers can be read/written in 16-bit units.
RSCANnTXQCCmLL, RSCANnTXQCCmLH, RSCANnTXQCCmHL, RSCANnTXQCCmHH registers can be read/written in 8-bit units.

Address: RSCANnTXQCCm: <RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCANnTXQCCmL: <RSCFDn_base> + $03A0_H$ + (04_H × m),
RSCANnTXQCCmH: <RSCFDn_base> + $03A2_H$ + (04_H × m)
RSCANnTXQCCmLL: <RSCFDn_base> + $03A0_H$ + (04_H × m),
RSCANnTXQCCmLH: <RSCFDn_base> + $03A1_H$ + (04_H × m),
RSCANnTXQCCmHL: <RSCFDn_base> + $03A2_H$ + (04_H × m),
RSCANnTXQCCmHH: <RSCFDn_base> + $03A3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]				—	—	—	—	—	—	—	TXQE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 14.76 RSCANnTXQCCm Register Contents

Bit Position	Bit Name	Function
31 to 14	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
13	TXQIM	Transmit Queue Interrupt Source Select 0: When the transmit queue becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted.
12	TXQIE	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC[3:0]	Transmit Queue Depth Configuration Setting these bits to g (g = 2 to 15) makes the (g + 1)-buffer transmit queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited.
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	TXQE	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated.

Set the TXQE bit to 0 before modifying the TXQIE bit.

TXQDC[3:0] Bits

These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from (m × 16 + 15) to (m × 16 + 0). For examples of how buffer allocation is done, see **Figure 14.9**. Modify these bits only in channel reset mode.

TXQE Bit

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.

Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to 0010_B or more.

14.3.12.2 RSCANnTXQSTSm ─ Transmit Queue Configuration and Control Register (m = 0 to 3)

Access: RSCANnTXQSTSm register can be read/written in 32-bit units.
RSCANnTXQSTSmL, RSCANnTXQSTSmH registers can be read/written in 16-bit units.
RSCANnTXQSTSmLL, RSCANnTXQSTSmLH, RSCANnTXQSTSmHL, RSCANnTXQSTSmHH registers can be read/written in 8-bit units.

Address: RSCANnTXQSTSm: <RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCANnTXQSTSmL: <RSCFDn_base> + $03C0_H$ + (04_H × m),
RSCANnTXQSTSmH: <RSCFDn_base> + $03C2_H$ + (04_H × m)
RSCANnTXQSTSmLL: <RSCFDn_base> + $03C0_H$ + (04_H × m),
RSCANnTXQSTSmLH: <RSCFDn_base> + $03C1_H$ + (04_H × m),
RSCANnTXQSTSmHL: <RSCFDn_base> + $03C2_H$ + (04_H × m),
RSCANnTXQSTSmHH: <RSCFDn_base> + $03C3_H$ + (04_H × m)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQ FLL	TXQ EMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.77 RSCANnTXQSTSm Register Contents

Bit Position	Bit Name	Function
31 to 13	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
12 to 8	Reserved	When read, the undefined value is returned. When writing to these bits, write the value after reset.
7 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	TXQIF	Transmit Queue Interrupt Request Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TXQFLL	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

TXQIF Flag

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCANnTXQCCm register has occurred.

The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCANnTXQCCm register to 0 (the transmit queue is not used).

TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCANnTXQCCm register.

This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

TXQEMP Flag

The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.

This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

14.3.12.3 RSCANnTXQPCTRm — Transmit Queue Pointer Control Register (m = 0 to 3)

Access: RSCANnTXQPCTRm register can only be written in 32-bit units.
RSCANnTXQPCTRmL, RSCANnTXQPCTRmH registers can only be written in 16-bit units.
RSCANnTXQPCTRmLL, RSCANnTXQPCTRmLH, RSCANnTXQPCTRmHL, RSCANnTXQPCTRmHH registers can only be written in 8-bit units.

Address: RSCANnTXQPCTRm: <RSCFDn_base> + $03E0_H$ + (04_H × m)
RSCANnTXQPCTRmL: <RSCFDn_base> + $03E0_H$ + (04_H × m),
RSCANnTXQPCTRmH: <RSCFDn_base> + $03E2_H$ + (04_H × m)
RSCANnTXQPCTRmLL: <RSCFDn_base> + $03E0_H$ + (04_H × m),
RSCANnTXQPCTRmLH: <RSCFDn_base> + $03E1_H$ + (04_H × m),
RSCANnTXQPCTRmHL: <RSCFDn_base> + $03E2_H$ + (04_H × m),
RSCANnTXQPCTRmHH: <RSCFDn_base> + $03E3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 14.78 RSCANnTXQPCTRm Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	TXQPC[7:0]	Transmit Queue Pointer Control Writing FF_H to these bits moves the write pointer of the transmit queue to the next queue buffer.

TXQPC[7:0] Bits

Writing FF_H to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCANnTMIDp, RSCANnTMPTRp, RSCANnTMDF0_p, and RSCANnTMDF1_p registers (p = 15, 31, 47, 63) before writing FF_H to the TXQPC[7:0] bits.

When writing FF_H to these bits, make sure that the TXQE bit in the RSCANnTXQCCm register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCANnTXQSTSm register is 0 (he transmit queue is not full).

14.3.13 Details of Transmit History-Related Registers

14.3.13.1 RSCANnTHLCCm — Transmit History Configuration and Control Register (m = 0 to 3)

Access: RSCANnTHLCCm register can be read/written in 32-bit units.
RSCANnTHLCCmL, RSCANnTHLCCmH registers can be read/written in 16-bit units.
RSCANnTHLCCmLL, RSCANnTHLCCmLH, RSCANnTHLCCmHL, RSCANnTHLCCmHH registers can be read/written in 8-bit units.

Address: RSCANnTHLCCm: <RSCFDn_base> + 0400_H + (04_H × m)
RSCANnTHLCCmL: <RSCFDn_base> + 0400_H + (04_H × m),
RSCANnTHLCCmH: <RSCFDn_base> + 0402_H + (04_H × m)
RSCANnTHLCCmLL: <RSCFDn_base> + 0400_H + (04_H × m),
RSCANnTHLCCmLH: <RSCFDn_base> + 0401_H + (04_H × m),
RSCANnTHLCCmHL: <RSCFDn_base> + 0402_H + (04_H × m),
RSCANnTHLCCmHH: <RSCFDn_base> + 0403_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 14.79 RSCANnTHLCCm Register Contents

Bit Position	Bit Name	Function
31 to 11	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
10	THLDTE	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	THLE	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

THLIM Bit

This bit is used to select a transmit history interrupt source.

Modify this bit only in channel reset mode.

THLIE Bit

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only when the THLE bit set to 0.

THLE Bit

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer.

Modify this bit in channel communication mode or channel halt mode.

This bit is cleared to 0 in cahnnel reset mode.

14.3.13.2 RSCANnTHLSTSm ─ Transmit History Status Register (m = 0 to 3)

Access: RSCANnTHLSTSm register can be read/written in 32-bit units.
RSCANnTHLSTSmL, RSCANnTHLSTSmH registers can be read/written in 16-bit units.
RSCANnTHLSTSmLL, RSCANnTHLSTSmLH, RSCANnTHLSTSmHL, RSCANnTHLSTSmHH registers can be read/written in 8-bit units.

Address: RSCANnTHLSTSm: <RSCFDn_base> + 0420_H + (04_H × m)
RSCANnTHLSTSmL: <RSCFDn_base> + 0420_H + (04_H × m),
RSCANnTHLSTSmH: <RSCFDn_base> + 0422_H + (04_H × m)
RSCANnTHLSTSmLL: <RSCFDn_base> + 0420_H + (04_H × m),
RSCANnTHLSTSmLH: <RSCFDn_base> + 0421_H + (04_H × m),
RSCANnTHLSTSmHL: <RSCFDn_base> + 0422_H + (04_H × m),
RSCANnTHLSTSmHH: <RSCFDn_base> + 0423_H + (04_H × m)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]					—	—	—	—	THLIF	THL ELT	THL FLL	THL EMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.80 RSCANnTHLSTSm Register Contents

Bit Position	Bit Name	Function
31 to 13	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
12 to 8	THLMC[4:0]	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	THLIF	Transmit History Interrupt Request Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
2	THLELT	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	Transmit history Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

THLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer. These bits are cleared to 0 in channel reset mode.

THLIF Flag

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCANnTHLCCm register occurs.

This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLELT Flag

The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLFLL Flag

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCANnTHLCCm register is set to 0 (transmit history buffer is not used).

THLEMP Flag

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.

This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCANnTHLCCm register is set to 0 (transmit history buffer is not used).

NOTE

To clear THLIF or THLELT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

14.3.13.3 RSCANnTHLPCTRm — Transmit History Pointer Control Register (m = 0 to 3)

Access: RSCANnTHLPCTRm register can only be written in 32-bit units.
RSCANnTHLPCTRmL, RSCANnTHLPCTRmH registers can only be written in 16-bit units.
RSCANnTHLPCTRmLL, RSCANnTHLPCTRmLH, RSCANnTHLPCTRmHL, RSCANnTHLPCTRmHH registers can only be written in 8-bit units.

Address: RSCANnTHLPCTRm: <RSCFDn_base> + 0440_H + (04_H × m)
RSCANnTHLPCTRmL: <RSCFDn_base> + 0440_H + (04_H × m),
RSCANnTHLPCTRmH: <RSCFDn_base> + 0442_H + (04_H × m)
RSCANnTHLPCTRmLL: <RSCFDn_base> + 0440_H + (04_H × m),
RSCANnTHLPCTRmLH: <RSCFDn_base> + 0441_H + (04_H × m),
RSCANnTHLPCTRmHL: <RSCFDn_base> + 0442_H + (04_H × m),
RSCANnTHLPCTRmHH: <RSCFDn_base> + 0443_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 14.81 RSCANnTHLPCTRm Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When writing to these bits, write the value after reset.
7 to 0	THLPC[7:0]	Transmit History List Pointer Control Writing FF_H to these bits moves the read pointer to the next unread data in the transmit history buffer.

THLPC[7:0] Bits

When the THLPC[7:0] bits are set to FF_H, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCANnTHLSTSm register is decremented. Write FF_H to the THLPC[7:0] bits after reading from the RSCANnTHLACCm register.

When writing FF_H to these bits, make sure that the THLE bit in the RSCANnTHLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCANnTHLSTSm register is 0.

14.3.13.4 RSCANnTHLACCm — Transmit History Access Register (m = 0 to 3)

Access: RSCANnTHLACCm register can be read only in 32-bit units.
RSCANnTHLACCmL, RSCANnTHLACCmH registers can be read only in 16-bit units.
RSCANnTHLACCmLL, RSCANnTHLACCmLH, RSCANnTHLACCmHL, RSCANnTHLACCmHH registers can be read only in 8-bit units.

Address: RSCANnTHLACCm: <RSCFDn_base> + 1800_H + (04_H × m)
RSCANnTHLACCmL: <RSCFDn_base> + 1800_H + (04_H × m),
RSCANnTHLACCmH: <RSCFDn_base> + 1802_H + (04_H × m)
RSCANnTHLACCmLL: <RSCFDn_base> + 1800_H + (04_H × m),
RSCANnTHLACCmLH: <RSCFDn_base> + 1801_H + (04_H × m),
RSCANnTHLACCmHL: <RSCFDn_base> + 1802_H + (04_H × m),
RSCANnTHLACCmHH: <RSCFDn_base> + 1803_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]								—	BN[3:0]				BT[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.82 RSCANnTHLACCm Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTS[15:0]	Timestamp Data The timestamp data of stored data can be read.
15 to 8	TID[7:0]	Label Data The label information of stored data can be read.
7	Reserved	When read, the value after reset is returned.
6 to 3	BN[3:0]	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.
2 to 0	BT[2:0]	Buffer Type Data b2 b1 b0 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 1 0 0: Transmit queue

TMTS[15:0] Bits

When the TMTSCE bit in the RSCANnGCFG register is 1, timestamp values in transmit history data stored in the transmit history buffer are displayed. When the TMTSCE bit is 0, these bits are always read as 0.

TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

BN[3:0] Bits

These bits indicate the transmit source buffer number in the transmit history data stored in the transmit history buffer.

BT[2:0] Bits

These bits indicate the type of the transmit source buffer in the transmit history data stored in the transmit history buffer.

14.3.14 Details of Test-Related Registers

14.3.14.1 RSCANnGTSTCFG ─ Global Test Configuration Register

Access: RSCANnGTSTCFG register can be read/written in 32-bit units.
RSCANnGTSTCFGL, RSCANnGTSTCFGH registers can be read/written in 16-bit units.
RSCANnGTSTCFGLL, RSCANnGTSTCFGLH, RSCANnGTSTCFGHL, RSCANnGTSTCFGHH registers can be read/written in 8-bit units.

Address: RSCANnGTSTCFG: <RSCFDn_base> + 0468_H
RSCANnGTSTCFGL: <RSCFDn_base> + 0468_H, RSCANnGTSTCFGH: <RSCFDn_base> + $046A_H$
RSCANnGTSTCFGLL: <RSCFDn_base> + 0468_H, RSCANnGTSTCFGLH: <RSCFDn_base> + 0469_H,
RSCANnGTSTCFGHL: <RSCFDn_base> + $046A_H$, RSCANnGTSTCFGHH: <RSCFDn_base> + $046B_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	C3ICBCE	C2ICBCE	C1ICBCE	C0ICBCE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 14.83 RSCANnGTSTCFG Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
22 to 16	RTMPS[6:0]	RAM Test Page Configuration Set a value within a range of page 0 (00_H) to page 39 (27_H).
15 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	C3ICBCE	CAN3 Inter-channel Communication Test Enable 0: CAN3 inter-channel communication test is disabled. 1: CAN3 inter-channel communication test is enabled.
2	C2ICBCE	CAN2 Inter-channel Communication Test Enable 0: CAN2 inter-channel communication test is disabled. 1: CAN2 inter-channel communication test is enabled.
1	C1ICBCE	CAN1 Inter-channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	CAN0 Inter-channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCANnGTSTCFG register only in global test mode.

RTMPS[6:0] Bits

These bits are used to set the RAM test target page number for RAM test. Set a value in the range of 00_H to 27_H, inclusive. In addition, do not access more than 128 bytes in the last page (RTMPS[6:0] = 27_H) of the RAM.

C3ICBCE Bit

Setting this bit to 1 enables the channel 3 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

C2ICBCE Bit

Setting this bit to 1 enables the channel 2 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

C1ICBCE Bit

Setting this bit to 1 enables the channel 1 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

C0ICBCE Bit

Setting this bit to 1 enables the channel 0 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

14.3.14.2 RSCANnGTSTCTR ─ Global Test Control Register

Access: RSCANnGTSTCTR register can be read/written in 32-bit units.
RSCANnGTSTCTRL, RSCANnGTSTCTRH registers can be read/written in 16-bit units.
RSCANnGTSTCTRLL, RSCANnGTSTCTRLH, RSCANnGTSTCTRHL, RSCANnGTSTCTRHH registers can be read/written in 8-bit units.

Address: RSCANnGTSTCTR: <RSCFDn_base> + $046C_H$
RSCANnGTSTCTRL: <RSCFDn_base> + $046C_H$, RSCANnGTSTCTRH: <RSCFDn_base> + $046E_H$
RSCANnGTSTCTRLL: <RSCFDn_base> + $046C_H$, RSCANnGTSTCTRLH: <RSCFDn_base> + $046D_H$,
RSCANnGTSTCTRHL: <RSCFDn_base> + $046E_H$, RSCANnGTSTCTRHH: <RSCFDn_base> + $046F_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	ICBCTME
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Table 14.84 RSCANnGTSTCTR Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	RTME	RAM Test Enable 0: RAM test is disabled. 1: RAM test is enabled.
1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	ICBCTME	Communication Test between Channels Enable 0: Communication test between channels disabled. 1: Communication test between channels enabled

RTME Bit

Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode. This bit is cleared to 0 in global reset mode.

1. Set the GMDC[1:0] bits in the RSCANnGCTR register to 10_B (Global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0 to 3) in the RSCANnGTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode. This bit is cleared to 0 in global reset mode.

14.3.14.3 RSCANnGLOCKK ─ Global Lock Key Register

Access: RSCANnGLOCKK register can only be written in 32-bit units.
RSCANnGLOCKKL, RSCANnGLOCKKH registers can only be written in 16-bit units.

Address: RSCANnGLOCKK: <RSCFDn_base> + $047C_H$
RSCANnGLOCKKL: <RSCFDn_base> + $047C_H$, RSCANnGLOCKKH: <RSCFDn_base> + $047E_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1

Note 1. Writing to these bits is effective only when the RS-CANFD module is in global test mode.

Table 14.85 RSCANnGLOCKK Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When writing these bits, write the value after reset.
15 to 0	LOCK[15:0]	Lock Key These bits are key bits to release protection of test mode.

The RSCANnGLOCKK register releases protection of special test bits and is write only.

For the protection release data, see **Section 14.11.4.2, Procedure for Releasing the Protection**.

LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCANnGTSTCTR register.

After the protection has been released, writing to the I/O register area (<RSCFDn_base> + 0000_H to <RSCFDn_base> + $04FF_H$) of the CAN (except the RAM) enables the protection again.

Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.

14.3.14.4 RSCANnRPGACCr — RAM Test Page Access Register (r = 0 to 63)

Access: RSCANnRPGACCr register can be read/written in 32-bit units.
RSCANnRPGACCrL, RSCANnRPGACCrH registers can be read/written in 16-bit units.
RSCANnRPGACCrLL, RSCANnRPGACCrLH, RSCANnRPGACCrHL, RSCANnRPGACCrHH registers can be read/written in 8-bit units.

Address: RSCANnRPGACCr: <RSCFDn_base> + 1900_H + (04_H × r)
RSCANnRPGACCrL: <RSCFDn_base> + 1900_H + (04_H × r),
RSCANnRPGACCrH: <RSCFDn_base> + 1902_H + (04_H × r)
RSCANnRPGACCrLL: <RSCFDn_base> + 1900_H + (04_H × r),
RSCANnRPGACCrLH: <RSCFDn_base> + 1901_H + (04_H × r),
RSCANnRPGACCrHL: <RSCFDn_base> + 1902_H + (04_H × r),
RSCANnRPGACCrHH: <RSCFDn_base> + 1903_H + (04_H × r)

Value after reset: $0000\ 0000_H$

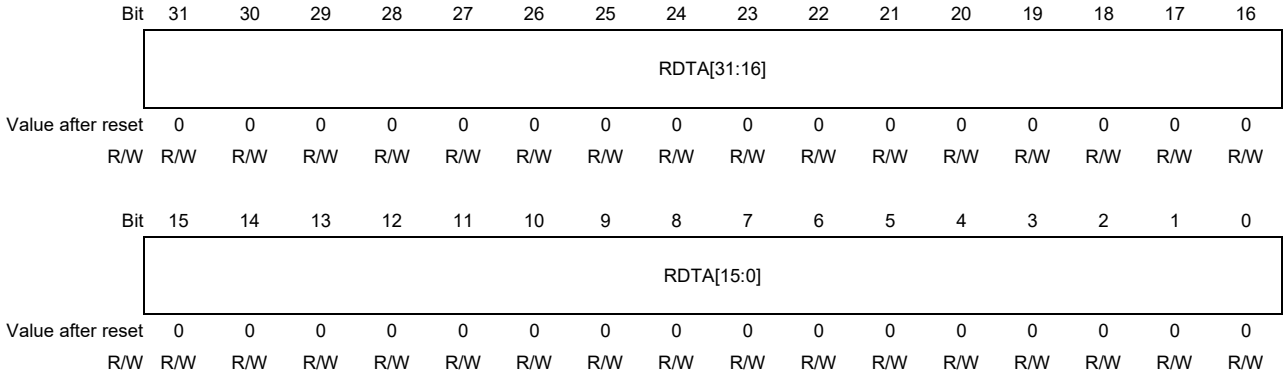


Table 14.86 RSCANnRPGACCr Register Contents

Bit Position	Bit Name	Function
31 to 0	RDTA[31:0]	RAM Data Test Access Data can be read and written in CAN RAM.

Modify the RSCANnRPGACCr register in global test mode with the RTME bit in the RSCANnGTSTCTR register set to 1 (RAM test is enabled).

The RSCANnRPGACCr register is readable and writable when the RTME bit is set to 1.

14.4 Registers (CAN FD Mode)

This section describes all registers to be used when the RS-CANFD is used in CAN FD mode.

14.4.1 List of Registers

The following tables list RS-CANFD registers to be used in CAN FD mode.

For details about <RSCFDn_base>, see **Section 14.1.2, Register Base Address**.

Table 14.87 Registers (1/4)

Module	Register	Symbol	Address
Interface mode-related registers			
RSCFDn	Global interface mode select register	RSCFDnCFDGRMCFG	<RSCFDn_base> + $04FC_H$
Channel-related registers			
RSCFDn	Channel m nominal bit rate configuration register	RSCFDnCFDCmNCFG	<RSCFDn_base> + 0000_H + (10_H × m)
RSCFDn	Channel m control register	RSCFDnCFDCmCTR	<RSCFDn_base> + 0004_H + (10_H × m)
RSCFDn	Channel m status register	RSCFDnCFDCmSTS	<RSCFDn_base> + 0008_H + (10_H × m)
RSCFDn	Channel m error flag register	RSCFDnCFDCmERFL	<RSCFDn_base> + $000C_H$ + (10_H × m)
RSCFDn	Channel m data bit rate configuration register	RSCFDnCFDCmDCFG	<RSCFDn_base> + 0500_H + (20_H × m)
RSCFDn	Channel m CAN FD configuration register	RSCFDnCFDCmFDCFG	<RSCFDn_base> + 0504_H + (20_H × m)
RSCFDn	Channel m CAN FD control register	RSCFDnCFDCmFDCTR	<RSCFDn_base> + 0508_H + (20_H × m)
RSCFDn	Channel m CAN FD status register	RSCFDnCFDCmFDSTS	<RSCFDn_base> + $050C_H$ + (20_H × m)
RSCFDn	Channel m CAN FD CRC register	RSCFDnCFDCmFDCRC	<RSCFDn_base> + 0510_H + (20_H × m)
Global-related registers			
RSCFDn	Global configuration register	RSCFDnCFDGCFG	<RSCFDn_base> + 0084_H
RSCFDn	Global control register	RSCFDnCFDGCTR	<RSCFDn_base> + 0088_H
RSCFDn	Global status register	RSCFDnCFDGSTS	<RSCFDn_base> + $008C_H$
RSCFDn	Global error flag register	RSCFDnCFDGERFL	<RSCFDn_base> + 0090_H
RSCFDn	Global timestamp counter register	RSCFDnCFDGTSC	<RSCFDn_base> + 0094_H
RSCFDn	Global TX Interrupt Status Register 0	RSCFDnCFDGTINTSTS0	<RSCFDn_base> + 0460_H
RSCFDn	Glboal FD configuration register	RSCFDnCFDGFDCFG	<RSCFDn_base> + 0474_H
RSCFDn	Global CRC configuration register	RSCFDnCFDGCRCCFG	<RSCFDn_base> + 0478_H
Receive rule-related registers			
RSCFDn	Receive Rule Entry Control Register	RSCFDnCFDGAFLECTR	<RSCFDn_base> + 0098_H
RSCFDn	Receive Rule Configuration Register 0	RSCFDnCFDGAFLCFG0	<RSCFDn_base> + $009C_H$
RSCFDn	Receive Rule ID Register j	RSCFDnCFDGAFLIDj	<RSCFDn_base> + 1000_H + (10_H × j)
RSCFDn	Receive Rule Mask Register j	RSCFDnCFDGAFLMj	<RSCFDn_base> + 1004_H + (10_H × j)

Table 14.87 Registers (2/4)

Module	Register	Symbol	Address
RSCFDn	Receive Rule Pointer 0 Register j	RSCFDnCFDGAFLP0_j	<RSCFDn_base> + 1008_H + (10_H × j)
RSCFDn	Receive Rule Pointer 1 Register j	RSCFDnCFDGAFLP1_j	<RSCFDn_base> + $100C_H$ + (10_H × j)
Receive buffer-related registers			
RSCFDn	Receive Buffer Number Register	RSCFDnCFDRMNB	<RSCFDn_base> + $00A4_H$
RSCFDn	Receive Buffer New Data Register y	RSCFDnCFDRMNDy	<RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCFDn	Receive Buffer ID Register q	RSCFDnCFDRMIDq	<RSCFDn_base> + 2000_H + (20_H × q)
RSCFDn	Receive Buffer Pointer Register q	RSCFDnCFDRMPTRq	<RSCFDn_base> + 2004_H + (20_H × q)
RSCFDn	Receive buffer CAN FD status register q	RSCFDnCFDRMFDSTSq	<RSCFDn_base> + 2008_H + (20_H × q)
RSCFDn	Receive Buffer Data Field b Register q	RSCFDnCFDRMDFb_q	<RSCFDn_base> + $200C_H$ + (04_H × b) + (20_H × q)
Receive FIFO buffer-related registers			
RSCFDn	Receive FIFO Buffer Configuration and Control Register x	RSCFDnCFDRFCCx	<RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCFDn	Receive FIFO Buffer Status Register x	RSCFDnCFDRFSTSx	<RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCFDn	Receive FIFO Buffer Pointer Control Register x	RSCFDnCFDRFPCTRx	<RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCFDn	Receive FIFO Buffer Access ID Register x	RSCFDnCFDRFIDx	<RSCFDn_base> + 3000_H + (80_H × x)
RSCFDn	Receive FIFO Buffer Access Pointer Register x	RSCFDnCFDRFPTRx	<RSCFDn_base> + 3004_H + (80_H × x)
RSCFDn	Receive FIFO CAN FD status register x	RSCFDnCFDRFFDSTSx	<RSCFDn_base> + 3008_H + (80_H × x)
RSCFDn	Receive FIFO Buffer Access Data Field d Register x	RSCFDnCFDRFDFd_x	<RSCFDn_base> + $300C_H$ + (04_H × d) + (80_H × x)
Transmit/Receive FIFO buffer related registers			
RSCFDn	Transmit/receive FIFO Buffer Configuration and Control Register k	RSCFDnCFDCFCCk	<RSCFDn_base> + 0118_H + (04_H × k)
RSCFDn	Transmit/receive FIFO Buffer Status Register k	RSCFDnCFDCFSTSk	<RSCFDn_base> + 0178_H + (04_H × k)
RSCFDn	Transmit/receive FIFO Buffer Pointer Control Register k	RSCFDnCFDCFPCTRk	<RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCFDn	Transmit/receive FIFO Buffer Access ID Register k	RSCFDnCFDCFIDk	<RSCFDn_base> + 3400_H + (80_H × k)
RSCFDn	Transmit/receive FIFO Buffer Access Pointer Register k	RSCFDnCFDCFPTRk	<RSCFDn_base> + 3404_H + (80_H × k)
RSCFDn	Transmit/receive FIFO CAN FD configuration/status register k	RSCFDnCFDCFFDCSTSk	<RSCFDn_base> + 3408_H + (80_H × k)
RSCFDn	Transmit/receive FIFO Buffer Access Data Field d Register k	RSCFDnCFDCFDFd_k	<RSCFDn_base> + $340C_H$ + (04_H × d) + (80_H × k)

Table 14.87 Registers (3/4)

Module	Register	Symbol	Address
FIFO status-related registers			
RSCFDn	FIFO Empty Status Register	RSCFDnCFDFESTS	<RSCFDn_base> + 0238_H
RSCFDn	FIFO Full Status Register	RSCFDnCFDFFSTS	<RSCFDn_base> + $023C_H$
RSCFDn	FIFO Message Lost Status Register	RSCFDnCFDFMSTS	<RSCFDn_base> + 0240_H
RSCFDn	Receive FIFO Buffer Interrupt Flag Status Register	RSCFDnCFDRFISTS	<RSCFDn_base> + 0244_H
RSCFDn	Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register	RSCFDnCFDCFRISTS	<RSCFDn_base> + 0248_H
RSCFDn	Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register	RSCFDnCFDCFTISTS	<RSCFDn_base> + $024C_H$
FIFO DMA-related registers			
RSCFDn	DMA enable register	RSCFDnCFDCDTCT	<RSCFDn_base> + 0490_H
RSCFDn	DMA status register	RSCFDnCFDCDTSTS	<RSCFDn_base> + 0494_H
Transmit buffer-related registers			
RSCFDn	Transmit Buffer Control Register p	RSCFDnCFDTMCp	<RSCFDn_base> + 0250_H + (01_H × p)
RSCFDn	Transmit Buffer Status Register p	RSCFDnCFDTMSTSp	<RSCFDn_base> + $02D0_H$ + (01_H × p)
RSCFDn	Transmit Buffer ID Register p	RSCFDnCFDTMIDp	<RSCFDn_base> + 4000_H + (20_H × p)
RSCFDn	Transmit Buffer Pointer Register p	RSCFDnCFDTMPTRp	<RSCFDn_base> + 4004_H + (20_H × p)
RSCFDn	Transmit buffer CAN FD configuration register p	RSCFDnCFDTMFDCTRp	<RSCFDn_base> + 4008_H + (20_H × p)
RSCFDn	Transmit Buffer Data Field b Register p	RSCFDnCFDTMDFb_p	<RSCFDn_base> + $400C_H$ + (04_H × b) + (20_H × p)
RSCFDn	Transmit Buffer Interrupt Enable Configuration Register y	RSCFDnCFDTMIECy	<RSCFDn_base> + 0390_H + (04_H × y)
Transmit buffer status-related registers			
RSCFDn	Transmit Buffer Transmit Request Status Register y	RSCFDnCFDTMTRSTSy	<RSCFDn_base> + 0350_H + (04_H × y)
RSCFDn	Transmit Buffer Transmit Abort Request Status Register y	RSCFDnCFDTMTARSTSy	<RSCFDn_base> + 0360_H + (04_H × y)
RSCFDn	Transmit Buffer Transmit Complete Status Register y	RSCFDnCFDTMTCSTSy	<RSCFDn_base> + 0370_H + (04_H × y)
RSCFDn	Transmit Buffer Transmit Abort Status Register y	RSCFDnCFDTMTASTSy	<RSCFDn_base> + 0380_H + (04_H × y)
Transmit queue-related registers			
RSCFDn	Transmit Queue Configuration and Control Register m	RSCFDnCFDTXQCCm	<RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCFDn	Transmit Queue Status Register m	RSCFDnCFDTXQSTSm	<RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCFDn	Transmit Queue Pointer Control Register m	RSCFDnCFDTXQPCTRm	<RSCFDn_base> + $03E0_H$ + (04_H × m)

Table 14.87 Registers (4/4)

Module	Register	Symbol	Address
Transmit history-related registers			
RSCFDn	Transmit History Configuration and Control Register m	RSCFDnCFDTHLCCm	<RSCFDn_base> + 0400_H + (04_H × m)
RSCFDn	Transmit History Status Register m	RSCFDnCFDTHLSTSm	<RSCFDn_base> + 0420_H + (04_H × m)
RSCFDn	Transmit History Pointer Control Register m	RSCFDnCFDTHLPCTRm	<RSCFDn_base> + 0440_H + (04_H × m)
RSCFDn	Transmit History Access Register m	RSCFDnCFDTHLACCm	<RSCFDn_base> + 6000_H + (04_H × m)
Test-related registers			
RSCFDn	Global Test Configuration Register	RSCFDnCFDGTSTCFG	<RSCFDn_base> + 0468_H
RSCFDn	Global Test Control Register	RSCFDnCFDGTSTCTR	<RSCFDn_base> + $046C_H$
RSCFDn	Global Lock Key Register	RSCFDnCFDGLOCKK	<RSCFDn_base> + $047C_H$
RSCFDn	RAM Test Page Access Register r	RSCFDnCFDRPGACCr	<RSCFDn_base> + 6400_H + (04_H × r)

Table 14.88 Transmit Buffer p Allocated to Each Channel

	CANm
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
	Transmit buffer 16 × m + 15

Table 14.89 Transmit/Receive FIFO Buffer k Allocated to Each Channel

	CANm
Transmit/receive FIFO buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

Table 14.90 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
0000_B	Transmit buffer 16 × m + 0
0001_B	Transmit buffer 16 × m + 1
0010_B	Transmit buffer 16 × m + 2
0011_B	Transmit buffer 16 × m + 3
0100_B	Transmit buffer 16 × m + 4
0101_B	Transmit buffer 16 × m + 5
0110_B	Transmit buffer 16 × m + 6
0111_B	Transmit buffer 16 × m + 7
1000_B	Transmit buffer 16 × m + 8
1001_B	Transmit buffer 16 × m + 9
1010_B	Transmit buffer 16 × m + 10
1011_B	Transmit buffer 16 × m + 11
1100_B	Transmit buffer 16 × m + 12
1101_B	Transmit buffer 16 × m + 13
1110_B	Transmit buffer 16 × m + 14
1111_B	Transmit buffer 16 × m + 15

Table 14.91 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

Setting of Bits TXQDC [3:0]	Transmit Buffer p Allocated to the Transmit Queue
0000_B	Setting prohibited
0001_B	Setting prohibited
0010_B	Transmit buffer 16 × m + 15 to 16 × m + 13
0011_B	Transmit buffer 16 × m + 15 to 16 × m + 12
0100_B	Transmit buffer 16 × m + 15 to 16 × m + 11
0101_B	Transmit buffer 16 × m + 15 to 16 × m + 10
0110_B	Transmit buffer 16 × m + 15 to 16 × m + 9
0111_B	Transmit buffer 16 × m + 15 to 16 × m + 8
1000_B	Transmit buffer 16 × m + 15 to 16 × m + 7
1001_B	Transmit buffer 16 × m + 15 to 16 × m + 6
1010_B	Transmit buffer 16 × m + 15 to 16 × m + 5
1011_B	Transmit buffer 16 × m + 15 to 16 × m + 4
1100_B	Transmit buffer 16 × m + 15 to 16 × m + 3
1101_B	Transmit buffer 16 × m + 15 to 16 × m + 2
1110_B	Transmit buffer 16 × m + 15 to 16 × m + 1
1111_B	Transmit buffer 16 × m + 15 to 16 × m + 0

14.4.2 Details of Interface Mode-Related Registers

14.4.2.1 RSCFDnCFDGRMCFG ─ Global Interface Mode Select Register

Access: RSCFDnCFDGRMCFG register can be read/written in 32-bit units.
RSCFDnCFDGRMCFGL, RSCFDnCFDGRMCFGH registers can be read/written in 16-bit units.
RSCFDnCFDGRMCFGLL, RSCFDnCFDGRMCFGLH, RSCFDnCFDGRMCFGHL, RSCFDnCFDGRMCFGHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGRMCFG: <RSCFDn_base> + $04FC_H$
RSCFDnCFDGRMCFGL: <RSCFDn_base> + $04FC_H$,
RSCFDnCFDGRMCFGH: <RSCFDn_base> + $04FE_H$
RSCFDnCFDGRMCFGLL: <RSCFDn_base> + $04FC_H$,
RSCFDnCFDGRMCFGLH: <RSCFDn_base> + $04FD_H$,
RSCFDnCFDGRMCFGHL: <RSCFDn_base> + $04FE_H$,
RSCFDnCFDGRMCFGHH: <RSCFDn_base> + $04FF_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RCMC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 14.92 RSCFDnCFDGRMCFG Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	RCMC	Interface Mode Select 0: Classical CAN mode 1: CAN FD mode

Note: RSCANnGRMCFG and RSCFDnCFDGRMCFG are the same register. Set either of these registers.

Modify the RSCFDnCFDGRMCFG register only in global reset mode. Before setting other RS- CANFD registers, set this register.

RCMC Bit

Setting this bit to 1 makes CAN FD mode available. To switch classical CAN mode to CAN FD mode, set the value after reset to all registers and bits allocated to the register map of classical CAN mode and then modify the RSCFDnCFDGRMCFG register.

14.4.3 Details of Channel-Related Registers

14.4.3.1 RSCFDnCFDCmNCFG ─ Channel Nominal Bit Rate Configuration Register (m = 0 to 3)

Access: RSCFDnCFDCmNCFG register can be read/written in 32-bit units.
RSCFDnCFDCmNCFGL, RSCFDnCFDCmNCFGH registers can be read/written in 16-bit units.
RSCFDnCFDCmNCFGLL, RSCFDnCFDCmNCFGLH, RSCFDnCFDCmNCFGHL, RSCFDnCFDCmNCFGHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCmNCFG: <RSCFDn_base> + 0000_H + (10_H × m)
RSCFDnCFDCmNCFGL: <RSCFDn_base> + 0000_H + (10_H × m),
RSCFDnCFDCmNCFGH: <RSCFDn_base> + 0002_H + (10_H × m)
RSCFDnCFDCmNCFGLL: <RSCFDn_base> + 0000_H + (10_H × m),
RSCFDnCFDCmNCFGLH: <RSCFDn_base> + 0001_H + (10_H × m),
RSCFDnCFDCmNCFGHL: <RSCFDn_base> + 0002_H + (10_H × m),
RSCFDnCFDCmNCFGHH: <RSCFDn_base> + 0003_H + (10_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	NTSEG2[4:0]					—	NTSEG1[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NSJW[4:0]					—	NBRP[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.93 RSCFDnCFDCmNCFG Register Contents

Bit Position	Bit Name	Function
31 to 29	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
28 to 24	NTSEG2[4:0]	Nominal Bit Rate Time Segment 2 Control b28 b27 b26 b25 b24 0 0 0 0 0: Setting prohibited 0 0 0 0 1: 2 Tq : : 1 1 1 1 0: 31 Tq 1 1 1 1 1: 32 Tq
23	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
22 to 16	NTSEG1[6:0]	Nominal Bit Rate Time Segment 1 Control b22 b21 b20 b19 b18 b17 b16 0 0 0 0 0 0 0: Setting prohibited 0 0 0 0 0 0 1: Setting prohibited 0 0 0 0 0 1 0: Setting prohibited 0 0 0 0 0 1 1: 4 Tq : : 1 1 1 1 1 1 0: 127 Tq 1 1 1 1 1 1 1: 128 Tq
15 to 11	NSJW[4:0]	Nominal Bit Rate Resynchronization Jump Width Control b15 b14 b13 b12 b11 0 0 0 0 0: 1 Tq 0 0 0 0 1: 2 Tq 0 0 0 1 0: 3 Tq : : 1 1 1 1 0: 31 Tq 1 1 1 1 1: 32 Tq
10	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
9 to 0	NBRP[9:0]	Nominal Bit Rate Prescaler Division Ratio Setting When the set value = P (0 to 1023), the nominal bit rate prescaler divides fCAN by (P + 1).

Modify the RSCFDnCFDCmNCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode, and then transition to channel communication mode or channel halt mode. For the description and settings of bit timing parameters, see **Section 14.11.1, Initial Settings**.

NTSEG2[4:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2) of nominal bit rate.

Allowed values are 2 Tq to 32 Tq, inclusive.

Set a value smaller than the value of the NTSEG1[6:0] bits.

NTSEG1[6:0] Bits

These bits specify the total length of propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1) of nominal bit rate as a Tq value.

A value of 4 to 128 Tq is settable.

NSJW[4:0] Bits

These bits specify the resynchronization jump width of nominal bit rate as a Tq value. A value of 1 to 32 Tq is settable. Specify a value equal to or smaller than the NTSEG2[4:0] value.

NBRP[9:0] Bits

The clock obtained by dividing the CAN clock (fCAN) by the nominal bit rate prescaler ((NBRP[9:0]) + 1) becomes CANmTq(N) clock (fCANTQ(N)m). One clock of the CANmTq(N) clock becomes one Time Quantum (Tq).

Set the NBRP[9:0] and DBRP[7:0] bits to the same value for the ordinary and data bit rates and the two bit rates to different values according to the segment values.

14.4.3.2 RSCFDnCFDCmCTR ─ Channel Control Register (m = 0 to 3)

Access: RSCFDnCFDCmCTR register can be read/written in 32-bit units.
RSCFDnCFDCmCTRL, RSCFDnCFDCmCTRH registers can be read/written in 16-bit units.
RSCFDnCFDCmCTRLL, RSCFDnCFDCmCTRLH, RSCFDnCFDCmCTRHL, RSCFDnCFDCmCTRHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCmCTR: <RSCFDn_base> + 0004_H + (10_H × m)
RSCFDnCFDCmCTRL: <RSCFDn_base> + 0004_H + (10_H × m),
RSCFDnCFDCmCTRH: <RSCFDn_base> + 0006_H + (10_H × m)
RSCFDnCFDCmCTRLL: <RSCFDn_base> + 0004_H + (10_H × m),
RSCFDnCFDCmCTRLH: <RSCFDn_base> + 0005_H + (10_H × m),
RSCFDnCFDCmCTRHL: <RSCFDn_base> + 0006_H + (10_H × m),
RSCFDnCFDCmCTRHH: <RSCFDn_base> + 0007_H + (10_H × m)

Value after reset: $0000\ 0005_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ROM	CRCT	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	TDCV FIE	SOC OIE	EOC OIE	TAIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 14.94 RSCFDnCFDCmCTR Register Contents (1/2)

Bit Position	Bit Name	Function
31	ROM	Restricted Operation Mode Enable 0: Restricted operation mode is disabled. 1: Restricted operation mode is enabled.
30	CRCT	CRC Error Test Enable 0: The first bit of the reception ID field is not inverted. 1: The first bit of the reception ID field is inverted.
29 to 27	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
26, 25	CTMS[1:0]	Communication Test Mode Select b26 b25 0 0: Standard test mode 0 1: Listen-only mode 1 0: Self-test mode 0 (external loopback mode) 1 1: Self-test mode 1 (internal loopback mode)
24	CTME	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	Error Display Mode Select 0: Error flags are displayed only for the first error information after bits 14 to 8 in the RSCFDnCFDCmERFL register are all cleared. 1: Error flags for all error information are displayed.
22, 21	BOM[1:0]	Bus Off Recovery Mode Select b22 b21 0 0: ISO11898-1:2015 compliant 0 1: Entry to channel halt mode automatically at bus-off entry 1 0: Entry to channel halt mode automatically at bus-off end 1 1: Entry to channel halt mode (in bus-off state) by program request
20	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
19	TDCVFIE	Transmitter Delay Compensation Violation Interrupt Enable 0: A transmitter delay compensation violation interrupt is disabled. 1: A transmitter delay compensation violation interrupt is enabled.
18	SOCOIE	Successful Occurrence Counter Overflow Interrupt Enable 0: A successful occurrence counter overflow interrupt is disabled. 1: A successful occurrence counter overflow interrupt is enabled.
17	EOCOIE	Error Occurrence Counter Overflow Interrupt Enable 0: An error occurrence counter overflow interrupt is disabled. 1: An error occurrence counter overflow interrupt is enabled.
16	TAIE	Transmit Abort Interrupt Enable 0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.
15	ALIE	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.
14	BLIE	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.
13	OLIE	Overload Frame Transmit Interrupt Enable 0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.

Table 14.94 RSCFDnCFDCmCTR Register Contents (2/2)

Bit Position	Bit Name	Function
12	BORIE	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.
11	BOEIE	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
3	RTBO	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.
2	CSLPR	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1, 0	CHMDC[1:0]	Mode Select b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

ROM Bit

When the ROM bit and the CTME bit in the RSCFDnCFDCmCTR register are set to 1, restricted operation mode is enabled. Use the restricted operation mode only when the CTMS[1:0] value in the RSCFDnCFDCmCTR register is 00_B (standard test mode). Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CRCT Bit

This bit is used to test the CRC generation circuit in the RS-CANFD module. Setting this bit to 1 inverts the first bit of the ID field when a message is received. With this inversion of bit, the CRC calculation result does not match the normal CRC value of the received frame, which can detect a CRC error (the CERR bit in the RSCFDnCFDCmERFL register is 1). When using this function, note the following.

- This function is available while the CTME bit in the RSCFDnCFDCmCTR register is 1 (communication test mode enabled).
- This function cannot communicate with other CAN nodes. Use this function for inter-channel communication test (the CmICBCE bit in the RSCFDnCFDGTSTCFG register is 1).
- Bit inversion in the ID field may cause bit stuffing rule violation. In that case, no CRC error is detected but a stuff error is detected.

Modify this bit only in channel halt mode. This bit is always 0 in channel reset mode.

CTMS[1:0] Bits

These bits are used to select a communication test mode. Modify these bits in channel halt mode only. These bits are set to 0 in channel reset mode.

CTME Bit

Setting this bit to 1 enables communication test mode. Modify these bits in channel halt mode. This bit is set to 0 in channel reset mode.

ERRD Bit

This bit is used to control the display mode of bits 14 to 8 in the RSCFDnCFDCmERFL register.

When this bit is clear to 0, if any error is detected while the flags of bits 14-8 in the RSCFDnCFDCmERFL register are all 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1.

When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order.

Modify this bit only in channel reset mode or channel halt mode.

BOM[1:0] Bits

These bits are used to select the bus off recovery mode of the RS-CANFD module.

When the BOM[1:0] bits are set to 00_B, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CANFD module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to 10_B (channel halt mode) before recessive bits are detected 128 times, the RS-CANFD module does not transition to channel halt mode until recessive bits are detected 128 times.

When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 01_B, the CHMDC[1:0] bits in the RSCFDnCFDCmCTR register (m = 0 to 3) are set to 10_B and the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCFDnCFDCmSTS register are cleared to 00_H.

When the RS-CANFD module reaches the bus off state when the BOM[1:0] bits are set to 10_B, the CHMDC[1:0] bits are set to 10_B and the RS-CANFD module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H.

When the BOM[1:0] bits are set to 11_B and the CHMDC[1:0] bits are set to 10_B while the RS-CANFD module is in the bus off state, the RS-CANFD module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H. However, if 11 consecutive recessive bits are detected 128 times and the RS-CANFD module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10_B, a bus off recovery interrupt request is generated.

If a program writes to the CHMDC[1:0] bit at the same time as the RS-CANFD module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01_B or at bus off end when the BOM[1:0] bits are 10_B), the program's writing takes precedence. Modify the BOM[1:0] bits only in channel reset mode.

TDCVFIE Bit

When the TDCVF flag in the RSCFDnCFDCmFDSTS register is set to 1 after the TDCVFIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

SOCOIE Bit

When the SOCO flag in the RSCFDnCFDCmFDSTS register is set to 1 after the SOCOIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

EOCOIE Bit

When the EOCO flag in the RSCFDnCFDCmFDSTS register is set to 1 after the EOCOIE bit is set to 1, an interrupt request occurs. Modify this bit only in channel reset mode.

TAIE Bit

When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.

ALIE Bit

When the ALF flag in the RSCFDnCFDCmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BLIE Bit

When the BLF flag in the RSCFDnCFDCmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

OLIE Bit

When the OVLF flag in the RSCFDnCFDCmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BORIE Bit

When the BORF flag in the RSCFDnCFDCmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BOEIE Bit

When the BOEF flag in the RSCFDnCFDCmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EPIE Bit

When the EPF flag in the RSCFDnCFDCmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EWIE Bit

When the EWF flag in the RSCFDnCFDCmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BEIE Bit

When the BEF flag in the RSCFDnCFDCmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

RTBO Bit

Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCFDnCFDCmSTS register to 00_H and also clears the BOSTS flag in the RSCFDnCFDCmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCFDnCFDCmCTR register are 00_B (ISO11898-1:2015 compliant).

A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RS-CANFD module transitions to the error active state. Set this bit to 1 in channel communication mode.

CSLPR Bit

Setting this bit to 1 places the channel into channel stop mode.

Clearing this bit to 0 makes the channel exit channel stop mode.

This bit should not be modified in channel communication mode or channel halt mode.

CHMDC[1:0] Bits

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see **Section 14.6.2, Channel Modes**. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to 11_B. When the RS-CANFD module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become 10_B.

14.4.3.3 RSCFDnCFDCmSTS ─ Channel Status Register (m = 0 to 3)

Access: RSCFDnCFDCmSTS register can be read/written in 32-bit units.
RSCFDnCFDCmSTSL, RSCFDnCFDCmSTSH registers can be read/written in 16-bit units.
RSCFDnCFDCmSTSLL, RSCFDnCFDCmSTSLH, RSCFDnCFDCmSTSHL, RSCFDnCFDCmSTSHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCmSTS: <RSCFDn_base> + 0008_H + (10_H × m)
RSCFDnCFDCmSTSL: <RSCFDn_base> + 0008_H + (10_H × m),
RSCFDnCFDCmSTSH: <RSCFDn_base> + $000A_H$ + (10_H × m)
RSCFDnCFDCmSTSLL: <RSCFDn_base> + 0008_H + (10_H × m),
RSCFDnCFDCmSTSLH: <RSCFDn_base> + 0009_H + (10_H × m),
RSCFDnCFDCmSTSHL: <RSCFDn_base> + $000A_H$ + (10_H × m),
RSCFDnCFDCmSTSHH: <RSCFDn_base> + $000B_H$ + (10_H × m)

Value after reset: $0000\ 0005_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ESIF	COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R	R	R/W *1	R	R	R	R	R	R	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.95 RSCFDnCFDCmSTS Register Contents

Bit Position	Bit Name	Function
31 to 24	TEC[7:0]	The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	The receive error counter (REC) can be read.
15 to 9	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
8	ESIF	Error State Indication Flag 0: No CAN FD message whose ESI bit is recessive has been received. 1: At least one CAN FD message whose ESI bit is recessive has been received.
7	COMSTS	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode
0	CRSTSTS	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

TEC[7:0] Bits

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1:2015).

These bits are cleared to 0 in channel reset mode.

REC[7:0] Bits

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1:2015).

These bits are cleared to 0 in channel reset mode.

ESIF Flag

When the recessive ESI bit is detected in a successfully received message, this flag is set to 1. In loopback mode or mirror mode, the own transmission message is regarded as a received message. To clear this flag to 0, write 0 to this bit by the program. This bit cannot be set to 1 by the program. If the flag setting (to 1) timing matches the writing 0 (by the program) timing, this flag is set to 1.

This flag is 0 in channel reset mode.

COMSTS Flag

This bit indicates that communication is ready.

This flag becomes 1 when the RS-CANFD module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.

RECSTS Flag

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

TRMSTS Flag

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

BOSTS Flag

This flag is set to 1 when the bus off state (TEC[7:0] > 255) is entered. It is cleared to 0 when the RS-CANFD module has exited the bus off state.

EPSTS Flag

This flag is set to 1 when the RS-CANFD module has entered the error passive state ((128 ≤ TEC[7:0] ≤ 255) or (128 ≤ REC[7:0])), It is cleared to 0 when the RS-CANFD module has exited the error passive state or has entered channel reset mode.

CSLPSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to channel stop mode, and is cleared to 0 when the RS-CANFD module has returned from channel stop mode.

CHLTSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to channel halt mode, and is cleared to 0 when the RS-CANFD module has returned from channel halt mode.

CRSTSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to channel reset mode, and is cleared to 0 when the RS-CANFD module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the RS-CANFD module transitions from channel reset mode to channel stop mode.

14.4.3.4 RSCFDnCFDCmERFL — Channel Error Flag Register (m = 0 to 3)

Access: RSCFDnCFDCmERFL register can be read/written in 32-bit units.
RSCFDnCFDCmERFLL, RSCFDnCFDCmERFLH registers can be read/written in 16-bit units.
RSCFDnCFDCmERFLLL, RSCFDnCFDCmERFLLH, RSCFDnCFDCmERFLHL, RSCFDnCFDCmERFLHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCmERFL: <RSCFDn_base> + $000C_H$ + (10_H × m)
RSCFDnCFDCmERFLL: <RSCFDn_base> + $000C_H$ + (10_H × m),
RSCFDnCFDCmERFLH: <RSCFDn_base> + $000E_H$ + (10_H × m)
RSCFDnCFDCmERFLLL: <RSCFDn_base> + $000C_H$ + (10_H × m),
RSCFDnCFDCmERFLLH: <RSCFDn_base> + $000D_H$ + (10_H × m),
RSCFDnCFDCmERFLHL: <RSCFDn_base> + $000E_H$ + (10_H × m),
RSCFDnCFDCmERFLHH: <RSCFDn_base> + $000F_H$ + (10_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.96 RSCFDnCFDCmERFL Register Contents (1/2)

Bit Position	Bit Name	Function
31	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
30 to 16	CRCREG[14:0]	CRC Calculation Data (CRC length:15 bits) A CRC value calculated based on the transmit message or receive message is indicated.
15	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
14	ADERR	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.

Table 14.96 RSCFDnCFDCmERFL Register Contents (2/2)

Bit Position	Bit Name	Function
10	AERR	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.
9	FERR	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.
7	ALF	Arbitration-lost Flag 0: No arbitration-lost is detected. 1: Arbitration-lost is detected.
6	BLF	Bus Lock Flag 0: No channel bus is detected. 1: Channel bus is detected.
5	OVLF	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORF	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWF	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	Bus Error Flag 0: No channel bus error is detected. 1: Channel bus error is detected.

See the CAN specification (ISO11898-1:2015) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these flags is set to 0 at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0.

If an error is detected with all of bits 14-8 of the RSCFDnCFDCmERFL register set to 0 when the ERRD bit of the RSCFDnCFDCmCTR is set to 0 (display only the first error information that occurred) for bits 14-8 of the RSCFDnCFDCmERFL register, the corresponding bit is set to 1.

CRCREG[14:0] Flag

When the CTME bit in the RSCFDnCFDCmCTR register is set to 1 (communication test mode is enabled), if transmit or receive message is a classical CAN frame (CRC length = 15 bits), this flag is updated and the CRC value calculated based on the message can be read. When a CAN FD frame is sent or received, the value of CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register is updated. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0.

ADERR Flag

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

B0ERR Flag

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

B1ERR Flag

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

CERR Flag

This flag is set to 1 when a CRC error has been detected.

AERR Flag

This flag is set to 1 when an ACK error has been detected.

FERR Flag

This flag is set to 1 when a form error has been detected.

SERR Flag

This flag is set to 1 when a stuff error has been detected.

ALF Flag

This flag is set to 1 when an arbitration-lost has been detected.

BLF Flag

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, detection of a dominant lock is restarted when either of the following conditions is met.

- A recessive bit is detected after the BLF bit has been cleared from 1 to 0.
- The RS-CANFD module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0.

OVLF Flag

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BORF Flag

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the RS-CANFD module returns from the bus off state. However, this flag is not set to 1 if the RS-CANFD module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCFDnCFDCmCTR register are set to 01_B (channel reset mode).
- The RTBO bit in the RSCFDnCFDCmCTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCFDnCFDCmCTR register are set to 01_B (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCFDnCFDCmCTR register are set to 10_B (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11_B (transition to channel halt mode upon a request from the program during bus off).

BOEF Flag

This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCFDnCFDCmCTR register (m = 0 to 3) set to 01_B (transition to channel halt mode at bus off entry).

EPF Flag

This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127). This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.

EWF Flag

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.

BEF Flag

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCFDnCFDCmERFL register is set to 1.

NOTE

To clear the flag of this register to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

14.4.3.5 RSCFDnCFDCmDCFG ─ Channel Data Bit Rate Configuration Register (m = 0 to 3)

Access: RSCFDnCFDCmDCFG register can be read/written in 32-bit units.
RSCFDnCFDCmDCFGL, RSCFDnCFDCmDCFGH registers can be read/written in 16-bit units.
RSCFDnCFDCmDCFGLL, RSCFDnCFDCmDCFGLH, RSCFDnCFDCmDCFGHL, RSCFDnCFDCmDCFGHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCmDCFG: <RSCFDn_base> + 0500_H + (20_H × m)
RSCFDnCFDCmDCFGL: <RSCFDn_base> + 0500_H + (20_H × m),
RSCFDnCFDCmDCFGH: <RSCFDn_base> + 0502_H + (20_H × m)
RSCFDnCFDCmDCFGLL: <RSCFDn_base> + 0500_H + (20_H × m),
RSCFDnCFDCmDCFGLH: <RSCFDn_base> + 0501_H + (20_H × m),
RSCFDnCFDCmDCFGHL: <RSCFDn_base> + 0502_H + (20_H × m),
RSCFDnCFDCmDCFGHH: <RSCFDn_base> + 0503_H + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	DSJW[2:0]			—	DTSEG2[2:0]			DTSEG1[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DBRP[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.97 RSCFDnCFDCmDCFG Register Contents

Bit Position	Bit Name	Function
31 to 27	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
26 to 24	DSJW[2:0]	Data Bit Rate Resynchronization Jump Width Control b26 b25 b24 0 0 0: 1 Tq 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq
23	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
22 to 20	DTSEG2[2:0]	Data Bit Rate Time Segment 2 Control b22 b21 b20 0 0 0: Setting prohibited 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq
19 to 16	DTSEG1[3:0]	Data Bit Rate Time Segment 1 Control b19 b18 b17 b16 0 0 0 0: Setting prohibited 0 0 0 1: 2 Tq 0 0 1 0: 3 Tq 0 0 1 1: 4 Tq 0 1 0 0: 5 Tq 0 1 0 1: 6 Tq 0 1 1 0: 7 Tq 0 1 1 1: 8 Tq 1 0 0 0: 9 Tq 1 0 0 1: 10 Tq 1 0 1 0: 11 Tq 1 0 1 1: 12 Tq 1 1 0 0: 13 Tq 1 1 0 1: 14 Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to 8	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
7 to 0	DBRP[7:0]	Data Bit Rate Prescaler Division Ratio Setting When the set value = P (0 to 255), the data bit rate prescaler divides fCAN by (P + 1).

Modify the RSCFDnCFDCmDCFG register in channel reset mode or channel halt mode. Set this register in channel reset mode, and then transition to channel communication mode or channel halt mode. When only classical CAN frames are used in CAN FD mode, set the RSCFDnCFDCmDCFG register to the value equal to the set RSCFDnCFDCmNCFG register value. For the description and settings of bit timing parameters, see **Section 14.11.1, Initial Settings**.

DSJW[2:0] Bits

These bits specify the resynchronization jump width of data bit rate as a Tq value. A value of 1 to 8 Tq is settable. Specify a value equal to or smaller than the DTSEG2[2:0] bits value.

DTSEG2[2:0] Bits

These bits are used to specify a Tq value for the length of phase segment 2 (PHASE_SEG2) of nominal bit rate.

Allowed values are 2 Tq to 8 Tq, inclusive.

Set a value smaller than or equal to the value of the DTSEG1[3:0] bits.

DTSEG1[3:0] Bits

These bits specify the total length of propagation segment (PROP_SEG) and phase segment 1 (PHASE_SEG1) of data bit rate as a Tq value.

A value of 2 to 16 Tq is settable.

DBRP[7:0] Bits

The clock obtained by dividing the CAN clock (fCAN) by the data bit rate prescaler ((DBRP[7:0]) + 1) becomes CANmTq(D) clock (fCANTQ(D)m). One clock of the CANmTq(D) clock becomes one Time Quantum (Tq).

Be sure to set the NBRP[9:0] and DBRP[7:0] bits to the same value.

When setting the ordinary and data bit rates to different values, change the RSCFDnCFDCmNCFG.NTSEG1 and NTSEG2 bits and the RSCFDnCFDCmDCFG.DTSEG1 and DTSEG2 bits to the desired bit rates respectively.

If the TDCE bit of the RSCFDnCFDCmFDCFG register is 1 (enabling transmitter delay compensation), set the NBRP[9:0] and DBRP[7:0] bits to the same value equal to or less than 1.

14.4.3.6 RSCFDnCFDCmFDCFG ─ Channel CAN FD Configuration Register (m = 0 to 3)

Access: RSCFDnCFDCmFDCFG register can be read/written in 32-bit units.
RSCFDnCFDCmFDCFGL, RSCFDnCFDCmFDCFGH registers can be read/written in 16-bit units.
RSCFDnCFDCmFDCFGLL, RSCFDnCFDCmFDCFGLH, RSCFDnCFDCmFDCFGHL, RSCFDnCFDCmFDCFGHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCmFDCFG: <RSCFDn_base> + 0504_H + (20_H × m)
RSCFDnCFDCmFDCFGL: <RSCFDn_base> + 0504_H + (20_H × m),
RSCFDnCFDCmFDCFGH: <RSCFDn_base> + 0506_H + (20_H × m)
RSCFDnCFDCmFDCFGLL: <RSCFDn_base> + 0504_H + (20_H × m),
RSCFDnCFDCmFDCFGLH: <RSCFDn_base> + 0505_H + (20_H × m),
RSCFDnCFDCmFDCFGHL: <RSCFDn_base> + 0506_H + (20_H × m),
RSCFDnCFDCmFDCFGHH: <RSCFDn_base> + 0507_H + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	REFE	FDOE	TMME	GWBRS	GWFDF	GWEN	—	TDCO[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ESIC	TDCE	TDCOC	—	—	—	—	—	EOCCFG[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 14.98 RSCFDnCFDCmFDCFG Register Contents

Bit Position	Bit Name	Function
31, 30	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
29	REFE	Reception data edge filter enable bit 0: Reception data edge filter disabled 1: Reception data edge filter enabled
28	FDOE	FD-only mode enable bit 0: FD-only mode disabled 1: FD-only mode enabled
27	TMME	Transmit Buffer Merge Mode Enable 0: Transmit buffer merge mode is disabled. 1: Transmit buffer merge mode is enabled.
26	GWBRS	Gateway BRS 0: A frame is transmitted with the BRS bit in the received frame set to 0. 1: A frame is transmitted with the BRS bit in the received frame set to 1.
25	GWFDF	Gateway FDF 0: A frame is transmitted regarding the received frame as a classical CAN frame. 1: A frame is transmitted regarding the received frame as a CAN FD frame.
24	GWEN	CAN-CAN FD Gateway Enable 0: CAN-CAN FD gateway is disabled. 1: CAN-CAN FD gateway is enabled.
23	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
22 to 16	TDCO[6:0]	Transmitter Delay Compensation Offset These bits are set to the transmitter delay compensation offset value.
15 to 11	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
10	ESIC	Error State Display Mode Select 0: Always transmits the error state of the channel as the ESI bit in the frame. 1: Transmits the error state of the message buffer as the ESI bit in the frame when the channel is not in the error passive state. Transmits the error state of the channel as the ESI bit in the frame when the channel is in the error passive state.
9	TDCE	Transmitter Delay Compensation Enable 0: Transmitter delay compensation is disabled. 1: Transmitter delay compensation is enabled.
8	TDCOC	Transmitter Delay Compensation Measurement Select 0: Measurement and offset 1: Only offset
7 to 3	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
2 to 0	EOCCFG[2:0]	Error Occurrence Counting Method Select b2 b1 b0 0 0 0: All transmit messages and receive messages 0 0 1: All transmit messages 0 1 0: All receive messages 0 1 1: Setting prohibited 1 0 0: Only data phase of transmitted or received CAN FD message 1 0 1: Only data phase of transmitted CAN FD message 1 1 0: Only data phase of received CAN FD message 1 1 1: Setting prohibited

REFE Bit

Setting this bit to 1 enables reception data edge filtering when the idle condition is detected, and a dominant level with less than 2 time quanta is ignored. A dominant level with more than or equal to 2 time quanta is detected as an edge. Modify this bit only in channle reset mode.

FDOE Bit

Setting this bit to 1 enables FD-only mode. When data is transmitted, a CAN FD frame will be sent regardless of the settings to the CFFDF bit in the RSCFDnCFDCFFDCSTSk register or the TMFDF bit in the RSCFDnCFDTMFDCTRp register. When a Classical CAN frame is received, a form error is detected. Modify this bit only in channel reset mode.

TMME Bit

Setting this bit to 1 enables transmit buffer merge mode. Modify this bit only in channel reset mode or channel halt mode.

GWBRS Bit

When the GWEN bit is 1, the BRS bit in a CAN FD frame to be transmitted by the gateway function is set. Write 0 to this bit to clear the GWBRS bit to 0. Modify this bit only in channel reset mode.

GWFDF Bit

When the GWEN bit is 1, the FDF bit in a CAN FD frame to be transmitted by the gateway function is set. Modify this bit only in channel reset mode.

GWEN Bit

This bit is used to control the operation of the transmit/receive FIFO buffer with the CFM[1:0] bits in the RSCFDnCFDCFCCk register set to 10_B (gateway mode).

Setting this bit to 1 enables the CAN-CAN FD gateway, enabling transmission in a format different from that of frames received by the gateway function. Received frames are replaced in accordance with the settings of the GWFDF bit and the GWBRS bit. When the DLC value in the received classical CAN frame is 1001_B or more and the GWFDF bit is 1 (CAN FD frame), the DLC value is replaced with 1000_B.

While this bit is 1, do not perform routing the following frames by using the gateway function.

- CAN FD frames with a payload length of more than 8 bytes
- Remote frames

While this bit is 1, the following frame shoudl be transmitted in the channel by setting of GWFDF.

- When GWFDF bit is set to 0, only classical CAN frame should be transmitted.
- When GWFDF bit is set to 1, only CAN FD frame should be transmitted.

Modify this bit only in channel reset mode.

Table 14.99 shows the settings and formats of transmit frame and receive frame while the CAN-CAN FD gateway is enabled.

Table 14.99 Operation when the CAN-CAN FD Gateway is Enabled

Receive Frame				Transmit Frame		
Format	BRS Bit	Received DLC Value	GWFDF Bit	Format	BRS Bit	DLC Value to be Transmitted
Classical CAN	None	DLC ≤ 1000_B	0	Classical CAN	None	Not replaced
		DLC > 1000_B				
CAN FD	Arbitrary	DLC ≤ 1000_B				
Classical CAN	None	DLC ≤ 1000_B	1	CAN FD	According to GWBRS bit setting	Not replaced
		DLC > 1000_B				Replaced with 1000_B
CAN FD	Arbitrary	DLC ≤ 1000_B				Not replaced

TDCO[6:0] Bits

These bits are set to the SSP offset value. How to use this value depends on the TDCOC bit in the RSCFDnCFDCmFDCFG register.

When the TDCOC bit is 0, the transmitter delay compensation result equals to the total value of the measured delay value and the TDCO[6:0] value (rounded to the nearest integer Tq).

When the TDCOC bit is 1, the transmitter delay compensation result equals to the TDCO[6:0] value.

The SSP offset value = (set value of TDCO[6:0] bits + 1).

Modify these bits only in channel reset mode or channel halt mode.

ESIC Bit

When the ESIC bit is set to 1, if the channel is in the error active state, the ESI bit value (CFESI bit in the RSCFDnCFDCFFDCSTSk register or TMESI bit in the RSCFDnCFDTMFDCTRp register) set in the transmit/receive FIFO buffer or transmit buffer is transmitted as an ESI bit value of the transmit message. When the channel is in the error passive state or the ESIC bit is 0, the channel status is transmitted as an ESI bit value. Modify this bit only in channel reset mode or channel halt mode.

Table 14.100 ESI Value to be Transmitted

ESIC Bit	Channel Status	ESI Value to be Transmitted
0	Error active	0 (error active node)
	Error passive	1 (error passive node)
1	Error active	ESI value set in the transmit/receive FIFO buffer or transmit buffer (CFESI bit in the RSCFDnCFDCFFDCSTSk register or TMESI bit in the RSCFDnCFDTMFDCTRp register)
	Error passive	1 (error passive node)

TDCE Bit

Setting this bit to 1 enables transmitter delay compensation. Modify this bit only in channel reset mode or channel halt mode.

TDCOC Bit

When this bit is 0, the SSP position is defined by the total of the measured delay value and the SSP offset value (fixed value).

When this bit is 1, the SSP position is defined only by the SSP offset value.

Modify this bit only in channel reset mode or channel halt mode.

EOCCFG[2:0] Bits

These bits are used to select a frame format and a transmission/reception direction when the error occurrence counter counts CAN bus errors.

Modify these bits only in channel reset mode or channel halt mode.

14.4.3.7 RSCFDnCFDCmFDCTR ─ Channel CAN FD Control Register (m = 0 to 3)

Access: RSCFDnCFDCmFDCTR register can be read/written in 32-bit units.
RSCFDnCFDCmFDCTRL, RSCFDnCFDCmFDCTRH registers can be read/written in 16-bit units.
RSCFDnCFDCmFDCTRLL, RSCFDnCFDCmFDCTRLH, RSCFDnCFDCmFDCTRHL, RSCFDnCFDCmFDCTRHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCmFDCTR: <RSCFDn_base> + 0508_H + (20_H × m)
RSCFDnCFDCmFDCTRL: <RSCFDn_base> + 0508_H + (20_H × m),
RSCFDnCFDCmFDCTRH: <RSCFDn_base> + $050A_H$ + (20_H × m)
RSCFDnCFDCmFDCTRLL: <RSCFDn_base> + 0508_H + (20_H × m),
RSCFDnCFDCmFDCTRLH: <RSCFDn_base> + 0509_H + (20_H × m),
RSCFDnCFDCmFDCTRHL: <RSCFDn_base> + $050A_H$ + (20_H × m),
RSCFDnCFDCmFDCTRHH: <RSCFDn_base> + $050B_H$ + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOC CLR	EOC CLR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 14.101 RSCFDnCFDCmFDCTR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
1	SOCCLR	Successful Occurrence Counter Clear Setting the SOCCLR bit to 1 clears the successful occurrence counter. This bit is always read as 0.
0	EOCCLR	Error Occurrence Counter Clear Setting the EOCCLR bit to 1 clears the error occurrence counter. This bit is always read as 0.

SOCCLR Bit

Setting this bit to 1 clears the successful occurrence counter (SOC[7:0] bits in the RSCFDnCFDCmFDSTS register). This bit is automatically cleared to 0.

EOCCLR Bit

Setting this bit to 1 clears the error occurrence counter (EOC[7:0] bits in the RSCFDnCFDCmFDSTS register). This bit is automatically cleared to 0.

14.4.3.8 RSCFDnCFDCmFDSTS – Channel CAN FD Status Register (m = 0 to 3)

Access: RSCFDnCFDCmFDSTS register can be read/written in 32-bit units.
RSCFDnCFDCmFDSTSL, RSCFDnCFDCmFDSTSH registers can be read/written in 16-bit units.
RSCFDnCFDCmFDSTSLL, RSCFDnCFDCmFDSTSLH, RSCFDnCFDCmFDSTSHL, RSCFDnCFDCmFDSTSHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCmFDSTS: <RSCFDn_base> + $050C_H$ + (20_H × m)
RSCFDnCFDCmFDSTSL: <RSCFDn_base> + $050C_H$ + (20_H × m),
RSCFDnCFDCmFDSTSH: <RSCFDn_base> + $050E_H$ + (20_H × m)
RSCFDnCFDCmFDSTSLL: <RSCFDn_base> + $050C_H$ + (20_H × m),
RSCFDnCFDCmFDSTSLH: <RSCFDn_base> + $050D_H$ + (20_H × m),
RSCFDnCFDCmFDSTSHL: <RSCFDn_base> + $050E_H$ + (20_H × m),
RSCFDnCFDCmFDSTSHH: <RSCFDn_base> + $050F_H$ + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SOC[7:0]								EOC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SOCO	EOCO	TDCVF	TDCR[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W *1	R/W *1	R/W *1	R	R	R	R	R	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.102 RSCFDnCFDCmFDSTS Register Contents

Bit Position	Bit Name	Function
31 to 24	SOC[7:0]	Successful Occurrence Counter The successful occurrence counter value can be read.
23 to 16	EOC[7:0]	Error Occurrence Counter The error occurrence counter value can be read.
15 to 10	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
9	SOCO	Successful Occurrence Counter Overflow Flag 0: The successful occurrence counter does not overflow. 1: The successful occurrence counter has overflowed.
8	EOCO	Error Occurrence Counter Overflow Flag 0: The error occurrence counter does not overflow. 1: The error occurrence counter has overflowed.
7	TDCVF	Transmitter Delay Compensation Violation Flag 0: No transmitter delay compensation violation is present. 1: A transmitter delay compensation violation is present.
6 to 0	TDCR[6:0]	Transmitter Delay Compensation Result Status The transmitter delay compensation result can be read.

SOC[7:0] Bits

These bits show the successful occurrence counter value. The successful occurrence counter is incremented upon completion of message reception or transmission without an error. This counter stops counting when it reaches FF_H. In loopback mode, this counter is incremented twice.

These bits are cleared to 0 by writing 1 to the SOCCLR bit in the RSCFDnCFDCmCTR register. These bits are 0 in channel reset mode.

EOC[7:0] Bits

These bits show the error occurrence counter value. The error occurrence counter is incremented each time an error occurs according to the condition specified by the EOCCFG[2:0] bits in the RSCFDnCFDCmFDCFG register. This counter stops counting when it reaches FF_H.

These bits are cleared to 0 by writing 1 to the EOCCLR bit in the RSCFDnCFDCmCTR register. These bits are 0 in channel reset mode.

SOCO Flag

This bit indicates that successful occurrence counter overflow has occurred.

This flag is set to 1 when message reception or transmission is completed while the SOC[7:0] value has reached FF_H. This flag is 0 in channel reset mode.

EOCO Flag

This bit indicates that error occurrence counter overflow has occurred.

This flag is set to 1 when a CAN bus error is detected under the condition specified by the EOCCFG[2:0] bits in the RSCFDnCFDCmFDCFG register when the EOC[7:0] value has reached FF_H. This flag is 0 in channel reset mode.

TDCVF Flag

This bit indicates violation of transmitter delay compensation.

The transmit data is compared with the reception CAN bus level delayed due to the transceiver's loop delay. This delay changes due to physical factors such as temperature. Because the TDCR[6:0] flags are updated for each message, temporary maximum delay cannot be confirmed.

This bit is set to 1 when the transmitter delay compensation exceeds the maximum compensation 3 CANm bit times - 2 fCAN (CANm bit time is the value of data bit rate).

This flag is 0 in channel reset mode.

TDCR[6:0] Flags

These bits indicate the transmitter delay compensation result as a multiple of CAN clock frequency (fCAN).

This result depends on the settings of the TDCOC bit and TDCO[6:0] bits in the RSCFDnCFDCmFDCFG register.

This flag is updated at a falling edge between the FDF bit and res bit when the TDCE bit (transmitter delay compensation enable) in the RSCFDnCFDCmFDCFG register is set to 1 and also the TDCOC bit (transmitter delay compensation measurement select) in the RSCFDnCFDCmFDCFG register is set to 0.

This flag is 0 in channel reset mode.

NOTE

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

14.4.3.9 RSCFDnCFDCmFDCRC ─ Channel CAN FD CRC Register (m = 0 to 3)

Access: RSCFDnCFDCmFDCRC register can be read only in 32-bit units.
RSCFDnCFDCmFDCRCL, RSCFDnCFDCmFDCRCH registers can be read only in 16-bit units.
RSCFDnCFDCmFDCRCLL, RSCFDnCFDCmFDCRCLH, RSCFDnCFDCmFDCRCHL, RSCFDnCFDCmFDCRCHH registers can be read only in 8-bit units.

Address: RSCFDnCFDCmFDCRC: <RSCFDn_base> + 0510_H + (20_H × m)
RSCFDnCFDCmFDCRCL: <RSCFDn_base> + 0510_H + (20_H × m)
RSCFDnCFDCmFDCRCH: <RSCFDn_base> + 0512_H + (20_H × m)
RSCFDnCFDCmFDCRCLL: <RSCFDn_base> + 0510_H + (20_H × m)
RSCFDnCFDCmFDCRCLH: <RSCFDn_base> + 0511_H + (20_H × m)
RSCFDnCFDCmFDCRCHL: <RSCFDn_base> + 0512_H + (20_H × m)
RSCFDnCFDCmFDCRCHH: <RSCFDn_base> + 0513_H + (20_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	SCNT[3:0]				—	—	—	CRCREG[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRCREG[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.103 RSCFDnCFDCmFDCRC Register Contents

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, the value after a reset is read.
27 to 24	SCNT[3:0]	Stuff count bit Indicate a value of the stuff count in a CAN FD frame. Bits 25-27 indicates the Gray-coded value of the stuff bit count modulo 8 in the transsmitted/received frames. Bit 24 indicates an even parity value of bits 25-27.
23 to 21	Reserved	When read, the value after a reset is read.
20 to 0	CRCREG[20:0]	CRC Calculation Data (CRC Length:17 Bit or 21 Bit) These bits show the CRC value calculated based on the transmit message or receive message. When the CRC length is 17 bits, bits b20 to b17 are read as 0.

SCNT[3:0] flags

When the CTME bit in the RSCFDnCFDCmCTR register is set to 1 (communication test mode enabled), a stuff count bit value of the CAN FD frame can be read if a message transmitted/received is a CAN FD frame. When the CTME bit is 0 (communication test mode disabled), this flag is always read as 0. This flag is updated at the first bit in the CRC field of the CAN FD frame. These bits are cleared to 0 in channel reset mode.

CRCREG[20:0] flags

When the CTME bit in the RSCFDnCFDCmCTR register is 1 (communication test mode enabled), if transmit or receive message is a CAN FD frame (CRC length = 17 or 21 bits), these flags are updated and the CRC value calculated based on the message can be read. When the CRC length of the message is 17 bits, bits b20 to b17 are always read as 0. When a classical CAN frame is transmitted or received, the CRCREG[14:0] value in the RSCFDnCFDCmERFL register is updated. When the CTME bit is 0 (communication test mode disabled), these bits are always read as 0.

14.4.4 Details of Global-Related Registers

14.4.4.1 RSCFDnCFDGCFG ─ Global Configuration Register

Access: RSCFDnCFDGCFG register can be read/written in 32-bit units.
RSCFDnCFDGCFGL, RSCFDnCFDGCFGH registers can be read/written in 16-bit units.
RSCFDnCFDGCFGLL, RSCFDnCFDGCFGLH, RSCFDnCFDGCFGHL, RSCFDnCFDGCFGHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGCFG: <RSCFDn_base> + 0084_H
RSCFDnCFDGCFGL: <RSCFDn_base> + 0084_H, RSCFDnCFDGCFGH: <RSCFDn_base> + 0086_H
RSCFDnCFDGCFGLL: <RSCFDn_base> + 0084_H, RSCFDnCFDGCFGLH: <RSCFDn_base> + 0085_H
RSCFDnCFDGCFGHL: <RSCFDn_base> + 0086_H, RSCFDnCFDGCFGHH: <RSCFDn_base> + 0087_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				—	—	CMP OC	DCS	MME	DRE	DCE	TPRI
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.104 RSCFDnCFDGCFG Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 16	ITRCP[15:0]	Interval Timer Prescaler Set When these bits are set to M, the pclk is divided by M. Setting 0000_H is prohibited when the interval timer is in use.
15 to 13	TSBTCS[2:0]	Timestamp Clock Source Select b15 b14 b13 0 0 0: Channel 0 nominal bit time clock 0 0 1: Channel 1 nominal bit time clock 0 1 0: Channel 2 nominal bit time clock 0 1 1: Channel 3 nominal bit time clock 1 0 0: Setting prohibited 1 0 1: Setting prohibited 1 1 0: Setting prohibited 1 1 1: Setting prohibited
12	TSSS	Timestamp Source Select 0: pclk/2*1 1: Nominal bit time clock

Table 14.104 RSCFDnCFDGCFG Register Contents (2/2)

Bit Position	Bit Name	Function
11 to 8	TSP[3:0]	Timestamp Clock Source Division b11 b10 b9 b8 0 0 0 0: Not divided 0 0 0 1: Divided by 2 0 0 1 0: Divided by 4 0 0 1 1: Divided by 8 0 1 0 0: Divided by 16 0 1 0 1: Divided by 32 0 1 1 0: Divided by 64 0 1 1 1: Divided by 128 1 0 0 0: Divided by 256 1 0 0 1: Divided by 512 1 0 1 0: Divided by 1024 1 0 1 1: Divided by 2048 1 1 0 0: Divided by 4096 1 1 0 1: Divided by 8192 1 1 1 0: Divided by 16384 1 1 1 1: Divided by 32768
7, 6	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
5	CMPOC	Payload Overflow Mode Select 0: No message is stored. 1: Messages are stored and payloads exceeding the buffer size are discarded.
4	DCS	CAN Clock Source Select 0: clkc 1: clk_xincan
3	MME	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000_B.

Modify the RSCFDnCFDGCFG register only in global reset mode.

ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. See **Section 14.8.3.1, Interval Transmission Function**.

TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the nominal bit time clock that will be the clock source of the timestamp counter. However, do not select the channel that handles the CAN FD frames.

TSSS Bit

This bit is used to select a clock source of the timestamp counter. Select pclk if there is no channel that handles only classical CAN frames.

TSP[3:0] Bits

A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.

CMPOC Bit

This bit is used to select operation in case the payload length of received message exceeds the payload storage size of the storage buffer.

When this bit is 0, the received message in which the payload overflows is not stored in the buffer.

When this bit is 1, the received message in which the payload overflows is stored in the buffer. In this case, the value of the DLC field, either that in the received message or from the receive rule table, depending on the value of the DRE bit, is stored in the buffer with the payload truncated to fit the available space.

The buffer's payload storage size is set by the following bits.

- Receive buffer: RMPLS[1:0] bits in the RSCFDnCFDRMNB register
- Receive FIFO buffer: RFPLS[2:0] bits in the RSCFDnCFDRFCCx register
- Transmit/receive FIFO buffer: CFPLS[2:0] bits in the RSCFDnCFDCFCCk register

DCS Bit

When this bit is set to 0, clkc is used as the clock source of the CAN clock (fCAN).

When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (fCAN).

MME Bit

Setting this bit to 1 makes the mirror function available.

DRE Bit

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of 00_H is stored in each data byte beyond the DLC value of the receive rule.

The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

DCE Bit

Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCFDnCFDGAFLP0_j register to 0000_B before clearing the DCE bit in the RSCFDnCFDGCFG register to 0.

TPRI Bit

This bit is used to set the transmit priority.

When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1:2015 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those has the highest priority.

While the transmit queue is in use, this bit should be set to 0.

14.4.4.2 RSCFDnCFDGCTR – Global Control Register

Access: RSCFDnCFDGCTR register can be read/written in 32-bit units.
RSCFDnCFDGCTRL, RSCFDnCFDGCTRH registers can be read/written in 16-bit units.
RSCFDnCFDGCTRLL, RSCFDnCFDGCTRLH, RSCFDnCFDGCTRHL, RSCFDnCFDGCTRHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGCTR: <RSCFDn_base> + 0088_H
RSCFDnCFDGCTRL: <RSCFDn_base> + 0088_H, RSCFDnCFDGCTRH: <RSCFDn_base> + $008A_H$
RSCFDnCFDGCTRLL: <RSCFDn_base> + 0088_H, RSCFDnCFDGCTRLH: <RSCFDn_base> + 0089_H,
RSCFDnCFDGCTRHL: <RSCFDn_base> + $008A_H$, RSCFDnCFDGCTRHH: <RSCFDn_base> + $008B_H$

Value after reset: $0000\ 0005_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CMPOFIE	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 14.105 RSCFDnCFDGCTR Register Contents

Bit Position	Bit Name	Function
31 to 17	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
16	TSRST	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 12	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
11	CMPOFIE	Payload Overflow Interrupt Enable 0: A payload overflow interrupt is disabled. 1: A payload overflow interrupt is enabled.
10	THLEIE	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	FIFO Message Lost Interrupt Enable 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	GSLPR	Global Stop Mode 0: Other than global stop mode 1: Global stop mode
1, 0	GMDC[1:0]	Global Mode Select b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCFDnCFDGTSC register is cleared to 0000_H.

CMPOFIE Bit

When the CMPOF flag in the RSCFDnCFDGERFL register is set to 1 after the CMPOFIE bit is set to 1, an interrupt request occurs. Modify this bit only in global reset mode.

THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

MEIE Bit

When the MEIE bit is set to 1 and the MES flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

DEIE Bit

When the DEIE bit is set to 1 and the DEF flag in the RSCFDnCFDGERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

GSLPR Bit

Setting this bit to 1 places the RS-CANFD module into global stop mode.

Clearing this bit to 0 makes the RS-CANFD module leave from global stop mode.

This bit should not be modified in global operating mode or global test mode.

GMDC[1:0] Bits

These bits are used to select the mode of entire RS-CANFD module (global operating mode, global reset mode, or global test mode). For details, see **Section 14.6.1, Global Modes**. Setting the GSLPR bit to 1 when in global reset mode places the RS-CANFD module into global stop mode.

14.4.4.3 RSCFDnCFDGSTS – Global Status Register

Access: RSCFDnCFDGSTS register can be read only in 32-bit units.
RSCFDnCFDGSTSL, RSCFDnCFDGSTSH registers can be read only in 16-bit units.
RSCFDnCFDGSTSLL, RSCFDnCFDGSTSLH, RSCFDnCFDGSTSHL, RSCFDnCFDGSTSHH registers can be read only in 8-bit units.

Address: RSCFDnCFDGSTS: <RSCFDn_base> + $008C_H$
RSCFDnCFDGSTSL: <RSCFDn_base> + $008C_H$, RSCFDnCFDGSTSH: <RSCFDn_base> + $008E_H$
RSCFDnCFDGSTSLL: <RSCFDn_base> + $008C_H$, RSCFDnCFDGSTSLH: <RSCFDn_base> + $008D_H$
RSCFDnCFDGSTSHL: <RSCFDn_base> + $008E_H$, RSCFDnCFDGSTSHH: <RSCFDn_base> + $008F_H$

Value after reset: $0000\ 000D_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAM INIT	GSLP STS	GHLT STS	GRST STS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.106 RSCFDnCFDGSTS Register Contents

Bit Position	Bit Name	Function
31 to 4	Reserved	When read, the value after reset is returned.
3	GRAMINIT	CAN RAM Initialization Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.

This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.

GSLPSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to global stop mode, and is cleared to 0 when the RS-CANFD module has returned from global stop mode.

GHLTSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to global test mode, and is cleared to 0 when the RS-CANFD module has exited global test mode.

GRSTSTS Flag

This flag is set to 1 when the RS-CANFD module has transitioned to global reset mode, and is cleared to 0 when the RS-CANFD module has exited global reset mode. This flag remains 1 even when the RS-CANFD module has transitioned from global reset mode to global stop mode.

14.4.4.4 RSCFDnCFDGERFL – Global Error Flag Register

Access: RSCFDnCFDGERFL register can be read/written in 32-bit units.
RSCFDnCFDGERFLL, RSCFDnCFDGERFLH registers can be read/written in 16-bit units.
RSCFDnCFDGERFLLL, RSCFDnCFDGERFLLH, RSCFDnCFDGERFLHL, RSCFDnCFDGERFLHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGERFL: <RSCFDn_base> + 0090_H
RSCFDnCFDGERFLL: <RSCFDn_base> + 0090_H, RSCFDnCFDGERFLH: <RSCFDn_base> + 0092_H
RSCFDnCFDGERFLLL: <RSCFDn_base> + 0090_H, RSCFDnCFDGERFLLH: <RSCFDn_base> + 0091_H
RSCFDnCFDGERFLHL: <RSCFDn_base> + 0092_H, RSCFDnCFDGERFLHH: <RSCFDn_base> + 0093_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	EEF3	EEF2	EEF1	EEF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R/W *1	R/W *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CMPOF	THLES	MES	DEF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R	R	R/W *1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.107 RSCFDnCFDGERFL Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after reset is returned. When writing these bits, write the value after reset.
19	EEF3	ECC Error Flag for Channel 3 0: No 2-bit ECC error when deciding transmission priority 1: A 2-bit ECC error when deciding transmission priority
18	EEF2	ECC Error Flag for Channel 2 0: No 2-bit ECC error when deciding transmission priority 1: A 2-bit ECC error when deciding transmission priority
17	EEF1	ECC Error Flag for Channel 1 0: No 2-bit ECC error when deciding transmission priority 1: A 2-bit ECC error when deciding transmission priority
16	EEF0	ECC Error Flag for Channel 0 0: No 2-bit ECC error when deciding transmission priority 1: A 2-bit ECC error when deciding transmission priority
15 to 4	Reserved	When read, the undefined value is returned. When writing these bits, write the value after reset
3	CMPOF	Payload Overflow Flag 0: No payload overflow has occurred. 1: A payload overflow has occurred.
2	THLES	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.
1	MES	FIFO Message Lost Status Flag 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.
0	DEF	DLC Error Flag 0: No DLC error has occurred. 1: A DLC error has occurred.

All flags in the RSCFDnCFDGERFL register are cleared to 0 in global reset mode.

EEFm Flag

When a 2-bit ECC error is detected during the transmission priority determination of channel m (m = 0 to 3), the EEFm flag is set to 1, disabling message transmission. This flag can be cleared to 0 by writing 0 by the program.

CMPOF Flag

When a payload overflow occurs in any of channel m (m = 0 to 3), the CMPOF flag is set to 1. This flag can be cleared to 0 by writing 0 to this bit by the program.

THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCFDnCFDTHLSTSm register (m = 0 to 3) is set to 1.

This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in the RSCFDnCFDRFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCFDnCFDCFSTSk register (k = 0 to 11) is set to 1.

This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.

NOTE

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

14.4.4.5 RSCFDnCFDGTSC ─ Global Timestamp Counter Register

Access: RSCFDnCFDGTSC register can be read only in 32-bit units.
RSCFDnCFDGTSCL, RSCFDnCFDGTSCH registers can be read only in 16-bit units.

Address: RSCFDnCFDGTSC: <RSCFDn_base> + 0094_H
RSCFDnCFDGTSCL: <RSCFDn_base> + 0094_H, RSCFDnCFDGTSCH: <RSCFDn_base> + 0096_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.108 RSCFDnCFDGTSC Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned.
15 to 0	TS[15:0]	Timestamp Value The timestamp counter value can be read. Counter Value: 0000_H to $FFFF_H$

TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. Furthermore, the TS[15:0] value is stored in the transmit history buffer. The timestamp counter is initialized in global reset mode.

The timestamp counter starts and stops counting differently, depending on the count source.

- When the TSSS bit in the RSCFDnCFDGCFG register is 0 (pclk):
  The timestamp counter starts counting when the RS-CANFD module has transitioned to global operating mode.
  This counter stops counting when the RS-CANFD module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CANm nominal bit time clock):
  The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode.
  This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

14.4.4.6 RSCFDnCFDGTINTSTS0 ─ Global TX Interrupt Status Register 0

Access: RSCFDnCFDGTINTSTS0 register can be read only in 32-bit units.
RSCFDnCFDGTINTSTS0L, RSCFDnCFDGTINTSTS0H registers can be read only in 16-bit units.
RSCFDnCFDGTINTSTS0LL, RSCFDnCFDGTINTSTS0LH, RSCFDnCFDGTINTSTS0HL,
RSCFDnCFDGTINTSTS0HH registers can be read only in 8-bit units.

Address: RSCFDnCFDGTINTSTS0: <RSCFDn_base> + 0460_H
RSCFDnCFDGTINTSTS0L: <RSCFDn_base> + 0460_H, RSCFDnCFDGTINTSTS0H: <RSCFDn_base> + 0462_H
RSCFDnCFDGTINTSTS0LL: <RSCFDn_base> + 0460_H, RSCFDnCFDGTINTSTS0LH: <RSCFDn_base> + 0461_H,
RSCFDnCFDGTINTSTS0HL: <RSCFDn_base> + 0462_H, RSCFDnCFDGTINTSTS0HH: <RSCFDn_base> + 0463_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	THIF3	CFTIF3	TQIF3	TAIF3	TSIF3	—	—	—	THIF2	CFTIF2	TQIF2	TAIF2	TSIF2
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

Table 14.109 RSCFDnCFDGTINTSTS0 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 29	Reserved	When read, the value after a reset is read.
28	THIF3	Channel 3 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
27	CFTIF3	Channel 3 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
26	TQIF3	Channel 3 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
25	TAIF3	Channel 3 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
24	TSIF3	Channel 3 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

Table 14.109 RSCFDnCFDGTINTSTS0 Register Contents (2/2)

Bit Position	Bit Name	Function
23 to 21	Reserved	When read, the value after a reset is read.
20	THIF2	Channel 2 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
19	CFTIF2	Channel 2 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
18	TQIF2	Channel 2 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
17	TAIF2	Channel 2 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
16	TSIF2	Channel 2 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
15 to 13	Reserved	When read, the value after a reset is read.
12	THIF1	Channel 1 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
11	CFTIF1	Channel 1 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
10	TQIF1	Channel 1 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
9	TAIF1	Channel 1 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
8	TSIF1	Channel 1 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
7 to 5	Reserved	When read, the value after a reset is read.
4	THIF0	Channel 0 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF0	Channel 0 Transmit/receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
2	TQIF0	Channel 0 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TAIF0	Channel 0 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
0	TSIF0	Channel 0 Transmit Buffer Interrupt Status Flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

TSIFm Bits

The TSIFm bit is set to 1 when the TMIEp bit in the RSCFDnCFDTMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDTMSTSp register are set to 10_B (transmit completed without abort request) or 11_B (transmit completed with abort request).

When the TMTRF[1:0] flags are cleared to 00_B under the condition that the TSIFm bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIEp bit to 0 also clears this flag to 0.

TAIFm Bits

The TAIFm bit is set to 1 when the TAIE bit in the RSCFDnCFDCmCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCFDnCFDTMSTSp register are set to 01_B (transmit abort completed).

This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00_B after the transmit abort is completed.

TQIFm Bits

When the TXQIE bit in the RSCFDnCFDTXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCFDnCFDTXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFm bit is set to 1.

When the TXQIF bit (transmit queue interrupt request) in the RSCFDnCFDTXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.

CFTIFm Bits

When the CFTXIE bit in the RSCFDnCFDCFCCk register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCFDnCFDCFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFm bit is set to 1.

When the CFTXIF bit is cleared to 0 under the conditions that the CFTIFm bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.

THIFm Bits

When the THLIE bit in the RSCFDnCFDTHLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCFDnCFDTHLSTSm register is set to 1 (transmit history interrupt request), the THIFm bit is set to 1.

When the THLIF bit in the RSCFDnCFDTHLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.

14.4.4.7 RSCFDnCFDGFDCFG ─ Global FD Configuration Register

Access: RSCFDnCFDGFDCFG register can be read/written in 32-bit units.
RSCFDnCFDGFDCFG L, RSCFDnCFDGFDCFG H registers can be read/written in 16-bit units.
RSCFDnCFDGFDCFG LL, RSCFDnCFDGFDCFG LH, RSCFDnCFDGFDCFG HL, RSCFDnCFDGFDCFG HH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGFDCFG: <RSCFDn_base> + 0474_H
RSCFDnCFDGFDCFG L: <RSCFDn_base> + 0474_H, RSCFDnCFDGFDCFG H: <RSCFDn_base> + 0476_H
RSCFDnCFDGFDCFG LL: <RSCFDn_base> + 0474_H, RSCFDnCFDGFDCFG LH: <RSCFDn_base> + 0475_H,
RSCFDnCFDGFDCFG HL: <RSCFDn_base> + 0476_H, RSCFDnCFDGFDCFG HH: <RSCFDn_base> + 0477_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSCCFG[1:0]		—	—	—	—	—	—	—	RPED
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 14.110 RSCFDnCFDGFDCFG Register Contents

Bit Position	Bit Name	Function
31 to 10	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
9, 8	TSCCFG[1:0]	Time-stamp capture setting bit b9 b8 0 0: Captured at a sample point in the SOF bit. 0 1: Captured when a valid frame has been transmitted/received. 1 0: Captured at a sample point of the res bit.*1 1 1: Setting prohibited
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	RPED	Protocol exception event detection disabled bit 0: Protocol exception event detection enabled 1: Protocol exception event detection disabled

Note 1. When a Classical CAN frame is transmitted/received, a time-stamp value will be captured at the sample point in the SOF bit.

TSCCFG Bit

Select a point where a time-stamp value is captured. Modify this bit only in global reset mode.

RPED Bit

Setting this bit to 1 disables the protocol exception event detection. When a protocol exception event is detected while this bit is set to 1, the event is regarded as a form error and an error frame will be output. Modify this bit only in global reset mode.

14.4.4.8 RSCFDnCFDGCRCCFG ─ Global CRC Configuration Register

Access: RSCFDnCFDGCRCCFG register can be read/written in 32-bit units.
RSCFDnCFDGCRCCFG L, RSCFDnCFDGCRCCFG H registers can be read/written in 16-bit units.
RSCFDnCFDGCRCCFG LL, RSCFDnCFDGCRCCFG LH, RSCFDnCFDGCRCCFG HL, RSCFDnCFDGCRCCFG HH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGCRCCFG: <RSCFDn_base> + 0478_H
RSCFDnCFDGCRCCFG L: <RSCFDn_base> + 0478_H, RSCFDnCFDGCRCCFG H: <RSCFDn_base> + $047A_H$
RSCFDnCFDGCRCCFG LL: <RSCFDn_base> + 0478_H, RSCFDnCFDGCRCCFG LH: <RSCFDn_base> + 0479_H,
RSCFDnCFDGCRCCFG HL: <RSCFDn_base> + $047A_H$, RSCFDnCFDGCRCCFG HH: <RSCFDn_base> + $047B_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 14.111 RSCFDnCFDGCRCCFG Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	NIE	ISO11898-1:2015 protocol switching bit 0: Supports CAN FD that conforms to the ISO11898-1:2015 protocol. 1: Supports CAN FD that conforms to the ISO/CD 11898-1 (2014-08-12 version) protocol (non-compliant to ISO11898-1:2015).

NIE Bit

A 1 in this bit causes the product to support CAN FD that conforms to the ISO/CD 11898-1 (2014-08-12 version). In this case, the product is non-compliant to the ISO11898-1:2015 protocol. Modify this bit only in global reset mode. This bit is set to 0 when the RCMC bit of the RSCFDn(CFD)GRMCFG register is set to 0 (classical CAN mode).

14.4.5 Details of Receive Rule-Related Registers

14.4.5.1 RSCFDnCFDGAFLECTR ─ Receive Rule Entry Control Register

Access: RSCFDnCFDGAFLECTR register can be read/written in 32-bit units.
RSCFDnCFDGAFLECTRL, RSCFDnCFDGAFLECTRH registers can be read/written in 16-bit units.
RSCFDnCFDGAFLECTRLL, RSCFDnCFDGAFLECTRLH, RSCFDnCFDGAFLECTRHL, RSCFDnCFDGAFLECTRHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGAFLECTR: <RSCFDn_base> + 0098_H
RSCFDnCFDGAFLECTRL: <RSCFDn_base> + 0098_H, RSCFDnCFDGAFLECTRH: <RSCFDn_base> + $009A_H$
RSCFDnCFDGAFLECTRLL: <RSCFDn_base> + 0098_H, RSCFDnCFDGAFLECTRLH: <RSCFDn_base> + 0099_H
RSCFDnCFDGAFLECTRHL: <RSCFDn_base> + $009A_H$, RSCFDnCFDGAFLECTRHH: <RSCFDn_base> + $009B_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFLDAE	—	—	—	AFLPN[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 14.112 RSCFDnCFDGAFLECTR Register Contents

Bit Position	Bit Name	Function
31 to 9	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
8	AFLDAE	Receive Rule Table Write Enable 0: Receive rule table write is disabled. 1: Receive rule table write is enabled.
7 to 5	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
4 to 0	AFLPN[4:0]	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000_B) to page 15 (01111_B).

AFLDAE Bit

Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit.

Set the AFLDAE bit to 1 only in global reset mode.

AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page.

Set these bits to a value within the range of 00000_B to 01111_B.

14.4.5.2 RSCFDnCFDGAFLCFG0 – Receive Rule Configuration Register 0

Access: RSCFDnCFDGAFLCFG0 register can be read/written in 32-bit units.
RSCFDnCFDGAFLCFG0L, RSCFDnCFDGAFLCFG0H registers can be read/written in 16-bit units.
RSCFDnCFDGAFLCFG0LL, RSCFDnCFDGAFLCFG0LH, RSCFDnCFDGAFLCFG0HL, RSCFDnCFDGAFLCFG0HH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGAFLCFG0: <RSCFDn_base> + $009C_H$
RSCFDnCFDGAFLCFG0L: <RSCFDn_base> + $009C_H$, RSCFDnCFDGAFLCFG0H: <RSCFDn_base> + $009E_H$
RSCFDnCFDGAFLCFG0LL: <RSCFDn_base> + $009C_H$, RSCFDnCFDGAFLCFG0LH: <RSCFDn_base> + $009D_H$,
RSCFDnCFDGAFLCFG0HL: <RSCFDn_base> + $009E_H$, RSCFDnCFDGAFLCFG0HH: <RSCFDn_base> + $009F_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RNC2[7:0]								RNC3[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.113 RSCFDnCFDGAFLCFG0 Register Contents

Bit Position	Bit Name	Function
31 to 24	RNC0[7:0]	Number of Rules for Channel 0 Set the number of receive rules exclusively used for channel 0.
23 to 16	RNC1[7:0]	Number of Rules for Channel 1 Set the number of receive rules exclusively used for channel 1.
15 to 8	RNC2[7:0]	Number of Rules for Channel 2 Set the number of receive rules exclusively used for channel 2.
7 to 0	RNC3[7:0]	Number of Rules for Channel 3 Set the number of receive rules exclusively used for channel 3.

Modify the RSCFDnCFDGAFLCFG0 register only in global reset mode.

Up to 64 × (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC2[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 2 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC3[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 3 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

14.4.5.3 RSCFDnCFDGAFLIDj ─ Receive Rule ID Register (j = 0 to 15)

Access: RSCFDnCFDGAFLIDj register can be read/written in 32-bit units.
RSCFDnCFDGAFLIDjL, RSCFDnCFDGAFLIDjH registers can be read/written in 16-bit units.
RSCFDnCFDGAFLIDjLL, RSCFDnCFDGAFLIDjLH, RSCFDnCFDGAFLIDjHL, RSCFDnCFDGAFLIDjHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGAFLIDj: <RSCFDn_base> + 1000_H + (10_H × j)
RSCFDnCFDGAFLIDjL: <RSCFDn_base> + 1000_H + (10_H × j)
RSCFDnCFDGAFLIDjH: <RSCFDn_base> + 1002_H + (10_H × j)
RSCFDnCFDGAFLIDjLL: <RSCFDn_base> + 1000_H + (10_H × j)
RSCFDnCFDGAFLIDjLH: <RSCFDn_base> + 1001_H + (10_H × j)
RSCFDnCFDGAFLIDjHL: <RSCFDn_base> + 1002_H + (10_H × j)
RSCFDnCFDGAFLIDjHH: <RSCFDn_base> + 1003_H + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.114 RSCFDnCFDGAFLIDj Register Contents

Bit Position	Bit Name	Function
31	GAFLIDE	IDE Select 0: Standard ID 1: Extended ID
30	GAFLRTR	RTR Select 0: Data frame 1: Remote frame
29	GAFLLB	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received
28 to 0	GAFLID[28:0]	ID Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCFDnCFDGAFLIDj register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLLB Bit

When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.

When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.

GAFLID[28:0] Bits

These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

14.4.5.4 RSCFDnCFDGAFLMj ─ Receive Rule Mask Register (j = 0 to 15)

Access: RSCFDnCFDGAFLMj register can be read/written in 32-bit units.
RSCFDnCFDGAFLMjL, RSCFDnCFDGAFLMjH registers can be read/written in 16-bit units.
RSCFDnCFDGAFLMjLL, RSCFDnCFDGAFLMjLH, RSCFDnCFDGAFLMjHL, RSCFDnCFDGAFLMjHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGAFLMj: <RSCFDn_base> + 1004_H + (10_H × j)
RSCFDnCFDGAFLMjL: <RSCFDn_base> + 1004_H + (10_H × j)
RSCFDnCFDGAFLMjH: <RSCFDn_base> + 1006_H + (10_H × j)
RSCFDnCFDGAFLMjLL: <RSCFDn_base> + 1004_H + (10_H × j)
RSCFDnCFDGAFLMjLH: <RSCFDn_base> + 1005_H + (10_H × j)
RSCFDnCFDGAFLMjHL: <RSCFDn_base> + 1006_H + (10_H × j)
RSCFDnCFDGAFLMjHH: <RSCFDn_base> + 1007_H + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFL IDEM	GAFL RTRM	—	GAFLIDM[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.115 RSCFDnCFDGAFLMj Register Contents

Bit Position	Bit Name	Function
31	GAFLIDEM	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared.
29	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
28 to 0	GAFLIDM[28:0]	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCFDnCFDGAFLMj register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCFDnCFDGAFLIDj register.

When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bit of the receive rule.

14.4.5.5 RSCFDnCFDGAFLP0_j ─ Receive Rule Pointer 0 Register (j = 0 to 15)

Access: RSCFDnCFDGAFLP0_j register can be read/written in 32-bit units.
RSCFDnCFDGAFLP0_jL, RSCFDnCFDGAFLP0_jH registers can be read/written in 16-bit units.
RSCFDnCFDGAFLP0_jLL, RSCFDnCFDGAFLP0_jLH, RSCFDnCFDGAFLP0_jHL, RSCFDnCFDGAFLP0_jHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGAFLP0_j: <RSCFDn_base> + 1008_H + (10_H × j)
RSCFDnCFDGAFLP0_jL: <RSCFDn_base> + 1008_H + (10_H × j),
RSCFDnCFDGAFLP0_jH: <RSCFDn_base> + $100A_H$ + (10_H × j)
RSCFDnCFDGAFLP0_jLL: <RSCFDn_base> + 1008_H + (10_H × j),
RSCFDnCFDGAFLP0_jLH: <RSCFDn_base> + 1009_H + (10_H × j),
RSCFDnCFDGAFLP0_jHL: <RSCFDn_base> + $100A_H$ + (10_H × j),
RSCFDnCFDGAFLP0_jHH: <RSCFDn_base> + $100B_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFL RMV	GAFLRMDP[6:0]							—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 14.116 RSCFDnCFDGAFLP0_j Register Contents

Bit Position	Bit Name	Function
31 to 28	GAFLDLC[3:0]	Receive Rule DLC

b31	b30	b29	b28	Classical CAN Frame	CAN FD Frame
0	0	0	0	No DLC check	
0	0	0	1	1 data byte	
0	0	1	0	2 data bytes	
0	0	1	1	3 data bytes	
0	1	0	0	4 data bytes	
0	1	0	1	5 data bytes	
0	1	1	0	6 data bytes	
0	1	1	1	7 data bytes	
1	0	0	0	8 data bytes	
1	0	0	1	8 data bytes	12 data bytes
1	0	1	0		16 data bytes
1	0	1	1		20 data bytes
1	1	0	0		24 data bytes
1	1	0	1		32 data bytes
1	1	1	0		48 data bytes
1	1	1	1		64 data bytes

27 to 16	GAFLPTR[11:0]	Receive Rule Label Set the 12-bit label information.
15	GAFLRMV	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP[6:0]	Receive Buffer Number Select Set the receive buffer number to store receive messages.
7 to 0	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.

Modify the RSCFDnCFDGAFLP0_j register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to 0000_B disables the DLC check function allowing messages with any data length to pass the DLC check.

GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

GAFLRMV Bit

When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLRMDP[6:0] Bits

These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCFDnCFDRMNB register.

14.4.5.6 RSCFDnCFDGAFLP1_j ─ Receive Rule Pointer 1 Register (j = 0 to 15)

Access: RSCFDnCFDGAFLP1_j register can be read/written in 32-bit units.
RSCFDnCFDGAFLP1_jL, RSCFDnCFDGAFLP1_jH registers can be read/written in 16-bit units.
RSCFDnCFDGAFLP1_jLL, RSCFDnCFDGAFLP1_jLH, RSCFDnCFDGAFLP1_jHL, RSCFDnCFDGAFLP1_jHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGAFLP1_j: <RSCFDn_base> + $100C_H$ + (10_H × j)
RSCFDnCFDGAFLP1_jL: <RSCFDn_base> + $100C_H$ + (10_H × j),
RSCFDnCFDGAFLP1_jH: <RSCFDn_base> + $100E_H$ + (10_H × j)
RSCFDnCFDGAFLP1_jLL: <RSCFDn_base> + $100C_H$ + (10_H × j),
RSCFDnCFDGAFLP1_jLH: <RSCFDn_base> + $100D_H$ + (10_H × j),
RSCFDnCFDGAFLP1_jHL: <RSCFDn_base> + $100E_H$ + (10_H × j),
RSCFDnCFDGAFLP1_jHH: <RSCFDn_base> + $100F_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	GAFLFDP[19:16]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLFDP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.117 RSCFDnCFDGAFLP1_j Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
19 to 8	GAFLFDP[19:8]	Transmit/Receive FIFO Buffer k Select (Bit position –8 = target transmit/receive FIFO buffer number k) 0: Transmit/receive FIFO buffer is not selected. 1: Transmit/receive FIFO buffer is selected.
7 to 0	GAFLFDP[7:0]	Receive FIFO Buffer x Select (Bit position = target receive FIFO buffer number x) 0: Receive FIFO buffer is not selected. 1: Receive FIFO buffer is selected.

Modify the RSCFDnCFDGAFLP1_j register when the AFLDAE bit in the RSCFDnCFDGAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP [19:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCFDnCFDGAFLP0_j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected. Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCFDnCFDCFCCk register are set to 00_B (receive mode) or 10_B (gateway mode) are selectable.

14.4.6 Details of Receive Buffer-Related Registers

14.4.6.1 RSCFDnCFDRMNB ─ Receive Buffer Number Register

Access: RSCFDnCFDRMNB register can be read/written in 32-bit units.
RSCFDnCFDRMNBL, RSCFDnCFDRMNBH registers can be read/written in 16-bit units.
RSCFDnCFDRMNBLL, RSCFDnCFDRMNBLH, RSCFDnCFDRMNBHL, RSCFDnCFDRMNBHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDRMNB: <RSCFDn_base> + $00A4_H$
RSCFDnCFDRMNBL: <RSCFDn_base> + $00A4_H$, RSCFDnCFDRMNBH: <RSCFDn_base> + $00A6_H$
RSCFDnCFDRMNBLL: <RSCFDn_base> + $00A4_H$, RSCFDnCFDRMNBLH: <RSCFDn_base> + $00A5_H$
RSCFDnCFDRMNBHL: <RSCFDn_base> + $00A6_H$, RSCFDnCFDRMNBHH: <RSCFDn_base> + $00A7_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMPLS[1:0]		NRXMB[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.118 RSCFDnCFDRMNB Register Contents

Bit Position	Bit Name	Function
31 to 10	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
9, 8	RMPLS[1:0]	Receive Buffer Payload Storage Size Select b9 b8 0 0: 8 bytes 0 1: 12 bytes 1 0: 16 bytes 1 1: 20 bytes
7 to 0	NRXMB[7:0]	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 64.

Modify the RSCFDnCFDRMNB register only in global reset mode.

RMPLS[1:0] Bits

These bits are used to select the maximum payload size that can be stored in the receive buffer.

NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the RS-CANFD module. The maximum value is 16 × (number of channels).

Setting these bits all to 0 makes receive buffers unavailable.

14.4.6.2 RSCFDnCFDRMNDy ─ Receive Buffer New Data Register (y = 0, 1)

Access: RSCFDnCFDRMNDy register can be read/written in 32-bit units.
RSCFDnCFDRMNDyL, RSCFDnCFDRMNDyH registers can be read/written in 16-bit units.
RSCFDnCFDRMNDyLL, RSCFDnCFDRMNDyLH, RSCFDnCFDRMNDyHL, RSCFDnCFDRMNDyHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDRMNDy: <RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCFDnCFDRMNDyL: <RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCFDnCFDRMNDyH: <RSCFDn_base> + $00AA_H$ + (04_H × y)
RSCFDnCFDRMNDyLL: <RSCFDn_base> + $00A8_H$ + (04_H × y)
RSCFDnCFDRMNDyLH: <RSCFDn_base> + $00A9_H$ + (04_H × y)
RSCFDnCFDRMNDyHL: <RSCFDn_base> + $00AA_H$ + (04_H × y)
RSCFDnCFDRMNDyHH: <RSCFDn_base> + $00AB_H$ + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.119 RSCFDnCFDRMNDy Register Contents

Bit Position	Bit Name	Function
31 to 16	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.
15 to 0	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.

Write 0 to the RSCFDnCFDRMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 63)

Each RMNSq flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.

To clear a flag to 0, the program must write 0 to the flag. Use a store instruction to write “0” to the flag and “1” to other flags. These bits cannot be set to 0 while a message is being stored. The message storing time depends on the storage payload size of the receive buffer. When the RMPLS[1:0] value in the RSCFDnCFDRMNB register is 00_B (8 bytes), the message storing time is 12 pclk clock cycles. When the RMPLS[1:0] value is 11_B (20 bytes), the message storing time is 18 pclk clock cycles. (2 pclk clock cycles per 4 bytes of storage payload size).

These flags are cleared to 0 in global reset mode.

14.4.6.3 RSCFDnCFDRMIDq ─ Receive Buffer ID Register (q = 0 to 63)

Access: RSCFDnCFDRMIDq register can be read only in 32-bit units.
RSCFDnCFDRMIDqL, RSCFDnCFDRMIDqH registers can be read only in 16-bit units.
RSCFDnCFDRMIDqLL, RSCFDnCFDRMIDqLH, RSCFDnCFDRMIDqHL, RSCFDnCFDRMIDqHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRMIDq: <RSCFDn_base> + 2000_H + (20_H × q)
RSCFDnCFDRMIDqL: <RSCFDn_base> + 2000_H + (20_H × q)
RSCFDnCFDRMIDqH: <RSCFDn_base> + 2002_H + (20_H × q)
RSCFDnCFDRMIDqLL: <RSCFDn_base> + 2000_H + (20_H × q)
RSCFDnCFDRMIDqLH: <RSCFDn_base> + 2001_H + (20_H × q)
RSCFDnCFDRMIDqHL: <RSCFDn_base> + 2002_H + (20_H × q)
RSCFDnCFDRMIDqHH: <RSCFDn_base> + 2003_H + (20_H × q)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMR TR	—	RMID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.120 RSCFDnCFDRMIDq Register Contents

Bit Position	Bit Name	Function
31	RMIDE	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	Receive Buffer RTR/RRS • When the received message is a classical CAN frame 0: Data frame 1: Remote frame • When the received message is a CAN FD frame The RRS bit value of the received message can be read.
29	Reserved	When read, the value after a reset is read.
28 to 0	RMID[28:0]	Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RMIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer.

RMRTR Bit

When the received message is a classical CAN frame, this bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer. When the received message is a CAN FD frame, this bit indicates the RRS bit value in the message.

RMID[28:0] Bits

These bits contain the ID of the message stored in the receive buffer.

14.4.6.4 RSCFDnCFDRMPTRq ─ Receive Buffer Pointer Register (q = 0 to 63)

Access: RSCFDnCFDRMPTRq register can be read only in 32-bit units.
RSCFDnCFDRMPTRqL, RSCFDnCFDRMPTRqH registers can be read only in 16-bit units.
RSCFDnCFDRMPTRqLL, RSCFDnCFDRMPTRqLH, RSCFDnCFDRMPTRqHL, RSCFDnCFDRMPTRqHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRMPTRq: <RSCFDn_base> + 2004_H + (20_H × q)
RSCFDnCFDRMPTRqL: <RSCFDn_base> + 2004_H + (20_H × q)
RSCFDnCFDRMPTRqH: <RSCFDn_base> + 2006_H + (20_H × q)
RSCFDnCFDRMPTRqLL: <RSCFDn_base> + 2004_H + (20_H × q)
RSCFDnCFDRMPTRqLH: <RSCFDn_base> + 2005_H + (20_H × q)
RSCFDnCFDRMPTRqHL: <RSCFDn_base> + 2006_H + (20_H × q)
RSCFDnCFDRMPTRqHH: <RSCFDn_base> + 2007_H + (20_H × q)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.121 RSCFDnCFDRMPTRq Register Contents

Bit Position	Bit Name	Function
31 to 28	RMDLC[3:0]	Receive Buffer DLC Data

b31	b30	b29	b28	Classical CAN Frame	CAN FD Frame
0	0	0	0	0 data bytes	
0	0	0	1	1 data byte	
0	0	1	0	2 data bytes	
0	0	1	1	3 data bytes	
0	1	0	0	4 data bytes	
0	1	0	1	5 data bytes	
0	1	1	0	6 data bytes	
0	1	1	1	7 data bytes	
1	0	0	0	8 data bytes	
1	0	0	1	8 data bytes	12 data bytes
1	0	1	0		16 data bytes
1	0	1	1		20 data bytes
1	1	0	0		24 data bytes
1	1	0	1		32 data bytes
1	1	1	0		48 data bytes
1	1	1	1		64 data bytes

27 to 16	RMPTR[11:0]	Receive Buffer Label Data Label information of the received message.
15 to 0	RMTS[15:0]	Receive Buffer Timestamp Data Timestamp value of the received message.

RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer. The number of bytes of the payload to be stored in the receive buffer is determined by the RMPLS[1:0] bits in the RSCFDnCFDRMNB register.

RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

14.4.6.5 RSCFDnCFDRMFDSTSq ─ Receive Buffer CAN FD Status Register (q = 0 to 63)

Access: RSCFDnCFDRMFDSTSq register can be read only in 32-bit units.
RSCFDnCFDRMFDSTSqL, RSCFDnCFDRMFDSTSqH registers can be read only in 16-bit units.
RSCFDnCFDRMFDSTSqLL, RSCFDnCFDRMFDSTSqLH, RSCFDnCFDRMFDSTSqHL,
RSCFDnCFDRMFDSTSqHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRMFDSTSq: <RSCFDn_base> + 2008_H + (20_H × q)
RSCFDnCFDRMFDSTSqL: <RSCFDn_base> + 2008_H + (20_H × q)
RSCFDnCFDRMFDSTSqH: <RSCFDn_base> + $200A_H$ + (20_H × q)
RSCFDnCFDRMFDSTSqLL: <RSCFDn_base> + 2008_H + (20_H × q)
RSCFDnCFDRMFDSTSqLH: <RSCFDn_base> + 2009_H + (20_H × q)
RSCFDnCFDRMFDSTSqHL: <RSCFDn_base> + $200A_H$ + (20_H × q)
RSCFDnCFDRMFDSTSqHH: <RSCFDn_base> + $200B_H$ + (20_H × q)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RMFDF	RMBRS	RMESI
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.122 RSCFDnCFDRMFDSTSq Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, the value after a reset is read.
2	RMFDF	FDF 0: Classical CAN frame 1: CAN FD frame
1	RMBRS	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	RMESI	ESI 0: Error active node 1: Error passive node

RMFDF Bit

This bit indicates the FD format (classical CAN frame or CAN FD frame) of the message stored in the receive buffer.

RMBRS Bit

When the RMFDF bit is 1, this bit indicates the BRS bit value of the message stored in the receive buffer. When the RMFDF bit is 0, this bit is always read as 0.

RMESI Bit

When the RMFDF bit is 1, this bit indicates the ESI bit value of the message stored in the receive buffer. When the RMFDF bit is 0, this bit is always read as 0.

14.4.6.6 RSCFDnCFDRMDFb_q — Receive Buffer Data Field b Register (b = 0 to 4, q = 0 to 63)

Access: RSCFDnCFDRMDFb_q register can be read only in 32-bit units.
RSCFDnCFDRMDFb_qL, RSCFDnCFDRMDFb_qH registers can be read only in 16-bit units.
RSCFDnCFDRMDFb_qLL, RSCFDnCFDRMDFb_qLH, RSCFDnCFDRMDFb_qHL, RSCFDnCFDRMDFb_qHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRMDFb_q: <RSCFDn_base> + $200C_H$ + (04_H × b) + (20_H × q)
RSCFDnCFDRMDFb_qL: <RSCFDn_base> + $200C_H$ + (04_H × b) + (20_H × q)
RSCFDnCFDRMDFb_qH: <RSCFDn_base> + $200E_H$ + (04_H × b) + (20_H × q)
RSCFDnCFDRMDFb_qLL: <RSCFDn_base> + $200C_H$ + (04_H × b) + (20_H × q)
RSCFDnCFDRMDFb_qLH: <RSCFDn_base> + $200D_H$ + (04_H × b) + (20_H × q)
RSCFDnCFDRMDFb_qHL: <RSCFDn_base> + $200E_H$ + (04_H × b) + (20_H × q)
RSCFDnCFDRMDFb_qHH: <RSCFDn_base> + $200F_H$ + (04_H × b) + (20_H × q)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB4 × b + 3 [7:0]								RMDB4 × b + 2 [7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB4 × b + 1 [7:0]								RMDB4 × b + 0 [7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.123 RSCFDnCFDRMDFb_q Register Contents

Bit Position	Bit Name	Function
31 to 24	RMDB4 × b + 3 [7:0]	Receive Buffer Data Byte 4 × b + 3
23 to 16	RMDB4 × b + 2 [7:0]	Receive Buffer Data Byte 4 × b + 2
15 to 8	RMDB4 × b + 1 [7:0]	Receive Buffer Data Byte 4 × b + 1
7 to 0	RMDB4 × b + 0 [7:0]	Receive Buffer Data Byte 4 × b + 0 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCFDnCFDRMPTRq register is smaller than the payload storage size of the receive buffer, data bytes for which no data is set are read as 00_H.

Specify the payload storage size of the receive buffer by the RMPLS[1:0] bits in the RSCFDnCFDRMNB register. Do not read or write the RSCFDnCFDRMDFb_q register corresponding to an area larger than the specified size.

14.4.7 Details of Receive FIFO Buffer-Related Registers

14.4.7.1 RSCFDnCFDRFCCx ─ Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

Access: RSCFDnCFDRFCCx register can be read/written in 32-bit units.
RSCFDnCFDRFCCxL, RSCFDnCFDRFCCxH registers can be read/written in 16-bit units.
RSCFDnCFDRFCCxLL, RSCFDnCFDRFCCxLH, RSCFDnCFDRFCCxHL, RSCFDnCFDRFCCxHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDRFCCx: <RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCFDnCFDRFCCxL: <RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCFDnCFDRFCCxH: <RSCFDn_base> + $00BA_H$ + (04_H × x)
RSCFDnCFDRFCCxLL: <RSCFDn_base> + $00B8_H$ + (04_H × x)
RSCFDnCFDRFCCxLH: <RSCFDn_base> + $00B9_H$ + (04_H × x)
RSCFDnCFDRFCCxHL: <RSCFDn_base> + $00BA_H$ + (04_H × x)
RSCFDnCFDRFCCxHH: <RSCFDn_base> + $00BB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	RFPLS[2:0]			—	—	RFIE	RFE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W

Table 14.124 RSCFDnCFDRFCCx Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
15 to 13	RFIGCV[2:0]	Receive FIFO Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
10 to 8	RFDC[2:0]	Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
6 to 4	RFPLS[2:0]	Receive FIFO Buffer Payload Storage Size Select b6 b5 b4 0 0 0: 8 bytes 0 0 1: 12 bytes 0 1 0: 16 bytes 0 1 1: 20 bytes 1 0 0: 24 bytes 1 0 1: 32 bytes 1 1 0: 48 bytes 1 1 1: 64 bytes
3, 2	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
1	RFIE	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.
0	RFE	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

RFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]).

When the RFDC[2:0] bits are set to 001_B (4 messages), set the RFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B. Modify these bits only in global reset mode.

RFIM Bit

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

RFDC[2:0] Bits

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. When these bits are set to 000_B, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

RFPLS[2:0] Bits

These bits are used to select the maximum payload size that can be stored in the receive FIFO buffer. Modify these bits only in global reset mode.

RFIE Bit

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).

RFE Bit

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCFDnCFDRFSTSx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode.

Set this bit to 1 with another instruction after the settings to all bits in the RSCFDnCFDRFCCx register have been done.

This bit is cleared to 0 in global reset mode.

14.4.7.2 RSCFDnCFDRFSTSx ─ Receive FIFO Buffer Status Register (x = 0 to 7)

Access: RSCFDnCFDRFSTSx register can be read/written in 32-bit units.
RSCFDnCFDRFSTSxL, RSCFDnCFDRFSTSxH registers can be read/written in 16-bit units.
RSCFDnCFDRFSTSxLL, RSCFDnCFDRFSTSxLH, RSCFDnCFDRFSTSxHL, RSCFDnCFDRFSTSxHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDRFSTSx: <RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCFDnCFDRFSTSxL: <RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCFDnCFDRFSTSxH: <RSCFDn_base> + $00DA_H$ + (04_H × x)
RSCFDnCFDRFSTSxLL: <RSCFDn_base> + $00D8_H$ + (04_H × x)
RSCFDnCFDRFSTSxLH: <RSCFDn_base> + $00D9_H$ + (04_H × x)
RSCFDnCFDRFSTSxHL: <RSCFDn_base> + $00DA_H$ + (04_H × x)
RSCFDnCFDRFSTSxHH: <RSCFDn_base> + $00DB_H$ + (04_H × x)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.125 RSCFDnCFDRFSTSx Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
15 to 8	RFMC[7:0]	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
3	RFIF	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).

RFMC[7:0] Flags

This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes 00_H when the RFE bit in the RSCFDnCFDRFCCx register is set to 0.

This flag is 00_H in global reset mode.

RFIF Flag

This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCFDnCFDRFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFMLT Flag

This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.

This flag is cleared to 0 in global reset mode or by writing 0 to this flag.

Modify this bit in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

RFFLL Flag

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCFDnCFDRFCCx register.

If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCFDnCFDRFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFEMP Flag

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCFDnCFDRFCCx register is 0 or in global reset mode.

This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

NOTE

To clear the RFMLT or RFIF flag to 0, use a store instruction to write “0” to the given flag and “1” to the other flags.

14.4.7.3 RSCFDnCFDRFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

Access: RSCFDnCFDRFPCTRx register can only be written in 32-bit units.
RSCFDnCFDRFPCTRxL, RSCFDnCFDRFPCTRxH registers can only be written in 16-bit units.
RSCFDnCFDRFPCTRxLL, RSCFDnCFDRFPCTRxLH, RSCFDnCFDRFPCTRxHL, RSCFDnCFDRFPCTRxHH registers can only be written in 8-bit units.

Address: RSCFDnCFDRFPCTRx: <RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCFDnCFDRFPCTRxL: <RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCFDnCFDRFPCTRxH: <RSCFDn_base> + $00FA_H$ + (04_H × x)
RSCFDnCFDRFPCTRxLL: <RSCFDn_base> + $00F8_H$ + (04_H × x)
RSCFDnCFDRFPCTRxLH: <RSCFDn_base> + $00F9_H$ + (04_H × x)
RSCFDnCFDRFPCTRxHL: <RSCFDn_base> + $00FA_H$ + (04_H × x)
RSCFDnCFDRFPCTRxHH: <RSCFDn_base> + $00FB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 14.126 RSCFDnCFDRFPCTRx Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	RFPC[7:0]	Receive FIFO Pointer Control When these bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer.

When the RFDMAEx value in the RSCFDnCFDCDTCT register is 1 (DMA transfer request enabled), do not write a value to this register.

RFPC[7:0] Bits

When the RFPC[7:0] bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCFDnCFDRFSTSx register is decremented. Read the RSCFDnCFDRFIDx, RSCFDnCFDRFPTRx, RSCFDnCFDRFDSTSx, and RSCFDnCFDRFDFd_x registers to read messages in the receive FIFO buffer, and then write FF_H to the RFPC[7:0] bits.

When writing FF_H to these bits, make sure that the RFE bit in the RSCFDnCFDRFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCFDnCFDRFSTSx register is 0 (the receive FIFO buffer contains unread messages).

14.4.7.4 RSCFDnCFDRFIDx ─ Receive FIFO Buffer Access ID Register (x = 0 to 7)

Access: RSCFDnCFDRFIDx register can be read only in 32-bit units.
RSCFDnCFDRFIDxL, RSCFDnCFDRFIDxH registers can be read only in 16-bit units.
RSCFDnCFDRFIDxLL, RSCFDnCFDRFIDxLH, RSCFDnCFDRFIDxHL, RSCFDnCFDRFIDxHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRFIDx: <RSCFDn_base> + 3000_H + (80_H × x)
RSCFDnCFDRFIDxL: <RSCFDn_base> + 3000_H + (80_H × x)
RSCFDnCFDRFIDxH: <RSCFDn_base> + 3002_H + (80_H × x)
RSCFDnCFDRFIDxLL: <RSCFDn_base> + 3000_H + (80_H × x)
RSCFDnCFDRFIDxLH: <RSCFDn_base> + 3001_H + (80_H × x)
RSCFDnCFDRFIDxHL: <RSCFDn_base> + 3002_H + (80_H × x)
RSCFDnCFDRFIDxHH: <RSCFDn_base> + 3003_H + (80_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.127 RSCFDnCFDRFIDx Register Contents

Bit Position	Bit Name	Function
31	RFIDE	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	Receive FIFO Buffer RTR/RRS • When the received message is a classical CAN frame 0: Data frame 1: Remote frame • When the received message is a CAN FD frame The RRS bit value of the received message can be read.
29	Reserved	When read, the value after a reset is read.
28 to 0	RFID[28:0]	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

RFRTR Bit

When the received message is a classical CAN frame, this bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer. When the received message is a CAN FD frame, this bit indicates the RRS bit value in the message.

RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

14.4.7.5 RSCFDnCFDRFPTRx ─ Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

Access: RSCFDnCFDRFPTRx register can be read only in 32-bit units.
RSCFDnCFDRFPTRxL, RSCFDnCFDRFPTRxH registers can be read only in 16-bit units.
RSCFDnCFDRFPTRxLL, RSCFDnCFDRFPTRxLH, RSCFDnCFDRFPTRxHL, RSCFDnCFDRFPTRxHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRFPTRx: <RSCFDn_base> + 3004_H + (80_H × x)
RSCFDnCFDRFPTRxL: <RSCFDn_base> + 3004_H + (80_H × x)
RSCFDnCFDRFPTRxH: <RSCFDn_base> + 3006_H + (80_H × x)
RSCFDnCFDRFPTRxLL: <RSCFDn_base> + 3004_H + (80_H × x)
RSCFDnCFDRFPTRxLH: <RSCFDn_base> + 3005_H + (80_H × x)
RSCFDnCFDRFPTRxHL: <RSCFDn_base> + 3006_H + (80_H × x)
RSCFDnCFDRFPTRxHH: <RSCFDn_base> + 3007_H + (80_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.128 RSCFDnCFDRFPTRx Register Contents

Bit Position	Bit Name	Function					
31 to 28	RFDLC[3:0]	Receive FIFO Buffer DLC Data					
		b31	b30	b29	b28	Classical CAN Frame	CAN FD Frame
		0	0	0	0	0 data bytes	
		0	0	0	1	1 data byte	
		0	0	1	0	2 data bytes	
		0	0	1	1	3 data bytes	
		0	1	0	0	4 data bytes	
		0	1	0	1	5 data bytes	
		0	1	1	0	6 data bytes	
		0	1	1	1	7 data bytes	
		1	0	0	0	8 data bytes	
		1	0	0	1	8 data bytes	12 data bytes
		1	0	1	0		16 data bytes
		1	0	1	1		20 data bytes
		1	1	0	0		24 data bytes
		1	1	0	1		32 data bytes
		1	1	1	0		48 data bytes
		1	1	1	1		64 data bytes
27 to 16	RFPTR[11:0]	Receive FIFO Buffer Label Data Label information of the received message can be read.					
15 to 0	RFTS[15:0]	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.					

RFDLC[3:0] Bits

These bits contain the data length of the message stored in the receive FIFO buffer.

RFPTR[11:0] Bits

These bits contain the label information of the message stored in the receive FIFO buffer.

RFTS[15:0] Bits

These bits contain the timestamp value of the message stored in the receive FIFO buffer.

14.4.7.6 RSCFDnCFDRFFDSTSx – Receive FIFO CAN FD Status Register (x = 0 to 7)

Access: RSCFDnCFDRFFDSTSx register can be read only in 32-bit units.
RSCFDnCFDRFFDSTSxL, RSCFDnCFDRFFDSTSxH registers can be read only in 16-bit units.
RSCFDnCFDRFFDSTSxLL, RSCFDnCFDRFFDSTSxLH, RSCFDnCFDRFFDSTSxHL, RSCFDnCFDRFFDSTSxHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRFFDSTSx: <RSCFDn_base> + 3008_H + (80_H × x)
RSCFDnCFDRFFDSTSxL: <RSCFDn_base> + 3008_H + (80_H × x)
RSCFDnCFDRFFDSTSxH: <RSCFDn_base> + $300A_H$ + (80_H × x)
RSCFDnCFDRFFDSTSxLL: <RSCFDn_base> + 3008_H + (80_H × x)
RSCFDnCFDRFFDSTSxLH: <RSCFDn_base> + 3009_H + (80_H × x)
RSCFDnCFDRFFDSTSxHL: <RSCFDn_base> + $300A_H$ + (80_H × x)
RSCFDnCFDRFFDSTSxHH: <RSCFDn_base> + $300B_H$ + (80_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RFFDF	RFBRS	RFESI
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.129 RSCFDnCFDRFFDSTSx Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, the value after a reset is read.
2	RFFDF	FDF 0: Classical CAN frame 1: CAN FD frame
1	RFBRS	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	RFESI	ESI 0: Error active node 1: Error passive node

RFFDF Bit

This bit indicates the FD format (classical CAN frame or CAN FD frame) of the message stored in the receive FIFO buffer.

RFBRS Bit

When the RFFDF bit is 1, this bit indicates the BRS bit value of the message stored in the receive FIFO buffer. When the RFFDF bit is 0, this bit is always read as 0.

RFESI Bit

When the RFFDF bit is 1, this bit indicates the ESI bit value of the message stored in the receive FIFO buffer. When the RFFDF bit is 0, this bit is always read as 0.

14.4.7.7 RSCFDnCFDRFDFd_x — Receive FIFO Buffer Access Data Field d Register (d = 0 to 15, x =0 to 7)

Access: RSCFDnCFDRFDFd_x register can be read only in 32-bit units.
RSCFDnCFDRFDFd_xL, RSCFDnCFDRFDFd_xH registers can be read only in 16-bit units.
RSCFDnCFDRFDFd_xLL, RSCFDnCFDRFDFd_xLH, RSCFDnCFDRFDFd_xHL, RSCFDnCFDRFDFd_xHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRFDFd_x: <RSCFDn_base> + $300C_H$ + (04_H × d) + (80_H × x)
RSCFDnCFDRFDFd_xL: <RSCFDn_base> + $300C_H$ + (04_H × d) + (80_H × x)
RSCFDnCFDRFDFd_xH: <RSCFDn_base> + $300E_H$ + (04_H × d) + (80_H × x)
RSCFDnCFDRFDFd_xLL: <RSCFDn_base> + $300C_H$ + (04_H × d) + (80_H × x)
RSCFDnCFDRFDFd_xLH: <RSCFDn_base> + $300D_H$ + (04_H × d) + (80_H × x)
RSCFDnCFDRFDFd_xHL: <RSCFDn_base> + $300E_H$ + (04_H × d) + (80_H × x)
RSCFDnCFDRFDFd_xHH: <RSCFDn_base> + $300F_H$ + (04_H × d) + (80_H × x)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB4 × d + 3 [7:0]								RFDB4 × d + 2 [7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB4 × d + 1 [7:0]								RFDB4 × d + 0 [7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.130 RSCFDnCFDRFDFd_x Register Contents

Bit Position	Bit Name	Function
31 to 24	RFDB4 × d + 3 [7:0]	Receive FIFO Buffer Data Byte 4 × d + 3
		Receive FIFO Buffer Data Byte 4 × d + 2
23 to 16	RFDB4 × d + 2 [7:0]	Receive FIFO Buffer Data Byte 4 × d + 1
		Receive FIFO Buffer Data Byte 4 × d + 0
15 to 8	RFDB4 × d + 1 [7:0]	Data for a message stored in the receive FIFO buffer can be read.
7 to 0	RFDB4 × d + 0 [7:0]	

When the RFDLC[3:0] value in the RSCFDnCFDRFPTRx register is smaller than the payload storage size of the receive FIFO buffer, data bytes for which no data is set are read as 00_H.

Specify the payload storage size of the receive FIFO buffer by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register. Do not read or write the RSCFDnCFDRFDFd_x register corresponding to an area larger than the specified size.

14.4.8 Transmit/Receive FIFO Buffer Related Registers

14.4.8.1 RSCFDnCFDCFCCk — Transmit/receive FIFO Buffer Configuration and Control Register k (k = 0 to 11)

Access: RSCFDnCFDCFCCk register can be read/written in 32-bit units.
RSCFDnCFDCFCCkL, RSCFDnCFDCFCCkH registers can be read/written in 16-bit units.
RSCFDnCFDCFCCkLL, RSCFDnCFDCFCCkLH, RSCFDnCFDCFCCkHL, RSCFDnCFDCFCCkHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCFCCk: <RSCFDn_base> + 0118_H + (04_H × k)
RSCFDnCFDCFCCkL: <RSCFDn_base> + 0118_H + (04_H × k)
RSCFDnCFDCFCCkH: <RSCFDn_base> + $011A_H$ + (04_H × k)
RSCFDnCFDCFCCkLL: <RSCFDn_base> + 0118_H + (04_H × k)
RSCFDnCFDCFCCkLH: <RSCFDn_base> + 0119_H + (04_H × k)
RSCFDnCFDCFCCkHL: <RSCFDn_base> + $011A_H$ + (04_H × k)
RSCFDnCFDCFCCkHH: <RSCFDn_base> + $011B_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFTML[3:0]				CFITR	CFITSS	CFM[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	—	CFDC[2:0]			—	CFPLS[2:0]			—	CFTXIE	CFRXIE	CFE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 14.131 RSCFDnCFDCFCCk Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 24	CFITT[7:0]	Set a message transmission interval. Set Value: 00_H to FF_H
23 to 20	CFTML[3:0]	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	Transmit/Receive FIFO Interval Timer Resolution 0: Clock dividing pclk by (ITRCP[15:0] bits) 1: Clock dividing pclk by (ITRCP[15:0] bits × 10)
18	CFITSS	Transmit/Receive FIFO Interval Timer Clock Source Select 0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the nominal bit time clock for the channel to which the FIFO is linked.
17, 16	CFM[1:0]	Transmit/Receive FIFO Mode Select b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited
15 to 13	CFIGCV[2:0]	Transmit/Receive FIFO Receive Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	CFIM	Transmit/Receive FIFO Interrupt Source Select 0: Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. 1: Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted.
11	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
10 to 8	CFDC[2:0]	Transmit/Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages

Table 14.131 RSCFDnCFDCFCCk Register Contents (2/2)

Bit Position	Bit Name	Function
7	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
6 to 4	CFPLS[2:0]	Transmit/Receive FIFO Buffer Payload Storage Size Select b6 b5 b4 0 0 0: 8 bytes 0 0 1: 12 bytes 0 1 0: 16 bytes 0 1 1: 20 bytes 1 0 0: 24 bytes 1 0 1: 32 bytes 1 1 0: 48 bytes 1 1 1: 64 bytes
3	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
2	CFTXIE	Transmit/Receive FIFO Transmit Interrupt Enable 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.
1	CFRXIE	Transmit/Receive FIFO Receive Interrupt Enable 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.
0	CFE	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

CFITT[7:0] Bits

These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode).

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

CFTML[3:0] Bits

These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode). There are three transmit/receive FIFO buffers per channel, so channel number m of FIFO buffer k is calculated as m = k/3 (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be ((16 × m) + CFTML[3:0]).

See **Table 14.88** and **Table 14.89**, as for the relationship between transmit/receive FIFO buffer k and transmit buffer p.

Setting the CFDC[2:0] bits to 001_B or more enables the setting of the CFTML[3:0] bits.

Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFITR Bit

This bit is enabled when the CFITSS bit is 0.

When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCFDnCFDGCFG register.

When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCFDnCFDGCFG register × 10).

Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFITSS Bit

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.

When this bit is 1, the nominal bit time clock of the channel to which the FIFO is linked is the count source of the interval timer. Use this count source only for the channel does not handle the CAN FD frames.

Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFM[1:0] Bits

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

CFIGCV[2:0] Bits

These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to 00_B (receive mode) or 10_B (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).

When the CFDC[2:0] bits are set to 001_B (4 messages), set the CFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B.

Modify these bits only in global reset mode.

CFIM Bit

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

CFDC[2:0] Bits

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to 000_B, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFPLS[2:0] Bits

These bits are used to select the maximum payload size that can be stored in the transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFTXIE Bit

When this bit is set to 1 and the CFTXIF flag in the RSCFDnCFDCFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFRXIE Bit

When this bit is set to 1 and the CFRXIF flag in the RSCFDnCFDCFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.

Modify this bit with the CFE bit set to 0.

CFE Bit

Setting this bit to 1 makes transmit/receive FIFO buffers available.

When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately.

This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode

Modify this bit in the following mode.

- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

After all other bits in the RSCFDnCFDCFCCk register have been set, set this bit to 1 by using another instruction.

14.4.8.2 RSCFDnCFDCFSTSk ─ Transmit/receive FIFO Buffer Status Register (k = 0 to 11)

Access: RSCFDnCFDCFSTSk register can be read/written in 32-bit units.
RSCFDnCFDCFSTSkL, RSCFDnCFDCFSTSkH registers can be read/written in 16-bit units.
RSCFDnCFDCFSTSkLL, RSCFDnCFDCFSTSkLH, RSCFDnCFDCFSTSkHL, RSCFDnCFDCFSTSkHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCFSTSk: <RSCFDn_base> + 0178_H + (04_H × k)
RSCFDnCFDCFSTSkL: <RSCFDn_base> + 0178_H + (04_H × k),
RSCFDnCFDCFSTSkH: <RSCFDn_base> + $017A_H$ + (04_H × k)
RSCFDnCFDCFSTSkLL: <RSCFDn_base> + 0178_H + (04_H × k),
RSCFDnCFDCFSTSkLH: <RSCFDn_base> + 0179_H + (04_H × k),
RSCFDnCFDCFSTSkHL: <RSCFDn_base> + $017A_H$ + (04_H × k),
RSCFDnCFDCFSTSkHH: <RSCFDn_base> + $017B_H$ + (04_H × k)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.132 RSCFDnCFDCFSTSk Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
15 to 8	CFMC[7:0]	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer.
7 to 5	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
4	CFTXIF	Transmit/Receive FIFO Transmit Interrupt Request Flag 0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.
3	CFRXIF	Transmit/Receive FIFO Receive Interrupt Request Flag 0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.
2	CFMLT	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFFLL	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCFDnCFDCFCCk register.

- When CFM[1:0] value is 01_B (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00_B (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10_B (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00_B: In global reset mode
- When CFM[1:0] value is 01_B or 10_B: In channel reset mode
- When the CFE bit in the RSCFDnCFDCFCCk register is cleared to 0.

CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01_B or 10_B, and the factor selected by the CFIM bit in the RSCFDnCFDCFCCk register occurs

The CFTXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFTXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B or 10_B, and the factor selected by the CFIM bit in the RSCFDnCFDCFCCk register occurs

The CFRXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFRXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFMLT flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

CFFLL Flag

The CFFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCFDnCFDCFCCk register.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits.
- When the CFE bit in the RSCFDnCFDCFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmit abort
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: All messages have been transmitted, or in channel reset mode
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B or 10_B: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01_B: A value of FF_H has been written to the RSCFDnCFDCFPCTRk register after data was written to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers.

NOTE

To clear CFTXIF, CFRXIF, or CFMLT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

14.4.8.3 RSCFDnCFDCFPCTRk ─ Transmit/receive FIFO Buffer Pointer Control Register (k = 0 to 11)

Access: RSCFDnCFDCFPCTRk register can only be written in 32-bit units.
RSCFDnCFDCFPCTRkL, RSCFDnCFDCFPCTRkH registers can only be written in 16-bit units.
RSCFDnCFDCFPCTRkLL, RSCFDnCFDCFPCTRkLH, RSCFDnCFDCFPCTRkHL, RSCFDnCFDCFPCTRkHH registers can only be written in 8-bit units.

Address: RSCFDnCFDCFPCTRk: <RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCFDnCFDCFPCTRkL: <RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCFDnCFDCFPCTRkH: <RSCFDn_base> + $01DA_H$ + (04_H × k)
RSCFDnCFDCFPCTRkLL: <RSCFDn_base> + $01D8_H$ + (04_H × k)
RSCFDnCFDCFPCTRkLH: <RSCFDn_base> + $01D9_H$ + (04_H × k)
RSCFDnCFDCFPCTRkHL: <RSCFDn_base> + $01DA_H$ + (04_H × k)
RSCFDnCFDCFPCTRkHH: <RSCFDn_base> + $01DB_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 14.133 RSCFDnCFDCFPCTRk Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	CFPC[7:0]	Transmit/Receive FIFO Pointer Control • Receive mode: Writing FF_H to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. • Transmit mode: Writing FF_H to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. • Gateway mode: Setting prohibited

When the corresponding transmit/receive FIFO buffer is the first transmit/receive FIFO buffer (k = 3 × m) allocated to channel m and when the CFDMAEm bit in the RSCFDnCFDCDTCT register is 1 (DMA transfer request enabled), do not write a value to this register.

CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 00_B):
  Writing FF_H to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCFDnCFDCFSTSk register is decremented. Read the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers to read messages from the transmit/receive FIFO buffer, and then write FF_H to the CFPC[7:0] bits.
  When writing FF_H to these bits, make sure that the CFE bit in the RSCFDnCFDCFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCFDnCFDCFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).
- Transmit mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B):
  Writing FF_H to the CFPC[7:0] bits stores the data written to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCFDnCFDCFIDk, RSCFDnCFDCFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDnCFDCFDFd_k registers before writing FF_H to the CFPC[7:0] bits.
  When writing FF_H to these bits, make sure that the CFE bit in the RSCFDnCFDCFCCk register is set to 1 and the CFFLL flag in the RSCFDnCFDCFSTSk register is 0 (the transmit/receive FIFO buffer is not full).
- Gateway mode (CFM[1:0] value in the RSCFDnCFDCFCCk register is 10_B): Setting prohibited

14.4.8.4 RSCFDnCFDCFIDk – Transmit/Receive FIFO Buffer Access ID Register (k = 0 to 11)

Access: RSCFDnCFDCFIDk register can be read/written in 32-bit units.
RSCFDnCFDCFIDkL, RSCFDnCFDCFIDkH registers can be read/written in 16-bit units.
RSCFDnCFDCFIDkLL, RSCFDnCFDCFIDkLH, RSCFDnCFDCFIDkHL, RSCFDnCFDCFIDkHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCFIDk: <RSCFDn_base> + 3400_H + (80_H × k)
RSCFDnCFDCFIDkL: <RSCFDn_base> + 3400_H + (80_H × k)
RSCFDnCFDCFIDkH: <RSCFDn_base> + 3402_H + (80_H × k)
RSCFDnCFDCFIDkLL: <RSCFDn_base> + 3400_H + (80_H × k)
RSCFDnCFDCFIDkLH: <RSCFDn_base> + 3401_H + (80_H × k)
RSCFDnCFDCFIDkHL: <RSCFDn_base> + 3402_H + (80_H × k)
RSCFDnCFDCFIDkHH: <RSCFDn_base> + 3403_H + (80_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.134 RSCFDnCFDCFIDk Register Contents

Bit Position	Bit Name	Function
31	CFIDE	Transmit/Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	CFRTR	Transmit/Receive FIFO Buffer RTR/RRS • When the CFM[1:0] value is 01_B (transmit mode) - When the transmit message is a classical CAN frame 0: Data frame 1: Remote frame - When the transmit message is a CAN FD frame Write 0 to this bit. • When the CFM[1:0] value is 00_B (receive mode) - When the received message is a classical CAN frame 0: Data frame 1: Remote frame - When the received message is a CAN FD frame The RRS bit value of the received message can be read.
29	THLEN	Transmit History Data Store Enable This bit is valid only when the CFM[1:0] value is 01_B (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	CFID[28:0]	Transmit/Receive FIFO Buffer ID Data • When CFM[1:0] value is 01_B (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11. • When CFM[1:0] value is 00_B (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits 10 to 0. Bits 28 to 11 are read as 0.

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This RSCFDnCFDCFIDk register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, these bits are used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

CFRTR Bit

If the the received message is a classical CAN frame, this bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. If the received message is a CAN FD frame, this bit indicates the RRS bit value of the received message.

When the CFM[1:0] value is 01_B, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer. When the CFFDF bit in the RSCFDnCFDCFFDCSTSk register is 1 (CAN FD frame), set this bit to 0.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp) of transmit messages is stored in the transmit history buffer after transmission is completed.

This bit is enabled when the CFM[1:0] value is 01_B (transmit mode).

CFID[28:0] Bits

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B.

When the CFM[1:0] value is 01_B, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

14.4.8.5 RSCFDnCFDCFPTRk – Transmit/Receive FIFO Buffer Access Pointer Register (k = 0 to 11)

Access: RSCFDnCFDCFPTRk register can be read/written in 32-bit units.
RSCFDnCFDCFPTRkL, RSCFDnCFDCFPTRkH registers can be read/written in 16-bit units.
RSCFDnCFDCFPTRkLL, RSCFDnCFDCFPTRkLH, RSCFDnCFDCFPTRkHL, RSCFDnCFDCFPTRkHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCFPTRk: <RSCFDn_base> + 3404_H + (80_H × k)
RSCFDnCFDCFPTRkL: <RSCFDn_base> + 3404_H + (80_H × k)
RSCFDnCFDCFPTRkH: <RSCFDn_base> + 3406_H + (80_H × k)
RSCFDnCFDCFPTRkLL: <RSCFDn_base> + 3404_H + (80_H × k)
RSCFDnCFDCFPTRkLH: <RSCFDn_base> + 3405_H + (80_H × k)
RSCFDnCFDCFPTRkHL: <RSCFDn_base> + 3406_H + (80_H × k)
RSCFDnCFDCFPTRkHH: <RSCFDn_base> + 3407_H + (80_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.135 RSCFDnCFDCFPTRk Register Contents (1/2)

Bit Position	Bit Name	Function					
31 to 28	CFDLC[3:0]	Transmit/Receive FIFO Buffer DLC Data					
		b31	b30	b29	b28	Classical CAN Frame	CAN FD Frame
		0	0	0	0	0 data bytes	
		0	0	0	1	1 data byte	
		0	0	1	0	2 data bytes	
		0	0	1	1	3 data bytes	
		0	1	0	0	4 data bytes	
		0	1	0	1	5 data bytes	
		0	1	1	0	6 data bytes	
		0	1	1	1	7 data bytes	
		1	0	0	0	8 data bytes	
		1	0	0	1	8 data bytes	12 data bytes
		1	0	1	0		16 data bytes
		1	0	1	1		20 data bytes
		1	1	0	0		24 data bytes
		1	1	0	1		32 data bytes
		1	1	1	0		48 data bytes
		1	1	1	1		64 data bytes

Table 14.135 RSCFDnCFDCFPTRk Register Contents (2/2)

Bit Position	Bit Name	Function
27 to 16	CFPTR[11:0]	Transmit/Receive FIFO Buffer Label Data • When CFM[1:0] value is 01_B (transmit mode): Set the label information to be stored in the transmit history buffer. Only bits CFPTR[7:0] are valid. • When CFM[1:0] value is 00_B (receive mode): The label information of the received message can be read.
15 to 0	CFTS[15:0]	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is 00_B (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFDLC[3:0] Bits

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B.

When the CFM[1:0] value is 01_B, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer.

When the CFDLC[3:0] bits are set to 1001_B or more while the CFFDF bit in the RSCFDnCFDCFFDCSTSk register is 0 (classical CAN frame), 8-byte data is transmitted actually. When the CFFDF bit is 1 (CAN FD frame), the settable value range varies depending on the settings of the TMME bit in the RSCFDnCFDCmFDCFG register and the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

- When TMME bit = 0 (transmit buffer merge mode disabled):
  A value of 0000_B to 1111_B is settable. If the specified data length exceeds the payload storage size specified by the CFPLS[2:0] bits, excessive payloads are padded by CC_H.
- When TMME bit = 1 (transmit buffer merge mode enabled):
  Set the data length within the payload storage size specified by the CFPLS[2:0] bits.

CFPTR[11:0] Bits

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

CFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer.

These bits are valid when the CFM[1:0] value is 00_B.

14.4.8.6 RSCFDnCFDCFFDCSTSk ─ Transmit/Receive FIFO CAN FD Configuration/Status Register (k = 0 to 11)

Access: RSCFDnCFDCFFDCSTSk register can be read/written in 32-bit units.
RSCFDnCFDCFFDCSTSkL, RSCFDnCFDCFFDCSTSkH registers can be read/written in 16-bit units.
RSCFDnCFDCFFDCSTSkLL, RSCFDnCFDCFFDCSTSkLH, RSCFDnCFDCFFDCSTSkHL,
RSCFDnCFDCFFDCSTSkHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCFFDCSTSk: <RSCFDn_base> + 3408_H + (80_H × k)
RSCFDnCFDCFFDCSTSkL: <RSCFDn_base> + 3408_H + (80_H × k)
RSCFDnCFDCFFDCSTSkH: <RSCFDn_base> + $340A_H$ + (80_H × k)
RSCFDnCFDCFFDCSTSkLL: <RSCFDn_base> + 3408_H + (80_H × k)
RSCFDnCFDCFFDCSTSkLH: <RSCFDn_base> + 3409_H + (80_H × k)
RSCFDnCFDCFFDCSTSkHL: <RSCFDn_base> + $340A_H$ + (80_H × k)
RSCFDnCFDCFFDCSTSkHH: <RSCFDn_base> + $340B_H$ + (80_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CFFDF	CFBRS	CFESI
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 14.136 RSCFDnCFDCFFDCSTSk Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
2	CFFDF	FDF 0: Classical CAN frame 1: CAN FD frame
1	CFBRS	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	CFESI	ESI 0: Error active node 1: Error passive node

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). Do not read or write this register when the CFM[1:0] value is 10_B (gateway mode).

CFFDF Bit

When the CFM[1:0] value is 00_B, this bit indicates the FD format (classical CAN frame or CAN FD frame) of the message stored in the transmit/receive FIFO buffer. When the CFM[1:0] value is 01_B, this bit is used to set the FD format of the message to be transmitted from the transmit/receive FIFO buffer.

CFBRS Bit

When the CFM[1:0] value is 00_B, if the CFFDF bit is 1, this bit indicates the BRS bit value of the message stored in the transmit/receive FIFO buffer. If the CFFDF bit is 0, this bit is always read as 0.

When the CFM[1:0] value is 01_B, if the CFFDF bit is 1, this bit is used to set the BRS bit value of the message to be transmitted from the transmit/receive FIFO buffer. If the CFFDF bit is 0, write 0 to this bit.

CFESI Bit

When the CFM[1:0] value is 00_B, if the CFFDF bit is 1, this bit indicates the ESI bit value of the message stored in the transmit/receive FIFO buffer. If the CFFDF bit is 0, this bit is always read as 0.

When the CFM[1:0] value is 01_B, if the CFFDF bit is 1, this bit is used to set the ESI bit value of the message to be transmitted from the transmit/receive FIFO buffer. The set value is transmitted when the ESIC bit in the RSCFDnCFDCmFDCFG register is 1 and the channel is in the error active state. When the channel is in the error passive state, the ESI bit value that shows an error passive node is transmitted regardless of this bit value. When the CFFDF bit is 0, write 0 to this bit.

14.4.8.7 RSCFDnCFDCFDFd_k — Transmit/Receive FIFO Buffer Access Data Field d Register (d = 0 to 15, k = 0 to 11)

Access: RSCFDnCFDCFDFd_k register can be read/written in 32-bit units.
RSCFDnCFDCFDFd_kL, RSCFDnCFDCFDFd_kH registers can be read/written in 16-bit units.
RSCFDnCFDCFDFd_kLL, RSCFDnCFDCFDFd_kLH, RSCFDnCFDCFDFd_kHL, RSCFDnCFDCFDFd_kHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCFDFd_k: <RSCFDn_base> + $340C_H$ + (04_H × d) + (80_H × k)
RSCFDnCFDCFDFd_kL: <RSCFDn_base> + $340C_H$ + (04_H × d) + (80_H × k)
RSCFDnCFDCFDFd_kH: <RSCFDn_base> + $340E_H$ + (04_H × d) + (80_H × k)
RSCFDnCFDCFDFd_kLL: <RSCFDn_base> + $340C_H$ + (04_H × d) + (80_H × k)
RSCFDnCFDCFDFd_kLH: <RSCFDn_base> + $340D_H$ + (04_H × d) + (80_H × k)
RSCFDnCFDCFDFd_kHL: <RSCFDn_base> + $340E_H$ + (04_H × d) + (80_H × k)
RSCFDnCFDCFDFd_kHH: <RSCFDn_base> + $340F_H$ + (04_H × d) + (80_H × k)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB4 × d + 3 [7:0]								CFDB4 × d + 2 [7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB4 × d + 1 [7:0]								CFDB4 × d + 0 [7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.137 RSCFDnCFDCFDFd_k Register Contents

Bit Position	Bit Name	Function
31 to 24	CFDB4 × d + 3 [7:0]	Transmit/Receive FIFO Buffer Data Byte 4 × d + 3 Transmit/Receive FIFO Buffer Data Byte 4 × d + 2 Transmit/Receive FIFO Buffer Data Byte 4 × d + 1 Transmit/Receive FIFO Buffer Data Byte 4 × d + 0 • When CFM[1:0] value is 01_B (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00_B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.
23 to 16	CFDB4 × d + 2 [7:0]	
15 to 8	CFDB4 × d + 1 [7:0]	
7 to 0	CFDB4 × d + 0 [7:0]	

This register is writable only when the CFM[1:0] value in the RSCFDnCFDCFCCk register is 01_B (transmit mode).

This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCFDnCFDCFPTRk register is smaller than the payload storage size of the transmit/receive FIFO buffer, data bytes for which no data is set are read as 00_H.

Specify the payload storage size of the transmit/receive FIFO buffer by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register. Do not read or write the RSCFDnCFDCFDFd_k register corresponding to an area larger than the specified size.

This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

14.4.9 Details of FIFO Status-Related Registers

14.4.9.1 RSCFDnCFDFESTS – FIFO Empty Status Register

Access: RSCFDnCFDFESTS register can be read only in 32-bit units.
RSCFDnCFDFESTSL, RSCFDnCFDFESTSH registers can be read only in 16-bit units.
RSCFDnCFDFESTSLL, RSCFDnCFDFESTSLH, RSCFDnCFDFESTSHL, RSCFDnCFDFESTSHH registers can be read only in 8-bit units.

Address: RSCFDnCFDFESTS: <RSCFDn_base> + 0238_H
RSCFDnCFDFESTSL: <RSCFDn_base> + 0238_H, RSCFDnCFDFESTSH: <RSCFDn_base> + $023A_H$
RSCFDnCFDFESTSLL: <RSCFDn_base> + 0238_H, RSCFDnCFDFESTSLH: <RSCFDn_base> + 0239_H
RSCFDnCFDFESTSHL: <RSCFDn_base> + $023A_H$, RSCFDnCFDFESTSHH: <RSCFDn_base> + $023B_H$

Value after reset: $03FF\ FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11 EMP	CF10 EMP	CF9 EMP	CF8 EMP
Value after reset	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7 EMP	CF6 EMP	CF5 EMP	CF4 EMP	CF3 EMP	CF2 EMP	CF1 EMP	CF0 EMP	RF7 EMP	RF6 EMP	RF5 EMP	RF4 EMP	RF3 EMP	RF2 EMP	RF1 EMP	RF0 EMP
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.138 RSCFDnCFDFESTS Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after a reset is read.
19	CF11EMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 11)
18	CF10EMP	
17	CF9EMP	
16	CF8EMP	
15	CF7EMP	
14	CF6EMP	
13	CF5EMP	
12	CF4EMP	
11	CF3EMP	
10	CF2EMP	
9	CF1EMP	
8	CF0EMP	
7	RF7EMP	Receive FIFO Buffer Empty Status Flag 0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message (buffer empty). (x = 0 to 7)
6	RF6EMP	
5	RF5EMP	
4	RF4EMP	
3	RF3EMP	
2	RF2EMP	
1	RF1EMP	
0	RF0EMP	

The RSCFDnCFDFESTS register is set to 03FF $FFFF_H$ in global reset mode.

CFkEMP Flag (k = 0 to 11)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCFDnCFDCFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCFDnCFDRFSTSx register is set to 1 (the receive FIFO buffer contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

14.4.9.2 RSCFDnCFDFFSTS ─ FIFO Full Status Register

Access: RSCFDnCFDFFSTS register can be read only in 32-bit units.
RSCFDnCFDFFSTSL, RSCFDnCFDFFSTSH registers can be read only in 16-bit units.
RSCFDnCFDFFSTSLL, RSCFDnCFDFFSTSLH, RSCFDnCFDFFSTSHL, RSCFDnCFDFFSTSHH registers can be read only in 8-bit units.

Address: RSCFDnCFDFFSTS: <RSCFDn_base> + $023C_H$
RSCFDnCFDFFSTSL: <RSCFDn_base> + $023C_H$, RSCFDnCFDFFSTSH: <RSCFDn_base> + $023E_H$
RSCFDnCFDFFSTSLL: <RSCFDn_base> + $023C_H$, RSCFDnCFDFFSTSLH: <RSCFDn_base> + $023D_H$
RSCFDnCFDFFSTSHL: <RSCFDn_base> + $023E_H$, RSCFDnCFDFFSTSHH: <RSCFDn_base> + $023F_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11 FLL	CF10 FLL	CF9 FLL	CF8 FLL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7 FLL	CF6 FLL	CF5 FLL	CF4 FLL	CF3 FLL	CF2 FLL	CF1 FLL	CF0 FLL	RF7 FLL	RF6 FLL	RF5 FLL	RF4 FLL	RF3 FLL	RF2 FLL	RF1 FLL	RF0 FLL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.139 RSCFDnCFDFFSTS Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after a reset is read.
19	CF11FLL	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full. (k = 0 to 11)
18	CF10FLL	
17	CF9FLL	
16	CF8FLL	
15	CF7FLL	
14	CF6FLL	
13	CF5FLL	
12	CF4FLL	
11	CF3FLL	
10	CF2FLL	
9	CF1FLL	
8	CF0FLL	
7	RF7FLL	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7)
6	RF6FLL	
5	RF5FLL	
4	RF4FLL	
3	RF3FLL	
2	RF2FLL	
1	RF1FLL	
0	RF0FLL	

The RSCFDnCFDFFSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkFLL Flag (k = 0 to 11)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCFDnCFDCFSTSk register is set to 1 (the transmit/receive FIFO buffer is full).

When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0.

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCFDnCFDRFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

14.4.9.3 RSCFDnCFDFMSTS ─ FIFO Message Lost Status Register

Access: RSCFDnCFDFMSTS register can be read only in 32-bit units.
RSCFDnCFDFMSTSL, RSCFDnCFDFMSTSH registers can be read only in 16-bit units.
RSCFDnCFDFMSTSLL, RSCFDnCFDFMSTSLH, RSCFDnCFDFMSTSHL, RSCFDnCFDFMSTSHH registers can be read only in 8-bit units.

Address: RSCFDnCFDFMSTS: <RSCFDn_base> + 0240_H
RSCFDnCFDFMSTSL: <RSCFDn_base> + 0240_H, RSCFDnCFDFMSTSH: <RSCFDn_base> + 0242_H
RSCFDnCFDFMSTSLL: <RSCFDn_base> + 0240_H, RSCFDnCFDFMSTSLH: <RSCFDn_base> + 0241_H
RSCFDnCFDFMSTSHL: <RSCFDn_base> + 0242_H, RSCFDnCFDFMSTSHH: <RSCFDn_base> + 0243_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11MLT	CF10MLT	CF9MLT	CF8MLT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7MLT	CF6MLT	CF5MLT	CF4MLT	CF3MLT	CF2MLT	CF1MLT	CF0MLT	RF7MLT	RF6MLT	RF5MLT	RF4MLT	RF3MLT	RF2MLT	RF1MLT	RF0MLT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.140 RSCFDnCFDFMSTS Register Contents

Bit Position	Bit Name	Function
31 to 20	Reserved	When read, the value after a reset is read.
19	CF11MLT	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 11)
18	CF10MLT	
17	CF9MLT	
16	CF8MLT	
15	CF7MLT	
14	CF6MLT	
13	CF5MLT	
12	CF4MLT	
11	CF3MLT	
10	CF2MLT	
9	CF1MLT	
8	CF0MLT	
7	RF7MLT	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	
5	RF5MLT	
4	RF4MLT	
3	RF3MLT	
2	RF2MLT	
1	RF1MLT	
0	RF0MLT	

The RSCFDnCFDFMSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkMLT Flag (k = 0 to 11)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO message is lost).

When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCFDnCFDRFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

14.4.9.4 RSCFDnCFDRFISTS ─ Receive FIFO Buffer Interrupt Flag Status Register

Access: RSCFDnCFDRFISTS register can be read only in 32-bit units.
RSCFDnCFDRFISTSL, RSCFDnCFDRFISTSH registers can be read only in 16-bit units.
RSCFDnCFDRFISTSLL, RSCFDnCFDRFISTSLH, RSCFDnCFDRFISTSHL, RSCFDnCFDRFISTSHH registers can be read only in 8-bit units.

Address: RSCFDnCFDRFISTS: <RSCFDn_base> + 0244_H
RSCFDnCFDRFISTSL: <RSCFDn_base> + 0244_H, RSCFDnCFDRFISTSH: <RSCFDn_base> + 0246_H
RSCFDnCFDRFISTSLL: <RSCFDn_base> + 0244_H, RSCFDnCFDRFISTSLH: <RSCFDn_base> + 0245_H
RSCFDnCFDRFISTSHL: <RSCFDn_base> + 0246_H, RSCFDnCFDRFISTSHH: <RSCFDn_base> + 0247_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.141 RSCFDnCFDRFISTS Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When read, the value after reset is returned.
7	RF7IF	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	
5	RF5IF	
4	RF4IF	
3	RF3IF	
2	RF2IF	
1	RF1IF	
0	RF0IF	

The RSCFDnCFDRFISTS register is cleared to $0000\ 0000_H$ in global reset mode.

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCFDnCFDRFSTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

14.4.9.5 RSCFDnCFDCFRISTS – Transmit/Receive FIFO Buffer Receive Interrupt Flag Status Register

Access: RSCFDnCFDCFRISTS register can be read only in 32-bit units.
RSCFDnCFDCFRISTSL, RSCFDnCFDCFRISTSH registers can be read only in 16-bit units.
RSCFDnCFDCFRISTSLL, RSCFDnCFDCFRISTSLH, RSCFDnCFDCFRISTSHL, RSCFDnCFDCFRISTSHH registers can be read only in 8-bit units.

Address: RSCFDnCFDCFRISTS: <RSCFDn_base> + 0248_H
RSCFDnCFDCFRISTSL: <RSCFDn_base> + 0248_H, RSCFDnCFDCFRISTSH: <RSCFDn_base> + $024A_H$
RSCFDnCFDCFRISTSLL: <RSCFDn_base> + 0248_H, RSCFDnCFDCFRISTSLH: <RSCFDn_base> + 0249_H
RSCFDnCFDCFRISTSHL: <RSCFDn_base> + $024A_H$, RSCFDnCFDCFRISTSHH: <RSCFDn_base> + $024B_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CF11 RXIF	CF10 RXIF	CF9 RXIF	CF8 RXIF	CF7 RXIF	CF6 RXIF	CF5 RXIF	CF4 RXIF	CF3 RXIF	CF2 RXIF	CF1 RXIF	CF0 RXIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.142 RSCFDnCFDCFRISTS Register Contents

Bit Position	Bit Name	Function
31 to 12	Reserved	When read, the value after reset is returned.
11	CF11RXIF	Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k receive interrupt request is present. 1: A transmit/receive FIFO buffer k receive interrupt request is present. (k = 0 to 11)
10	CF10RXIF	
9	CF9RXIF	
8	CF8RXIF	
7	CF7RXIF	
6	CF6RXIF	
5	CF5RXIF	
4	CF4RXIF	
3	CF3RXIF	
2	CF2RXIF	
1	CF1RXIF	
0	CF0RXIF	

The RSCFDnCFDCFRISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkRXIF Flag (k = 0 to 11)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0.

14.4.9.6 RSCFDnCFDCFTISTS – Transmit/Receive FIFO Buffer Transmit Interrupt Flag Status Register

Access: RSCFDnCFDCFTISTS register can be read only in 32-bit units.
RSCFDnCFDCFTISTSL, RSCFDnCFDCFTISTSH registers can be read only in 16-bit units.
RSCFDnCFDCFTISTSLL, RSCFDnCFDCFTISTSLH, RSCFDnCFDCFTISTSHL, RSCFDnCFDCFTISTSHH registers can be read only in 8-bit units.

Address: RSCFDnCFDCFTISTS: <RSCFDn_base> + $024C_H$
RSCFDnCFDCFTISTSL: <RSCFDn_base> + $024C_H$, RSCFDnCFDCFTISTSH: <RSCFDn_base> + $024E_H$
RSCFDnCFDCFTISTSLL: <RSCFDn_base> + $024C_H$, RSCFDnCFDCFTISTSLH: <RSCFDn_base> + $024D_H$
RSCFDnCFDCFTISTSHL: <RSCFDn_base> + $024E_H$, RSCFDnCFDCFTISTSHH: <RSCFDn_base> + $024F_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CF11 TXIF	CF10 TXIF	CF9 TXIF	CF8 TXIF	CF7 TXIF	CF6 TXIF	CF5 TXIF	CF4 TXIF	CF3 TXIF	CF2 TXIF	CF1 TXIF	CF0 TXIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.143 RSCFDnCFDCFTISTS Register Contents

Bit Position	Bit Name	Function
31 to 12	Reserved	When read, the value after a reset is read.
11	CF11TXIF	Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k transmit interrupt request is present. 1: A transmit/receive FIFO buffer k transmit interrupt request is present. (k = 0 to 11)
10	CF10TXIF	
9	CF9TXIF	
8	CF8TXIF	
7	CF7TXIF	
6	CF6TXIF	
5	CF5TXIF	
4	CF4TXIF	
3	CF3TXIF	
2	CF2TXIF	
1	CF1TXIF	
0	CF0TXIF	

The RSCFDnCFDCFTISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkTXIF Flag (k = 0 to 11)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCFDnCFDCFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

14.4.10 Details of FIFO DMA-Related Registers

14.4.10.1 RSCFDnCFDCDTCT – DMA Enable Register

Access: RSCFDnCFDCDTCT register can be read/written in 32-bit units.
RSCFDnCFDCDTCTL, RSCFDnCFDCDTCTH registers can be read/written in 16-bit units.
RSCFDnCFDCDTCTLL, RSCFDnCFDCDTCTLH, RSCFDnCFDCDTCTHL, RSCFDnCFDCDTCTHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDCDTCT: <RSCFDn_base> + 0490_H
RSCFDnCFDCDTCTL: <RSCFDn_base> + 0490_H
RSCFDnCFDCDTCTH: <RSCFDn_base> + 0492_H
RSCFDnCFDCDTCTLL: <RSCFDn_base> + 0490_H
RSCFDnCFDCDTCTLH: <RSCFDn_base> + 0491_H
RSCFDnCFDCDTCTHL: <RSCFDn_base> + 0492_H
RSCFDnCFDCDTCTHH: <RSCFDn_base> + 0493_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CFDMAE3	CFDMAE2	CFDMAE1	CFDMAE0	RFDMAE7	RFDMAE6	RFDMAE5	RFDMAE4	RFDMAE3	RFDMAE2	RFDMAE1	RFDMAE0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.144 RSCFDnCFDCDTCT Register Contents

Bit Position	Bit Name	Function
31 to 12	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
11	CFDMAE3	Transmit/Receive FIFO Buffer 9 DMA Enable 0: A DMA transfer request of transmit/receive FIFO buffer 9 is disabled. 1: A DMA transfer request of transmit/receive FIFO buffer 9 is enabled.
10	CFDMAE2	Transmit/Receive FIFO Buffer 6 DMA Enable 0: A DMA transfer request of transmit/receive FIFO buffer 6 is disabled. 1: A DMA transfer request of transmit/receive FIFO buffer 6 is enabled.
9	CFDMAE1	Transmit/Receive FIFO Buffer 3 DMA Enable 0: A DMA transfer request of transmit/receive FIFO buffer 3 is disabled. 1: A DMA transfer request of transmit/receive FIFO buffer 3 is enabled.
8	CFDMAE0	Transmit/Receive FIFO Buffer 0 DMA Enable 0: A DMA transfer request of transmit/receive FIFO buffer 0 is disabled. 1: A DMA transfer request of transmit/receive FIFO buffer 0 is enabled.
7	RFDMAE7	Receive FIFO Buffer x DMA Enable 0: A DMA transfer request of receive FIFO buffer x is disabled. 1: A DMA transfer request of receive FIFO buffer x is enabled. (x = 0 to 7)
6	RFDMAE6	
5	RFDMAE5	
4	RFDMAE4	
3	RFDMAE3	
2	RFDMAE2	
1	RFDMAE1	
0	RFDMAE0	

Modify the RSCFDnCFDCDTCT register in global operating mode or global test mode.

CFDMAEm Bit

This bit is used to enable DMA transfer for transmit/receive FIFO buffer 3 × m (the first transmit/receive FIFO buffer allocated to channel m). DMA transfer is enabled only for transmit/receive FIFO buffers for which the CFM[1:0] bits in the RSCFDnCFDCFCCk register is set to 00_B (receive mode). Set this bit to 0 when the CFM[1:0] value is 01_B (transmit mode) or 10_B (gateway mode).

RFDMAEx Bit

This bit is used to enable DMA transfer for receive FIFO buffer x.

14.4.10.2 RSCFDnCFDCDTSTS ─ DMA Status Register

Access: RSCFDnCFDCDTSTS register can be read only in 32-bit units.
RSCFDnCFDCDTSTSL, RSCFDnCFDCDTSTSH registers can be read only in 16-bit units.
RSCFDnCFDCDTSTSLL, RSCFDnCFDCDTSTSLH, RSCFDnCFDCDTSTSHL, RSCFDnCFDCDTSTSHH registers can be read only in 8-bit units.

Address: RSCFDnCFDCDTSTS: <RSCFDn_base> + 0494_H
RSCFDnCFDCDTSTSL: <RSCFDn_base> + 0494_H
RSCFDnCFDCDTSTSH: <RSCFDn_base> + 0496_H
RSCFDnCFDCDTSTSLL: <RSCFDn_base> + 0494_H
RSCFDnCFDCDTSTSLH: <RSCFDn_base> + 0495_H
RSCFDnCFDCDTSTSHL: <RSCFDn_base> + 0496_H
RSCFDnCFDCDTSTSHH: <RSCFDn_base> + 0497_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CFDMASTS3	CFDMASTS2	CFDMASTS1	CFDMASTS0	RFDMASTS7	RFDMASTS6	RFDMASTS5	RFDMASTS4	RFDMASTS3	RFDMASTS2	RFDMASTS1	RFDMASTS0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.145 RSCFDnCFDCDTSTS Register Contents

Bit Position	Bit Name	Function
31 to 12	Reserved	When read, the value after a reset is read.
11	CFDMASTS3	Transmit/Receive FIFO Buffer 9 DMA Status 0: DMA transfer of transmit/receive FIFO buffer 9 is not in progress. 1: DMA transfer of transmit/receive FIFO buffer 9 is in progress.
10	CFDMASTS2	Transmit/Receive FIFO Buffer 6 DMA Status 0: DMA transfer of transmit/receive FIFO buffer 6 is not in progress. 1: DMA transfer of transmit/receive FIFO buffer 6 is in progress.
9	CFDMASTS1	Transmit/Receive FIFO Buffer 3 DMA Status 0: DMA transfer of transmit/receive FIFO buffer 3 is not in progress. 1: DMA transfer of transmit/receive FIFO buffer 3 is in progress.
8	CFDMASTS0	Transmit/Receive FIFO Buffer 0 DMA Status 0: DMA transfer of transmit/receive FIFO buffer 0 is not in progress. 1: DMA transfer of transmit/receive FIFO buffer 0 is in progress.
7	RFDMASTS7	Receive FIFO Buffer x DMA Status 0: DMA transfer of receive FIFO buffer x is not in progress. 1: DMA transfer of receive FIFO buffer x is in progress. (x = 0 to 7)
6	RFDMASTS6	
5	RFDMASTS5	
4	RFDMASTS4	
3	RFDMASTS3	
2	RFDMASTS2	
1	RFDMASTS1	
0	RFDMASTS0	

CFDMASTSm Bit

When DMA transfer is enabled (CFDMAEm bit in the RSCFDnCFDCDTCT register is 1) for the transmit/receive FIFO buffer 3 × m (the first transmit/receive FIFO buffer allocated to channel m) while the transmit/receive FIFO buffer contains one of more messages, the CFDMASTSm bit is set to 1 indicating that DMA transfer is in progress.

When all messages in the transmit/receive FIFO buffer have been transferred or DMA transfer is disabled (CFDMAEm bit is 0), the CFDMASTSm bit is cleared to 0 indicating that DMA transfer has been completed. If the CFDMAEm bit is set to 0 during DMA transfer, the CFDMASTSm bit is cleared to 0 after the ongoing DMA transfer has been completed (when the message that is being transferred has been transferred to the last byte in the payload storage area).

These bits are cleared to 0 in global reset mode.

RFDMASTSx Bit

When DMA transfer is enabled (corresponding RFDMAEx bit in the RSCFDnCFDCDTCT register is 1) for the receive FIFO buffer x and the receive FIFO buffer contains one of more messages, the RFDMASTSx bit is set to 1 indicating that DMA transfer is in progress.

When all messages in the receive FIFO buffer x have been transferred or DMA transfer is disabled (RFDMAEx bit = 0), the RFDMASTSx bit is cleared to 0 indicating that DMA transfer has been completed. If the RFDMAEx bit is set to 0 during DMA transfer, the RFDMASTSx bit is cleared to 0 after the ongoing DMA transfer has been completed (when the message that is being transferred has been transferred to the last byte in the payload storage area).

These bits are cleared to 0 in global reset mode.

14.4.11 Details of Transmit Buffer-Related Registers

14.4.11.1 RSCFDnCFDTMCp ─ Transmit Buffer Control Register (p = 0 to 63)

Access: RSCFDnCFDTMCp registers can be read/written in 8-bit units.

Address: RSCFDnCFDTMCp: <RSCFDn_base> + 0250_H + (01_H × p)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W*1	R/W*1

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

Table 14.146 RSCFDnCFDTMCp Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after a reset is read. When writing, write the value after a reset.
2	TMOM	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	Transmit Abort Request 0: Transmit abort is not requested. 1: Transmit abort is requested.
0	TMTR	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

When the RSCFDnCFDTMCp register meets any of the following conditions, set it to 00_H.

- The RSCFDnCFDTMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCFDnCFDCFCCk register (p = m × 16 + the value of CFTML[3:0] bits).
- The RSCFDnCFDTMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCFDnCFDTXQCCm (m = 0 to 3) register (p = (m × 16 + 15) to (m × 16 + 15 - the value of TXQDC[3:0] bits)).
- RSCFDnCFDTMCp register (p = (m × 16) + 1, (m × 16) + 2), (m × 16) + 4, (m × 16) + 5) corresponding to the transmit buffer allocated as a payload storage area when the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode).

Bits in the RSCFDnCFDTMCp register are all cleared to 0 in channel reset mode. Modify the RSCFDnCFDTMCp register in channel communication mode or channel halt mode.

TMOM Bit

Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed.

Modify the TMOM bit when the TMTRM flag in the RSCFDnCFDTMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

TMTAR Bit

Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted.

The TMTAR bit can be set to 1 when TMTR bit is 1.

The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed.
- An error or arbitration loss has been detected.

If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0.

TMTR Bit

Setting this bit to 1 transmits the message stored in the transmit buffer.

The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed after the TMTAR bit was set to 1.
- An error or arbitration-lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCFDnCFDTMSTSp register is 00_B.

14.4.11.2 RSCFDnCFDTMSTSp ─ Transmit Buffer Status Register (p = 0 to 63)

Access: RSCFDnCFDTMSTSp registers can be read/written in 8-bit units.

Address: RSCFDnCFDTMSTSp: <RSCFDn_base> + $02D0_H$ + (01_H × p)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R

Table 14.147 RSCFDnCFDTMSTSp Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
4	TMTARM	Transmit Buffer Transmit Abort Request Status Flag 0: No transmit abort request is present. 1: A transmit abort request is present.
3	TMTRM	Transmit Buffer Transmit Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2, 1	TMTRF[1:0]	Transmit Buffer Transmit Result Status Flag b2 b1 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmit abort has been completed. 1 0: Transmission has been completed (without transmit abort request). 1 1: Transmission has been completed (with transmit abort request).
0	TMTSTS	Transmit Buffer Transmit Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCFDnCFDTMSTSp register is cleared to all 0 in channel reset mode.

TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCFDnCFDTMCp register is set to 1.

The TMTARM flag is set to 0 when the TMTAR bit in the RSCFDnCFDTMCp register is set to 0.

TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCFDnCFDTMCp register is set to 1.

The TMTRM flag is set to 0 when the TMTR bit in the RSCFDnCFDTMCp register is set to 0.

TMTRF[1:0] Flag

This flag indicates the result of transmission from the transmit buffer.

00_B: Transmission is in progress or no transmit request is present.

01_B: Transmission from the transmit buffer was aborted.

10_B: Transmission has been completed with the TMTAR bit in the RSCFDnCFDTMCp register set to 0 (transmit abort is not requested).

11_B: Transmission has been completed with the TMTAR bit in the RSCFDnCFDTMCp register set to 1 (transmit abort is requested).

Write 00_B to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00_B to this flag.

TMTSTS Flag

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.

14.4.11.3 RSCFDnCFDTMIDp ─ Transmit Buffer ID Register (p = 0 to 63)

Access: RSCFDnCFDTMIDp register can be read/written in 32-bit units.
RSCFDnCFDTMIDpL, RSCFDnCFDTMIDpH registers can be read/written in 16-bit units.
RSCFDnCFDTMIDpLL, RSCFDnCFDTMIDpLH, RSCFDnCFDTMIDpHL, RSCFDnCFDTMIDpHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTMIDp: <RSCFDn_base> + 4000_H + (20_H × p)
RSCFDnCFDTMIDpL: <RSCFDn_base> + 4000_H + (20_H × p)
RSCFDnCFDTMIDpH: <RSCFDn_base> + 4002_H + (20_H × p)
RSCFDnCFDTMIDpLL: <RSCFDn_base> + 4000_H + (20_H × p)
RSCFDnCFDTMIDpLH: <RSCFDn_base> + 4001_H + (20_H × p)
RSCFDnCFDTMIDpHL: <RSCFDn_base> + 4002_H + (20_H × p)
RSCFDnCFDTMIDpHH: <RSCFDn_base> + 4003_H + (20_H × p)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.148 RSCFDnCFDTMIDp Register Contents

Bit Position	Bit Name	Function
31	TMIDE	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	Transmit Buffer RTR/RRS • When the transmit message is a classical CAN frame 0: Data frame 1: Remote frame • When the transmit message is a CAN FD frame Write 0 to this bit.
29	THLEN	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID[28:0]	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11.

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

Set this bit to 0 when the TMFDF bit in the RSCFDnCFDTMFDCTRp register is 1 (CAN FD frame).

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, buffer type, and timestamp) of transmit messages is stored in the transmit history buffer after transmission is completed.

TMID[28:0] Bits

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

14.4.11.4 RSCFDnCFDTMPTRp — Transmit Buffer Pointer Register (p = 0 to 63)

Access: RSCFDnCFDTMPTRp register can be read/written in 32-bit units.
RSCFDnCFDTMPTRpL, RSCFDnCFDTMPTRpH registers can be read/written in 16-bit units.
RSCFDnCFDTMPTRpLL, RSCFDnCFDTMPTRpLH, RSCFDnCFDTMPTRpHL, RSCFDnCFDTMPTRpHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTMPTRp: <RSCFDn_base> + 4004_H + (20_H × p)
RSCFDnCFDTMPTRpL: <RSCFDn_base> + 4004_H + (20_H × p)
RSCFDnCFDTMPTRpH: <RSCFDn_base> + 4006_H + (20_H × p)
RSCFDnCFDTMPTRpLL: <RSCFDn_base> + 4004_H + (20_H × p)
RSCFDnCFDTMPTRpLH: <RSCFDn_base> + 4005_H + (20_H × p)
RSCFDnCFDTMPTRpHL: <RSCFDn_base> + 4006_H + (20_H × p)
RSCFDnCFDTMPTRpHH: <RSCFDn_base> + 4007_H + (20_H × p)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.149 RSCFDnCFDTMPTRp Register Contents (1/2)

Bit Position	Bit Name	Function			
31 to 28	TMDLC[3:0]	Transmit Buffer DLC Data			
		b31 b30 b29 b28	Classical CAN Frame	CAN FD Frame	
		0 0 0 0	0 data bytes		
		0 0 0 1	1 data byte		
		0 0 1 0	2 data bytes		
		0 0 1 1	3 data bytes		
		0 1 0 0	4 data bytes		
		0 1 0 1	5 data bytes		
		0 1 1 0	6 data bytes		
		0 1 1 1	7 data bytes		
		1 0 0 0	8 data bytes		
		1 0 0 1	8 data bytes	12 data bytes	
		1 0 1 0		16 data bytes	
		1 0 1 1		20 data bytes	
		1 1 0 0		24 data bytes	
		1 1 0 1		32 data bytes	
		1 1 1 0		48 data bytes	
		1 1 1 1		64 data bytes	

Table 14.149 RSCFDnCFDTMPTRp Register Contents (2/2)

Bit Position	Bit Name	Function
27 to 24	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
23 to 16	TMPTR[7:0]	Transmit Buffer Label Data Set the label information to be stored in the transmit history buffer.
15 to 0	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCFDnCFDTMIDp register is set to 0 (data frame).

When the TMDLC[3:0] bits are set to 1001_B or more while the TMFDF bit in the RSCFDnCFDTMFDCTRp register is 0 (classical CAN frame), 8-byte data is transmitted actually. When the MFDF bit is 1 (CAN FD frame), the settable value range varies depending on the setting of the TMME bit in the RSCFDnCFDCmFDCFG register.

- When the TMME bit = 0 (transmit buffer merge mode disabled):
  A value of 0000_B to 1111_B is settable. If a value larger than 1100_B is set, payloads exceeding 20 bytes are padded by CC_H.
- When the TMME bit = 1 (transmit buffer merge mode enabled):
  When the corresponding transmit buffer number p = (m × 16) + 0 or (m × 16) + 3, a value of 0000_B to 1111_B is settable. In other cases, set a value of 0000_B to 1011_B (20 data bytes).

When the TMRTR bit is 1 (remote frame), set the data length of a message to be requested.

TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

14.4.11.5 RSCFDnCFDTMFDCTRp ─ Transmit Buffer CAN FD Configuration Register (p = 0 to 63)

Access: RSCFDnCFDTMFDCTRp register can be read/written in 32-bit units.
RSCFDnCFDTMFDCTRpL, RSCFDnCFDTMFDCTRpH registers can be read/written in 16-bit units.
RSCFDnCFDTMFDCTRpLL, RSCFDnCFDTMFDCTRpLH, RSCFDnCFDTMFDCTRpHL,
RSCFDnCFDTMFDCTRpHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTMFDCTRp: <RSCFDn_base> + 4008_H + (20_H × p)
RSCFDnCFDTMFDCTRpL: <RSCFDn_base> + 4008_H + (20_H × p)
RSCFDnCFDTMFDCTRpH: <RSCFDn_base> + $400A_H$ + (20_H × p)
RSCFDnCFDTMFDCTRpLL: <RSCFDn_base> + 4008_H + (20_H × p)
RSCFDnCFDTMFDCTRpLH: <RSCFDn_base> + 4009_H + (20_H × p)
RSCFDnCFDTMFDCTRpHL: <RSCFDn_base> + $400A_H$ + (20_H × p)
RSCFDnCFDTMFDCTRpHH: <RSCFDn_base> + $400B_H$ + (20_H × p)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TMFDF	TMBRS	TMESI
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 14.150 RSCFDnCFDTMFDCTRp Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	TMFDF	FDF 0: Classical CAN frame 1: CAN FD frame
1	TMBRS	BRS 0: The bit rate in the data area does not change. 1: The bit rate in the data area changes.
0	TMESI	ESI 0: Error active node 1: Error passive node

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is 0 (transmission not requested). When this register is linked to the transmit/receive FIFO buffer, do not write data to this register. When this register is allocated to the transmit queue, write data only to transmit buffer p (p = m × 16 + 15) of the corresponding channel.

TMFDF Bit

This bit is used to set the FD format of the message to be transmitted from the transmit buffer.

TMBRS Bit

When this bit is set to 1 while the TMFDF bit is 1, the data area of a transmit message is transmitted at the data bit rate. When the TMFDF bit is 0, write 0 to this bit.

TMESI Bit

This bit is used to set the ESI bit value of the message to be transmitted from the transmit buffer when the TMFDF bit is 1. The set value is transmitted when the ESIC bit in the RSCFDnCFDCmFDCFG register is 1 and the channel is in the error active state. When the channel is in the error passive state, the ESI bit value that shows an error passive node is transmitted regardless of this bit value. When the TMFDF bit is 0, write 0 to this bit.

14.4.11.6 RSCFDnCFDTMDFb_p ─ Transmit Buffer Data Field b Register (b = 0 to 4, p = 0 to 63)

Access: RSCFDnCFDTMDFb_p register can be read/written in 32-bit units.
RSCFDnCFDTMDFb_pL, RSCFDnCFDTMDFb_pH registers can be read/written in 16-bit units.
RSCFDnCFDTMDFb_pLL, RSCFDnCFDTMDFb_pLH, RSCFDnCFDTMDFb_pHL, RSCFDnCFDTMDFb_pHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTMDFb_p: <RSCFDn_base> + $400C_H$ + (04_H × b) + (20_H × p)
RSCFDnCFDTMDFb_pL: <RSCFDn_base> + $400C_H$ + (04_H × b) + (20_H × p)
RSCFDnCFDTMDFb_pH: <RSCFDn_base> + $400E_H$ + (04_H × b) + (20_H × p)
RSCFDnCFDTMDFb_pLL: <RSCFDn_base> + $400C_H$ + (04_H × b) + (20_H × p)
RSCFDnCFDTMDFb_pLH: <RSCFDn_base> + $400D_H$ + (04_H × b) + (20_H × p)
RSCFDnCFDTMDFb_pHL: <RSCFDn_base> + $400E_H$ + (04_H × b) + (20_H × p)
RSCFDnCFDTMDFb_pHH: <RSCFDn_base> + $400F_H$ + (04_H × b) + (20_H × p)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB4 × b + 3 [7:0]								TMDB4 × b + 2 [7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB4 × b + 1 [7:0]								TMDB4 × b + 0 [7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.151 RSCFDnCFDTMDFb_p Register Contents

Bit Position	Bit Name	Function
31 to 24	TMDB4 × b + 3 [7:0]	Transmit buffer data byte 4 × b + 3
23 to 16	TMDB4 × b + 2 [7:0]	Transmit buffer data byte 4 × b + 2
15 to 8	TMDB4 × b + 1 [7:0]	Transmit buffer data byte 4 × b + 1
7 to 0	TMDB4 × b + 0 [7:0]	Transmit buffer data byte 4 × b + 0 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCFDnCFDTMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

14.4.11.7 RSCFDnCFDTMIECy ─ Transmit Buffer Interrupt Enable Configuration Register (y = 0, 1)

Access: RSCFDnCFDTMIECy register can be read/written in 32-bit units.
RSCFDnCFDTMIECyL, RSCFDnCFDTMIECyH registers can be read/written in 16-bit units.
RSCFDnCFDTMIECyLL, RSCFDnCFDTMIECyLH, RSCFDnCFDTMIECyHL, RSCFDnCFDTMIECyHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTMIECy: <RSCFDn_base> + 0390_H + (04_H × y)
RSCFDnCFDTMIECyL: <RSCFDn_base> + 0390_H + (04_H × y)
RSCFDnCFDTMIECyH: <RSCFDn_base> + 0392_H + (04_H × y)
RSCFDnCFDTMIECyLL: <RSCFDn_base> + 0390_H + (04_H × y)
RSCFDnCFDTMIECyLH: <RSCFDn_base> + 0391_H + (04_H × y)
RSCFDnCFDTMIECyHL: <RSCFDn_base> + 0392_H + (04_H × y)
RSCFDnCFDTMIECyHH: <RSCFDn_base> + 0393_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.152 RSCFDnCFDTMIECy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMIEp	Transmit Buffer Interrupt Enable p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.
15 to 0	TMIEp	Transmit Buffer Interrupt Enable p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

TMIEp Bits (p = 0 to 63)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCFDnCFDTMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue. When the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode enable), set the bit corresponding to the transmit buffer allocated as a payload storage area to 0.

Table 14.153 shows the bit assignment.

Table 14.153 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.4.12 Details of Transmit Buffer Status-Related Registers

14.4.12.1 RSCFDnCFDTMTRSTSy — Transmit Buffer Transmit Request Status Register (y = 0, 1)

Access: RSCFDnCFDTMTRSTSy register can be read only in 32-bit units.
RSCFDnCFDTMTRSTSyL, RSCFDnCFDTMTRSTSyH registers can be read only in 16-bit units.
RSCFDnCFDTMTRSTSyLL, RSCFDnCFDTMTRSTSyLH, RSCFDnCFDTMTRSTSyHL, RSCFDnCFDTMTRSTSyHH registers can be read only in 8-bit units.

Address: RSCFDnCFDTMTRSTSy: <RSCFDn_base> + 0350_H + (04_H × y)
RSCFDnCFDTMTRSTSyL: <RSCFDn_base> + 0350_H + (04_H × y)
RSCFDnCFDTMTRSTSyH: <RSCFDn_base> + 0352_H + (04_H × y)
RSCFDnCFDTMTRSTSyLL: <RSCFDn_base> + 0350_H + (04_H × y)
RSCFDnCFDTMTRSTSyLH: <RSCFDn_base> + 0351_H + (04_H × y)
RSCFDnCFDTMTRSTSyHL: <RSCFDn_base> + 0352_H + (04_H × y)
RSCFDnCFDTMTRSTSyHH: <RSCFDn_base> + 0353_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.154 RSCFDnCFDTMTRSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit request is present. 1: A transmit request is present.
15 to 0	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

TMTRSTSp Flags (p = 0 to 63)

These flags indicate the status of the TMTR bit in the RSCFDnCFDTMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 14.155 shows the bit assignment.

Table 14.155 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.4.12.2 RSCFDnCFDTMTARSTSy ─ Transmit Buffer Transmit Abort Request Status Register (y = 0, 1)

Access: RSCFDnCFDTMTARSTSy register can be read only in 32-bit units.
RSCFDnCFDTMTARSTSyL, RSCFDnCFDTMTARSTSyH registers can be read only in 16-bit units.
RSCFDnCFDTMTARSTSyLL, RSCFDnCFDTMTARSTSyLH, RSCFDnCFDTMTARSTSyHL,
RSCFDnCFDTMTARSTSyHH registers can be read only in 8-bit units.

Address: RSCFDnCFDTMTARSTSy: <RSCFDn_base> + 0360_H + (04_H × y)
RSCFDnCFDTMTARSTSyL: <RSCFDn_base> + 0360_H + (04_H × y)
RSCFDnCFDTMTARSTSyH: <RSCFDn_base> + 0362_H + (04_H × y)
RSCFDnCFDTMTARSTSyLL: <RSCFDn_base> + 0360_H + (04_H × y)
RSCFDnCFDTMTARSTSyLH: <RSCFDn_base> + 0361_H + (04_H × y)
RSCFDnCFDTMTARSTSyHL: <RSCFDn_base> + 0362_H + (04_H × y)
RSCFDnCFDTMTARSTSyHH: <RSCFDn_base> + 0363_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.156 RSCFDnCFDTMTARSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit abort request is present. 1: A transmit abort request is present.
15 to 0	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present.

TMTARSTSp Flags (p = 0 to 63)

These flags indicate the status of the TMTAR bit in the RSCFDnCFDTMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 14.157 shows the bit assignment.

Table 14.157 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.4.12.3 RSCFDnCFDTMTCSTSy ─ Transmit Buffer Transmit Complete Status Register (y = 0, 1)

Access: RSCFDnCFDTMTCSTSy register can be read only in 32-bit units.
RSCFDnCFDTMTCSTSyL, RSCFDnCFDTMTCSTSyH registers can be read only in 16-bit units.
RSCFDnCFDTMTCSTSyLL, RSCFDnCFDTMTCSTSyLH, RSCFDnCFDTMTCSTSyHL, RSCFDnCFDTMTCSTSyHH registers can be read only in 8-bit units.

Address: RSCFDnCFDTMTCSTSy: <RSCFDn_base> + 0370_H + (04_H × y)
RSCFDnCFDTMTCSTSyL: <RSCFDn_base> + 0370_H + (04_H × y)
RSCFDnCFDTMTCSTSyH: <RSCFDn_base> + 0372_H + (04_H × y)
RSCFDnCFDTMTCSTSyLL: <RSCFDn_base> + 0370_H + (04_H × y)
RSCFDnCFDTMTCSTSyLH: <RSCFDn_base> + 0371_H + (04_H × y)
RSCFDnCFDTMTCSTSyHL: <RSCFDn_base> + 0372_H + (04_H × y)
RSCFDnCFDTMTCSTSyHH: <RSCFDn_base> + 0373_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.158 RSCFDnCFDTMTCSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission has not been completed. 1: Transmission has been completed.
15 to 0	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

TMTCSTSp Flags (p = 0 to 63)

When the TMTRF[1:0] flag in the RSCFDnCFDTMSTSp register is set to 10_B (transmission has been completed (without transmit abort request)) or 11_B (transmission has been completed (with transmit abort request)), the corresponding TMTCSTSp flag is set to 1.

A TMTCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 14.159 shows the bit assignment.

Table 14.159 TMTCSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.4.12.4 RSCFDnCFDTMTASTSy – Transmit Buffer Transmit Abort Status Register (y = 0, 1)

Access: RSCFDnCFDTMTASTSy register can be read only in 32-bit units.
RSCFDnCFDTMTASTSyL, RSCFDnCFDTMTASTSyH registers can be read only in 16-bit units.
RSCFDnCFDTMTASTSyLL, RSCFDnCFDTMTASTSyLH, RSCFDnCFDTMTASTSyHL, RSCFDnCFDTMTASTSyHH registers can be read only in 8-bit units.

Address: RSCFDnCFDTMTASTSy: <RSCFDn_base> + 0380_H + (04_H × y)
RSCFDnCFDTMTASTSyL: <RSCFDn_base> + 0380_H + (04_H × y)
RSCFDnCFDTMTASTSyH: <RSCFDn_base> + 0382_H + (04_H × y)
RSCFDnCFDTMTASTSyLL: <RSCFDn_base> + 0380_H + (04_H × y)
RSCFDnCFDTMTASTSyLH: <RSCFDn_base> + 0381_H + (04_H × y)
RSCFDnCFDTMTASTSyHL: <RSCFDn_base> + 0382_H + (04_H × y)
RSCFDnCFDTMTASTSyHH: <RSCFDn_base> + 0383_H + (04_H × y)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTASTSp (p = y × 32 + 31 to y x 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.160 RSCFDnCFDTMTASTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted. 1: Transmission is aborted.
15 to 0	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

TMTASTSp Flags (p = 0 to 63)

When the TMTRF[1:0] flag in the RSCFDnCFDTMSTSp register is set to 01_B (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 14.161 shows the bit assignment.

Table 14.161 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.4.13 Details of Transmit Queue-Related Registers

14.4.13.1 RSCFDnCFDTXQCCm — Transmit Queue Configuration and Control Register (m = 0 to 3)

Access: RSCFDnCFDTXQCCm register can be read/written in 32-bit units.
RSCFDnCFDTXQCCmL, RSCFDnCFDTXQCCmH registers can be read/written in 16-bit units.
RSCFDnCFDTXQCCmLL, RSCFDnCFDTXQCCmLH, RSCFDnCFDTXQCCmHL, RSCFDnCFDTXQCCmHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTXQCCm: <RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCFDnCFDTXQCCmL: <RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCFDnCFDTXQCCmH: <RSCFDn_base> + $03A2_H$ + (04_H × m)
RSCFDnCFDTXQCCmLL: <RSCFDn_base> + $03A0_H$ + (04_H × m)
RSCFDnCFDTXQCCmLH: <RSCFDn_base> + $03A1_H$ + (04_H × m)
RSCFDnCFDTXQCCmHL: <RSCFDn_base> + $03A2_H$ + (04_H × m)
RSCFDnCFDTXQCCmHH: <RSCFDn_base> + $03A3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]				—	—	—	—	—	—	—	TXQE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 14.162 RSCFDnCFDTXQCCm Register Contents

Bit Position	Bit Name	Function
31 to 14	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
13	TXQIM	Transmit Queue Interrupt Source Select 0: When the transmit queue becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted.
12	TXQIE	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC[3:0]	Transmit Queue Depth Configuration Setting these bits to g (g = 2 to 15) makes the (g + 1)-buffer transmit queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited. For transmit buffer merge mode, set g to 2 to 9.
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	TXQE	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated.

Set the TXQE bit to 0 before modifying the TXQIE bit.

TXQDC[3:0] Bits

These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from (m × 16 + 15) to (m × 16 + 0). For examples of how buffer allocation is done, see **Figure 14.9**.

When the TMME bit in the RSCFDnCFDCmFDCFG register is 1 (transmit buffer merge mode), transmit buffers (m × 16 + 5) to (m × 16 + 0) are merged and cannot be allocated to the transmit queue. Therefore, do not set TXQDC[3:0] bits to 10 to 15.

Modify these bits only in channel reset mode.

TXQE Bit

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.

Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to 0010_B or more.

14.4.13.2 RSCFDnCFDTXQSTSm – Transmit Queue Status Register (m = 0 to 3)

Access: RSCFDnCFDTXQSTSm register can be read/written in 32-bit units.
RSCFDnCFDTXQSTSmL, RSCFDnCFDTXQSTSmH registers can be read/written in 16-bit units.
RSCFDnCFDTXQSTSmLL, RSCFDnCFDTXQSTSmLH, RSCFDnCFDTXQSTSmHL, RSCFDnCFDTXQSTSmHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTXQSTSm: <RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCFDnCFDTXQSTSmL: <RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCFDnCFDTXQSTSmH: <RSCFDn_base> + $03C2_H$ + (04_H × m)
RSCFDnCFDTXQSTSmLL: <RSCFDn_base> + $03C0_H$ + (04_H × m)
RSCFDnCFDTXQSTSmLH: <RSCFDn_base> + $03C1_H$ + (04_H × m)
RSCFDnCFDTXQSTSmHL: <RSCFDn_base> + $03C2_H$ + (04_H × m)
RSCFDnCFDTXQSTSmHH: <RSCFDn_base> + $03C3_H$ + (04_H × m)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQFLL	TXQEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.163 RSCFDnCFDTXQSTSm Register Contents

Bit Position	Bit Name	Function
31 to 13	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
12 to 8	Reserved	When read, the undefined value is returned. When writing to these bits, write the value after reset.
7 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	TXQIF	Transmit Queue Interrupt Request Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TXQFLL	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

TXQIF Flag

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCFDnCFDTXQCCm register has occurred.

The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCFDnCFDTXQCCm register to 0 (the transmit queue is not used).

TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCFDnCFDTXQCCm register.

This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

TXQEMP Flag

The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.

This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

14.4.13.3 RSCFDnCFDTXQPCTRm ─ Transmit Queue Pointer Control Register (m = 0 to 3)

Access: RSCFDnCFDTXQPCTRm register can only be written in 32-bit units.
RSCFDnCFDTXQPCTRmL, RSCFDnCFDTXQPCTRmH registers can only be written in 16-bit units.
RSCFDnCFDTXQPCTRmLL, RSCFDnCFDTXQPCTRmLH, RSCFDnCFDTXQPCTRmHL,
RSCFDnCFDTXQPCTRmHH registers can only be written in 8-bit units.

Address: RSCFDnCFDTXQPCTRm: <RSCFDn_base> + $03E0_H$ + (04_H × m)
RSCFDnCFDTXQPCTRmL: <RSCFDn_base> + $03E0_H$ + (04_H × m)
RSCFDnCFDTXQPCTRmH: <RSCFDn_base> + $03E2_H$ + (04_H × m)
RSCFDnCFDTXQPCTRmLL: <RSCFDn_base> + $03E0_H$ + (04_H × m)
RSCFDnCFDTXQPCTRmLH: <RSCFDn_base> + $03E1_H$ + (04_H × m)
RSCFDnCFDTXQPCTRmHL: <RSCFDn_base> + $03E2_H$ + (04_H × m)
RSCFDnCFDTXQPCTRmHH: <RSCFDn_base> + $03E3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 14.164 RSCFDnCFDTXQPCTRm Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	TXQPC[7:0]	Transmit Queue Pointer Control Writing FF_H to these bits moves the write pointer of the transmit queue to the next queue buffer.

TXQPC[7:0] Bits

Writing FF_H to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCFDnCFDTMIDp, RSCFDnCFDTMPTRp, RSCFDnCFDTMFDCTRp, and RSCFDnCFDTMDFb_p registers (p = 15, 31, 47, 63) before writing FF_H to the TXQPC[7:0] bits.

When writing FF_H to these bits, make sure that the TXQE bit in the RSCFDnCFDTXQCCm register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCFDnCFDTXQSTSm register is 0 (he transmit queue is not full).

14.4.14 Details of Transmit History-Related Registers

14.4.14.1 RSCFDnCFDTHLCCm ─ Transmit History Configuration and Control Register (m = 0 to 3)

Access: RSCFDnCFDTHLCCm register can be read/written in 32-bit units.
RSCFDnCFDTHLCCmL, RSCFDnCFDTHLCCmH registers can be read/written in 16-bit units.
RSCFDnCFDTHLCCmLL, RSCFDnCFDTHLCCmLH, RSCFDnCFDTHLCCmHL, RSCFDnCFDTHLCCmHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTHLCCm: <RSCFDn_base> + 0400_H + (04_H × m)
RSCFDnCFDTHLCCmL: <RSCFDn_base> + 0400_H + (04_H × m)
RSCFDnCFDTHLCCmH: <RSCFDn_base> + 0402_H + (04_H × m)
RSCFDnCFDTHLCCmLL: <RSCFDn_base> + 0400_H + (04_H × m)
RSCFDnCFDTHLCCmLH: <RSCFDn_base> + 0401_H + (04_H × m)
RSCFDnCFDTHLCCmHL: <RSCFDn_base> + 0402_H + (04_H × m)
RSCFDnCFDTHLCCmHH: <RSCFDn_base> + 0403_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 14.165 RSCFDnCFDTHLCCm Register Contents

Bit Position	Bit Name	Function
31 to 11	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
10	THLDTE	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	THLE	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

THLIM Bit

This bit is used to select a transmit history interrupt source.

Modify this bit only in channel reset mode.

THLIE Bit

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated.

Modify the THLIE bit only when the THLE bit set to 0.

THLE Bit

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer.

Modify this bit in channel communication mode or channel halt mode.

This bit is cleared to 0 in channel reset mode.

14.4.14.2 RSCFDnCFDTHLSTSm ─ Transmit History Status Register (m = 0 to 3)

Access: RSCFDnCFDTHLSTSm register can be read/written in 32-bit units.
RSCFDnCFDTHLSTSmL, RSCFDnCFDTHLSTSmH registers can be read/written in 16-bit units.
RSCFDnCFDTHLSTSmLL, RSCFDnCFDTHLSTSmLH, RSCFDnCFDTHLSTSmHL, RSCFDnCFDTHLSTSmHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDTHLSTSm: <RSCFDn_base> + 0420_H + (04_H × m)
RSCFDnCFDTHLSTSmL: <RSCFDn_base> + 0420_H + (04_H × m)
RSCFDnCFDTHLSTSmH: <RSCFDn_base> + 0422_H + (04_H × m)
RSCFDnCFDTHLSTSmLL: <RSCFDn_base> + 0420_H + (04_H × m)
RSCFDnCFDTHLSTSmLH: <RSCFDn_base> + 0421_H + (04_H × m)
RSCFDnCFDTHLSTSmHL: <RSCFDn_base> + 0422_H + (04_H × m)
RSCFDnCFDTHLSTSmHH: <RSCFDn_base> + 0423_H + (04_H × m)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]					—	—	—	—	THLIF	THLELT	THLFLL	THLEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W *1	R/W *1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.166 RSCFDnCFDTHLSTSm Register Contents

Bit Position	Bit Name	Function
31 to 13	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
12 to 8	THLMC[4:0]	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	THLIF	Transmit History Interrupt Request Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
2	THLELT	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	Transmit history Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

THLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer. These bits are cleared to 0 in channel reset mode.

THLIF Flag

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCFDnCFDTHLCCm register occurs.

This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLELT Flag

The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flags of the register to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using store instruction, set the bit to be set to “0” to “0”, and the bits not to be set to “0” to “1”.

THLFLL Flag

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCFDnCFDTHLCCm register is set to 0 (transmit history buffer is not used).

THLEMP Flag

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.

This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCFDnCFDTHLCCm register is set to 0 (transmit history buffer is not used).

NOTE

To clear THLIF or THLELT flag to 0, the program must write 0. When writing, use a store instruction to write “0” to the given flag and “1” to other flags.

14.4.14.3 RSCFDnCFDTHLPCTRm ─ Transmit History Pointer Control Register (m = 0 to 3)

Access: RSCFDnCFDTHLPCTRm register can only be written in 32-bit units.
RSCFDnCFDTHLPCTRmL, RSCFDnCFDTHLPCTRmH registers can only be written in 16-bit units.
RSCFDnCFDTHLPCTRmLL, RSCFDnCFDTHLPCTRmLH, RSCFDnCFDTHLPCTRmHL,
RSCFDnCFDTHLPCTRmHH registers can only be written in 8-bit units.

Address: RSCFDnCFDTHLPCTRm: <RSCFDn_base> + 0440_H + (04_H × m)
RSCFDnCFDTHLPCTRmL: <RSCFDn_base> + 0440_H + (04_H × m)
RSCFDnCFDTHLPCTRmH: <RSCFDn_base> + 0442_H + (04_H × m)
RSCFDnCFDTHLPCTRmLL: <RSCFDn_base> + 0440_H + (04_H × m)
RSCFDnCFDTHLPCTRmLH: <RSCFDn_base> + 0441_H + (04_H × m)
RSCFDnCFDTHLPCTRmHL: <RSCFDn_base> + 0442_H + (04_H × m)
RSCFDnCFDTHLPCTRmHH: <RSCFDn_base> + 0443_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 14.167 RSCFDnCFDTHLPCTRm Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When writing to these bits, write the value after reset.
7 to 0	THLPC[7:0]	Transmit History List Pointer Control Writing FF_H to these bits moves the read pointer to the next unread data in the transmit history buffer.

THLPC[7:0] Bits

When the THLPC[7:0] bits are set to FF_H, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCFDnCFDTHLSTSm register is decremented. Write FF_H to the THLPC[7:0] bits after reading from the RSCFDnCFDTHLACCm register.

When writing FF_H to these bits, make sure that the THLE bit in the RSCFDnCFDTHLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCFDnCFDTHLSTSm register is 0.

14.4.14.4 RSCFDnCFDTHLACCm – Transmit History Access Register (m = 0 to 3)

Access: RSCFDnCFDTHLACCm register can be read only in 32-bit units.
RSCFDnCFDTHLACCmL, RSCFDnCFDTHLACCmH registers can be read only in 16-bit units.
RSCFDnCFDTHLACCmLL, RSCFDnCFDTHLACCmLH, RSCFDnCFDTHLACCmHL, RSCFDnCFDTHLACCmHH registers can be read only in 8-bit units.

Address: RSCFDnCFDTHLACCm: <RSCFDn_base> + 6000_H + (04_H × m)
RSCFDnCFDTHLACCmL: <RSCFDn_base> + 6000_H + (04_H × m)
RSCFDnCFDTHLACCmH: <RSCFDn_base> + 6002_H + (04_H × m)
RSCFDnCFDTHLACCmLL: <RSCFDn_base> + 6000_H + (04_H × m),
RSCFDnCFDTHLACCmLH: <RSCFDn_base> + 6001_H + (04_H × m)
RSCFDnCFDTHLACCmHL: <RSCFDn_base> + 6002_H + (04_H × m)
RSCFDnCFDTHLACCmHH: <RSCFDn_base> + 6003_H + (04_H × m)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]								—	BN[3:0]				BT[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.168 RSCFDnCFDTHLACCm Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTS[15:0]	Timestamp Data The timestamp data of stored data can be read.
15 to 8	TID[7:0]	Label Data The label information of stored data can be read.
7	Reserved	When read, the value after reset is returned.
6 to 3	BN[3:0]	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.
2 to 0	BT[2:0]	Buffer Type Data b2 b1 b0 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 1 0 0: Transmit queue

TMTS[15:0] Bits

Timestamp values in transmit history data stored in the transmit history buffer are displayed.

TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

BN[3:0] Bits

These bits indicate the transmit source buffer number in the transmit history data stored in the transmit history buffer.

BT[2:0] Bits

These bits indicate the type of the transmit source buffer in the transmit history data stored in the transmit history buffer.

14.4.15 Details of Test-Related Registers

14.4.15.1 RSCFDnCFDGTSTCFG ─ Global Test Configuration Register

Access: RSCFDnCFDGTSTCFG register can be read/written in 32-bit units.
RSCFDnCFDGTSTCFGL, RSCFDnCFDGTSTCFGH registers can be read/written in 16-bit units.
RSCFDnCFDGTSTCFGLL, RSCFDnCFDGTSTCFGLH, RSCFDnCFDGTSTCFGHL, RSCFDnCFDGTSTCFGHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGTSTCFG: <RSCFDn_base> + 0468_H
RSCFDnCFDGTSTCFGL: <RSCFDn_base> + 0468_H, RSCFDnCFDGTSTCFGH: <RSCFDn_base> + $046A_H$
RSCFDnCFDGTSTCFGLL: <RSCFDn_base> + 0468_H, RSCFDnCFDGTSTCFGLH: <RSCFDn_base> + 0469_H,
RSCFDnCFDGTSTCFGHL: <RSCFDn_base> + $046A_H$, RSCFDnCFDGTSTCFGHH: <RSCFDn_base> + $046B_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	C3ICBCE	C2ICBCE	C1ICBCE	C0ICBCE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 14.169 RSCFDnCFDGTSTCFG Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
22 to 16	RTMPS[6:0]	RAM Test Page Configuration Set a value within a range of page 0 (00_H) to page 55 (37_H).
15 to 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3	C3ICBCE	CAN3 Inter-channel Communication Test Enable 0: CAN3 inter-channel communication test is disabled. 1: CAN3 inter-channel communication test is enabled.
2	C2ICBCE	CAN2 Inter-channel Communication Test Enable 0: CAN2 inter-channel communication test is disabled. 1: CAN2 inter-channel communication test is enabled.
1	C1ICBCE	CAN1 Inter-channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	CAN0 Inter-channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCFDnCFDGTSTCFG register only in global test mode.

RTMPS[6:0] Bits

These bits are used to set the RAM test target page number for RAM test. Set a value in the range of 00_H to 37_H, inclusive. In CAN FD mode, do not access more than 128 bytes in the last page (RTMPS[6:0] = 37_H) during RAM test.

C3ICBCE Bit

Setting this bit to 1 enables the channel 3 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

C2ICBCE Bit

Setting this bit to 1 enables the channel 2 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

C1ICBCE Bit

Setting this bit to 1 enables the channel 1 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

C0ICBCE Bit

Setting this bit to 1 enables the channel 0 inter-channel communication test.

This bit is cleared to 0 in global reset mode.

14.4.15.2 RSCFDnCFDGTSTCTR ─ Global Test Control Register

Access: RSCFDnCFDGTSTCTR register can be read/written in 32-bit units.
RSCFDnCFDGTSTCTRL, RSCFDnCFDGTSTCTRH registers can be read/written in 16-bit units.
RSCFDnCFDGTSTCTRLL, RSCFDnCFDGTSTCTRLH, RSCFDnCFDGTSTCTRHL, RSCFDnCFDGTSTCTRHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDGTSTCTR: <RSCFDn_base> + $046C_H$
RSCFDnCFDGTSTCTRL: <RSCFDn_base> + $046C_H$, RSCFDnCFDGTSTCTRH: <RSCFDn_base> + $046E_H$
RSCFDnCFDGTSTCTRLL: <RSCFDn_base> + $046C_H$, RSCFDnCFDGTSTCTRLH: <RSCFDn_base> + $046D_H$,
RSCFDnCFDGTSTCTRHL: <RSCFDn_base> + $046E_H$, RSCFDnCFDGTSTCTRHH: <RSCFDn_base> + $046F_H$

Value after reset: 0000 0000 H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	ICBCTME
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Table 14.170 RSCFDnCFDGTSTCTR Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	RTME	RAM Test Enable 0: RAM test is disabled. 1: RAM test is enabled.
1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	ICBCTME	Communication Test between Channels Enable 0: Communication test between channels disabled 1: Communication test between channels enabled

RTME Bit

Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode. This bit is cleread to 0 in global reset mode.

1. Set the GMDC[1:0] bits in the RSCFDnCFDGCTR register to 10_B (Global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0 to 3) in the RSCFDnCFDGTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode. This bit is cleared to 0 in global reset mode.

14.4.15.3 RSCFDnCFDGLOCKK ─ Global Lock Key Register

Access: RSCFDnCFDGLOCKK register can only be written in 32-bit units.
RSCFDnCFDGLOCKKL, RSCFDnCFDGLOCKKH registers can only be written in 16-bit units.

Address: RSCFDnCFDGLOCKK: <RSCFDn_base> + $047C_H$
RSCFDnCFDGLOCKKL: <RSCFDn_base> + $047C_H$, RSCFDnCFDGLOCKKH: <RSCFDn_base> + $047E_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1

Note 1. Writing to these bits is effective only when the RS-CANFD module is in global test mode.

Table 14.171 RSCFDnCFDGLOCKK Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When writing these bits, write the value after reset.
15 to 0	LOCK[15:0]	Lock Key These bits are key bits to release protection of test mode.

The RSCFDnCFDGLOCKK register releases protection of special test bits and is write only.

For the protection release data, see **Section 14.11.4.2, Procedure for Releasing the Protection**.

LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCFDnCFDGTSTCTR register.

After the protection has been released, writing to the I/O register area (<RSCFDn_base> + 0000_H to <RSCFDn_base> + $0FFF_H$) of the CAN (except the RAM) enables the protection again.

Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.

14.4.15.4 RSCFDnCFDRPGACCr ─ RAM Test Page Access Register (r = 0 to 63)

Access: RSCFDnCFDRPGACCr register can be read/written in 32-bit units.
RSCFDnCFDRPGACCrL, RSCFDnCFDRPGACCrH registers can be read/written in 16-bit units.
RSCFDnCFDRPGACCrLL, RSCFDnCFDRPGACCrLH, RSCFDnCFDRPGACCrHL, RSCFDnCFDRPGACCrHH registers can be read/written in 8-bit units.

Address: RSCFDnCFDRPGACCr: <RSCFDn_base> + 6400_H + (04_H × r)
RSCFDnCFDRPGACCrL: <RSCFDn_base> + 6400_H + (04_H × r)
RSCFDnCFDRPGACCrH: <RSCFDn_base> + 6402_H + (04_H × r)
RSCFDnCFDRPGACCrLL: <RSCFDn_base> + 6400_H + (04_H × r)
RSCFDnCFDRPGACCrLH: <RSCFDn_base> + 6401_H + (04_H × r)
RSCFDnCFDRPGACCrHL: <RSCFDn_base> + 6402_H + (04_H × r)
RSCFDnCFDRPGACCrHH: <RSCFDn_base> + 6403_H + (04_H × r)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDTA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDTA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.172 RSCFDnCFDRPGACCr Register Contents

Bit Position	Bit Name	Function
31 to 0	RDTA[31:0]	RAM Data Test Access Data can be read and written in CAN RAM.

Modify the RSCFDnCFDRPGACCr register in global test mode with the RTME bit in the RSCFDnCFDGTSTCTR register set to 1 (RAM test is enabled).

The RSCFDnCFDRPGACCr register is readable and writable when the RTME bit is set to 1.

14.5 Interrupt Sources and DMA Trigger

14.5.1 Interrupt Sources

The RS-CANFD module has 14 interrupts that are grouped into global interrupts and channel interrupts.

- Global interrupts (2 sources):
  Receive FIFO interrupt
  Global error interrupt
- Channel interrupts (3 sources/channel):
  CANm transmit interrupt (m = 0 to 3)
  - CANm transmit complete interrupt
  - CANm transmit abort interrupt
  - CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode)
  - CANm transmit history interrupt
  - CANm transmit queue Interrupt

CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode)
CANm error interrupt

When an interrupt request is generated, the corresponding interrupt request flag is set to 1 (interrupt request present). In that case, when the interrupt enable bit is set to 1 (enabling interrupts), an interrupt request is output from the RS-CANFD module. (Generation of interrupts also depends on the interrupt control register settings of the interrupt controller.)

Setting the interrupt request flag to 0 (no interrupt request present) or setting the interrupt enable bit to 0 (disabling interrupts) clears the current interrupt request. The current interrupt request is still output until the interrupt request flag is cleared.

Table 14.173 lists the CAN interrupt sources. **Figure 14.2** shows the CAN global interrupt block diagram. **Figure 14.3** shows the CAN channel interrupt block diagram.

Table 14.173 List of CAN Interrupt Sources

	Interrupt Source		Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit
Global interrupts	Receive FIFO	Receive FIFO 0	RFIF in the RSCFDn(CFD)RFSTS0 register	RFIE in the RSCFDn(CFD)RFCC0 register
		Receive FIFO 1	RFIF in the RSCFDn(CFD)RFSTS1 register	RFIE in the RSCFDn(CFD)RFCC1 register
		Receive FIFO 2	RFIF in the RSCFDn(CFD)RFSTS2 register	RFIE in the RSCFDn(CFD)RFCC2 register
		Receive FIFO 3	RFIF in the RSCFDn(CFD)RFSTS3 register	RFIE in the RSCFDn(CFD)RFCC3 register
		Receive FIFO 4	RFIF in the RSCFDn(CFD)RFSTS4 register	RFIE in the RSCFDn(CFD)RFCC4 register
		Receive FIFO 5	RFIF in the RSCFDn(CFD)RFSTS5 register	RFIE in the RSCFDn(CFD)RFCC5 register
		Receive FIFO 6	RFIF in the RSCFDn(CFD)RFSTS6 register	RFIE in the RSCFDn(CFD)RFCC6 register
		Receive FIFO 7	RFIF in the RSCFDn(CFD)RFSTS7 register	RFIE in the RSCFDn(CFD)RFCC7 register
	Global error		DEF in the RSCFDn(CFD)GERFL register MES in the RSCFDn(CFD)GERFL register THLES in the RSCFDn(CFD)GERFL register CMPOF in the RSCFDnCFDGERFL register	DEIE in the RSCFDn(CFD)GCTR register MEIE in the RSCFDn(CFD)GCTR register THLEIE in the RSCFDn(CFD)GCTR register CMPOFIE in the RSCFDnCFDGCTR register
Channel interrupts (m = 0 to 3)	CANm transmit	CANm transmit complete	TMTRF[1:0] in the RSCFDn(CFD)TMSTSp register	TMIEp in the RSCFDn(CFD)TMIECy register
		CANm transmit abort	TMTRF[1:0] in the RSCFDn(CFD)TMSTSp register	TAIE in the RSCFDn(CFD)CmCTR register
		CANm transmit/ receive FIFO transmit complete	CFTXIF in the RSCFDn(CFD)CFSTSk register	CFTXIE in the RSCFDn(CFD)CFCCk register
		CANm transmit queue	TXQIF in the RSCFDn(CFD)TXQSTSm register	TXQIE in the RSCFDn(CFD)TXQCCm register
		CANm transmit history	THLIF in the RSCFDn(CFD)THLSTSm register	THLIE in the RSCFDn(CFD)THLCCm register
	CANm transmit/receive FIFO receive complete		CFRXIF in the RSCFDn(CFD)CFSTSk register	CFRXIE in the RSCFDn(CFD)CFCCk register
	CANm error		BEF in the RSCFDn(CFD)CmERFL register ALF in the RSCFDn(CFD)CmERFL register BLF in the RSCFDn(CFD)CmERFL register OVLF in the RSCFDn(CFD)CmERFL register BORF in the RSCFDn(CFD)CmERFL register BOEF in the RSCFDn(CFD)CmERFL register EPF in the RSCFDn(CFD)CmERFL register EWF in the RSCFDn(CFD)CmERFL register SOCO in the RSCFDnCFDCmFDSTS register EOCO in the RSCFDnCFDCmFDSTS register TDCVF in the RSCFDnCFDCmFDSTS register	BEIE in the RSCFDn(CFD)CmCTR register ALIE in the RSCFDn(CFD)CmCTR register BLIE in the RSCFDn(CFD)CmCTR register OLIE in the RSCFDn(CFD)CmCTR register BORIE in the RSCFDn(CFD)CmCTR register BOEIE in the RSCFDn(CFD)CmCTR register EPIE in the RSCFDn(CFD)CmCTR register EWIE in the RSCFDn(CFD)CmCTR register SOCOIE in the RSCFDnCFDCmCTR register EOCOIE in the RSCFDnCFDCmCTR register TDCVFIE in the RSCFDnCFDCmCTR register

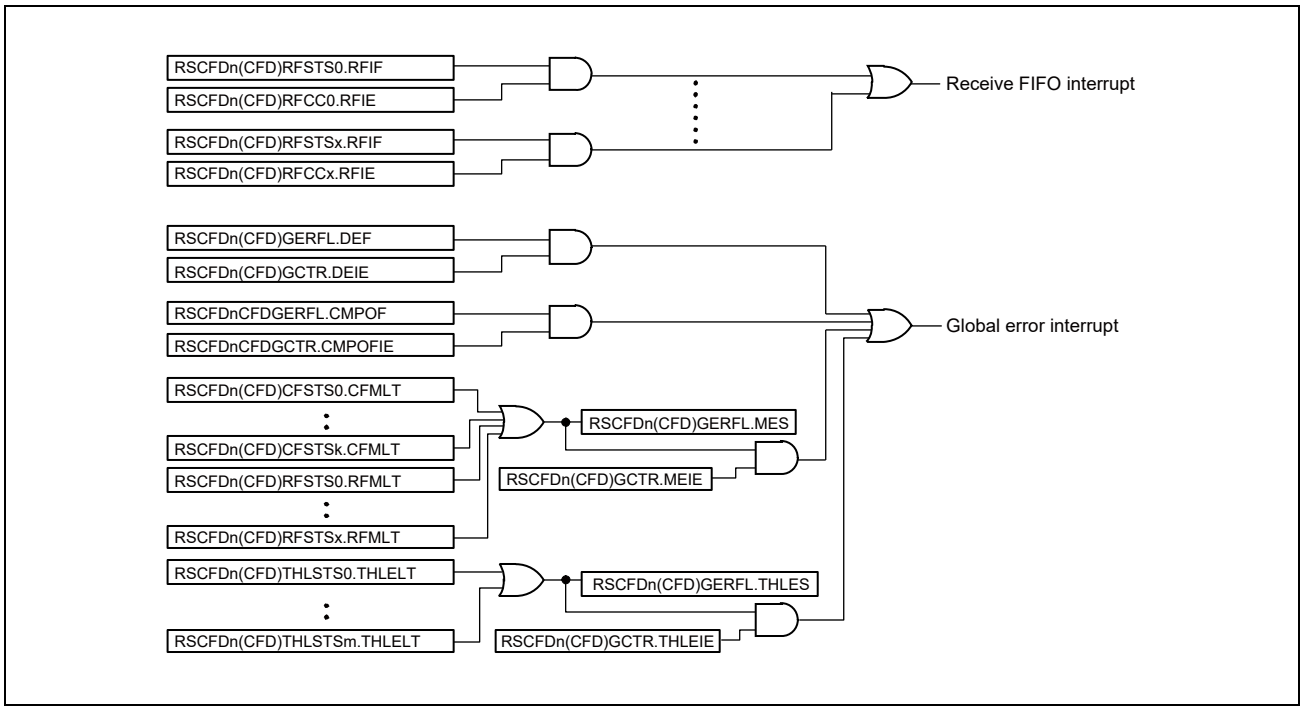


Figure 14.2 CAN Global Interrupt Block Diagram

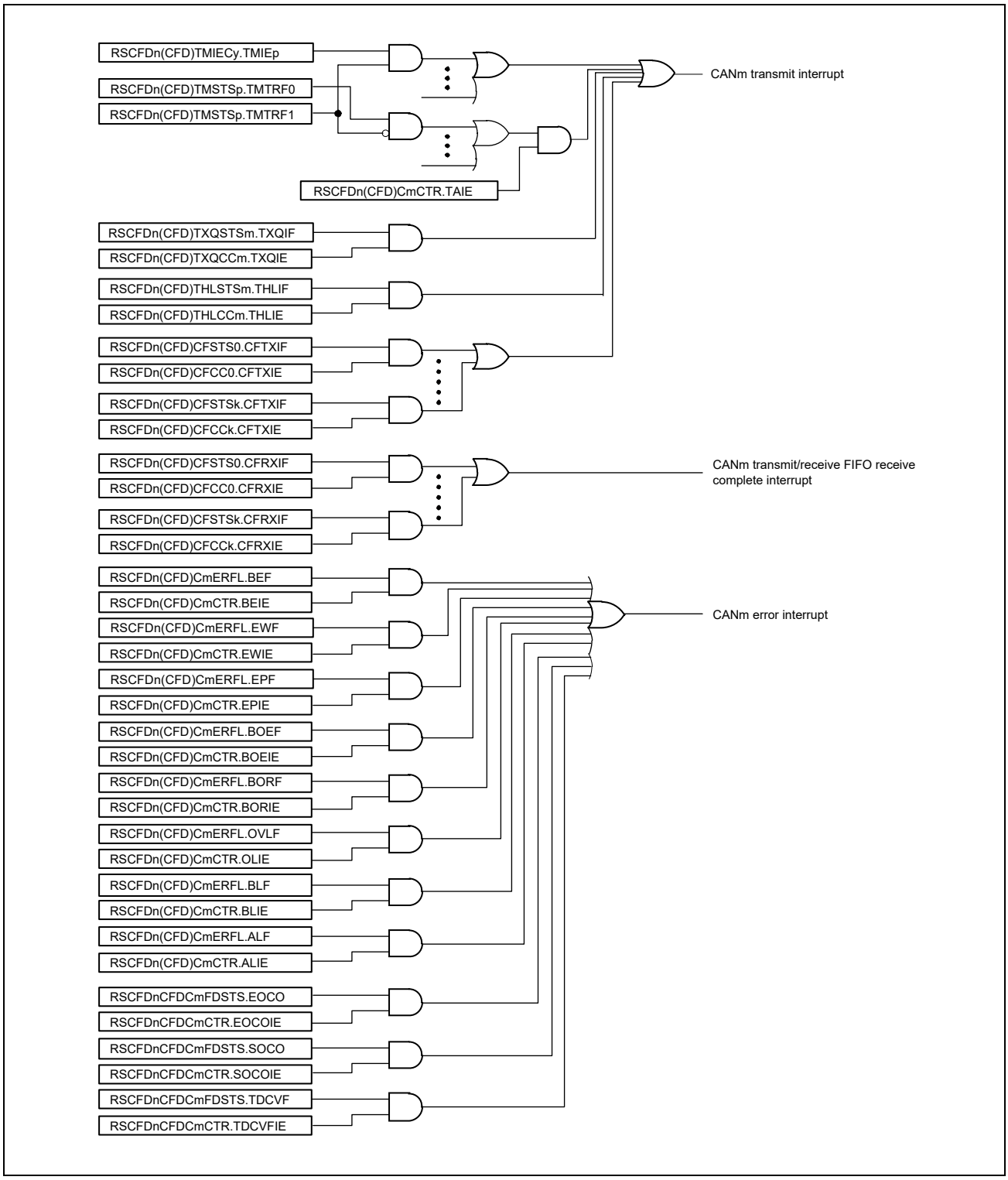


Figure 14.3 CAN Channel Interrupt Block Diagram

14.5.2 DMA Trigger (Only in CAN FD Mode)

In CAN FD mode, receive FIFO buffers can be related to DMA channels. The following 12 FIFO buffers can be related.

- All receive FIFO buffers x (x = 0 to 7)
- The first transmit/receive FIFO buffer k (k = 3 × m, m = 0 to 3) allocated to channel m

When the DMA enable bit (RFDMAEx or CFDMAEm bit in the RSCFDnCFDCDTCT register) is set to 1 and an unread message is remaining in the related FIFO, a DMA transfer request trigger is generated.

14.6 CAN Modes

The RS-CANFD module has four global modes to control the entire RS-CANFD module status and four channel modes to control individual channel status. Details of global modes are described in **Section 14.6.1, Global Modes**, and details of channel modes are described in **Section 14.6.2, Channel Modes**.

- Global stop mode:
  Stops the clocks of the entire module to achieve low power consumption.
- Global reset mode:
  Performs initial settings for the entire module.
- Global test mode:
  Performs test settings and performs the RAM test.
- Global operating mode:
  Makes the entire module operable.
- Channel stop mode:
  Stops the channel clock.
- Channel reset mode:
  Performs initial settings for the channels.
- Channel halt mode:
  Stops CAN communication and allows channel testing.
- Channel communication mode:
  Performs CAN communication.

14.6.1 Global Modes

Figure 14.4 shows the transitions of global modes.

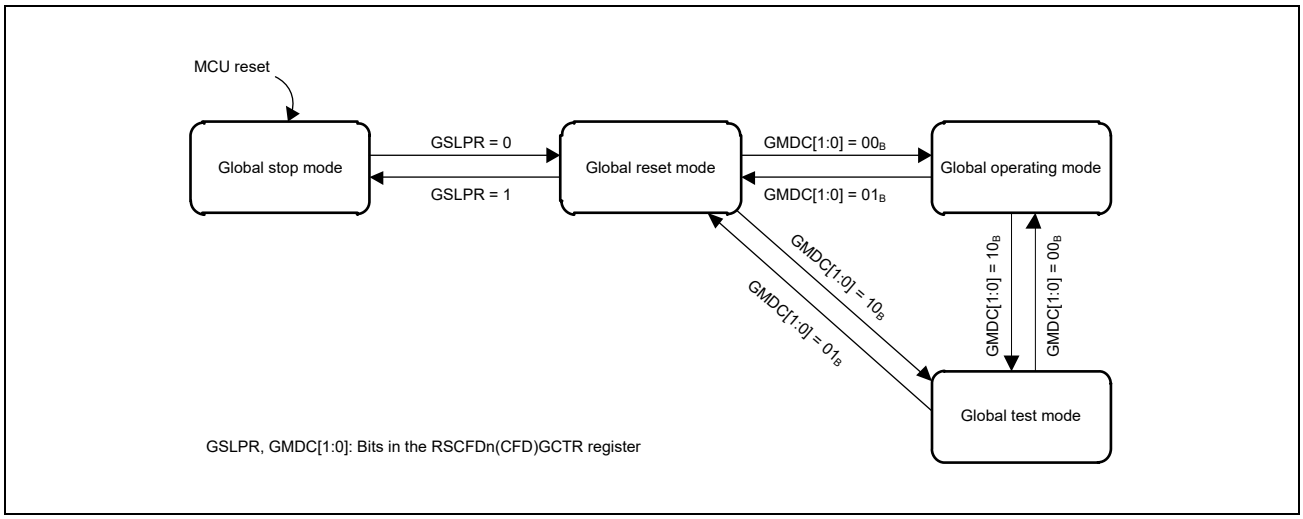


Figure 14.4 Transitions of Global Modes

In some cases, global mode transitions also force channel mode transitions. **Table 14.174** shows the channel mode transitions depending on the global mode setting dictated by the GMDC[1:0] bits and the GSLPR bit.

Table 14.174 Transitions of Channel Modes Depending on Global Mode Setting (GMDC[1:0] and GSLPR Bits)

	Channel Mode after Setting			
Channel Mode before Setting	GMDC[1:0] = 00_B GSLPR = 0 (Global Operation)	GMDC[1:0] = 10_B GSLPR = 0 (Global Test)	GMDC[1:0] = 01_B GSLPR = 0 (Global Reset)	GMDC[1:0] = 01_B GSLPR = 1 (Global Stop)
Channel communication	Channel communication	Channel halt	Channel reset	Transition prohibited
Channel halt	Channel halt	Channel halt	Channel reset	Transition prohibited
Channel reset	Channel reset	Channel reset	Channel reset	Channel stop
Channel stop	Channel stop	Channel stop	Channel stop	Channel stop

Note: GMDC[1:0], GSLPR: Bits in the RSCFDn(CFD)GCTR register

Table 14.175 shows the global mode transition time.

Table 14.175 Global Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Global stop	Global reset	Three pclk cycles
Global reset	Global stop	Three pclk cycles
Global reset	Global test	Ten pclk cycles
Global reset	Global operating	Ten pclk cycles
Global test	Global reset	Two CAN bit times*1 *2
Global test	Global operating	Three pclk cycles
Global operating	Global reset	Two CAN bit times*1 *2
Global operating	Global test	Two CAN frames*1

Note 1. CAN frame time and CAN bit time of the lowest communication speed of the channels in use

Note 2. In CAN FD mode, this time value is the CAN bit time of the nominal bit rate.

14.6.1.1 Global Stop Mode

In global stop mode, clocks of the CAN do not run and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained. Only the clock used by the CPU for writing to the GSLPR bit runs in this mode.

After the MCU is reset, the RS-CANFD module transitions to global stop mode. Setting the GSLPR bit in the RSCFDn(CFD)GCTR register to 1 (in global stop mode) in global reset mode sets the CSLPR bit in each of the RSCFDn(CFD)CmCTR register to 1 (channel stop mode). Afterwards, if all channels are forced to transition to channel stop mode, the RS-CANFD module transitions to global stop mode. The GSLPR bit should not be modified in global operating mode or global test mode.

14.6.1.2 Global Reset Mode

In global reset mode, RS-CANFD module settings are performed. When the RS-CANFD module transitions to global reset mode, some registers are initialized. For registers to be initialized, see **Table 14.178, Registers Initialized in Global Reset Mode or Channel Reset Mode** and **Table 14.179, Registers Initialized Only in Global Reset Mode**.

Setting the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register to 01_B sets the CHMDC[1:0] bits in each of the RSCFDn(CFD)CmCTR registers (m = 0 to 3) to 01_B (channel reset mode). If all channels are forced to transition to channel reset mode, the RS-CANFD module transitions to global reset mode. Channels that are already in channel reset mode or channel stop mode do not transition (because the CHMDC[1:0] bits have already been set to 01_B).

14.6.1.3 Global Test Mode

In global test mode, settings for test-related registers are performed. When the RS-CANFD module transitions to global test mode, all CAN communications are disabled.

Setting the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register to 10_B sets the CHMDC[1:0] bits in each of the RSCFDn(CFD)CmCTR register to 10_B (channel halt mode). If all channels are forced to transition to channel halt mode, the RS-CANFD module transitions to global test mode. Channels that are in channel stop mode, channel reset mode, or channel halt mode do not transition.

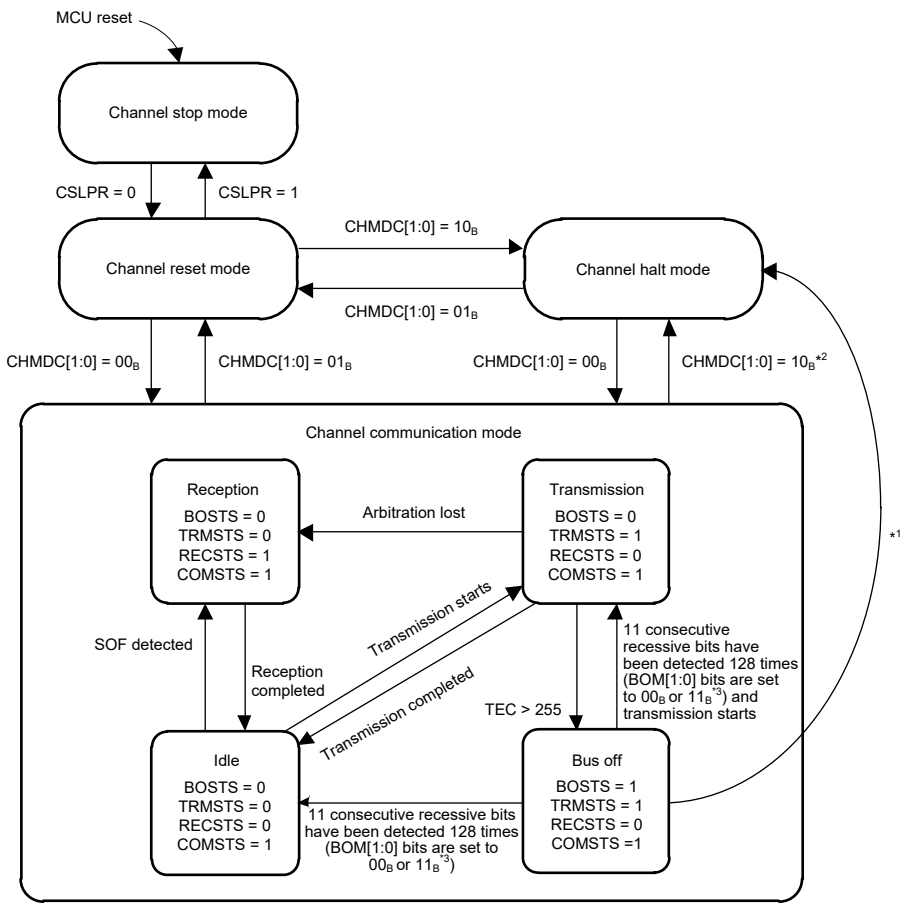
14.6.1.4 Global Operating Mode

The RS-CANFD module operates in global operating mode.

When the GMDC[1:0] bits in the RSCFDn(CFD)GCTR register are set to 00_B, the RS-CANFD module transitions to global operating mode.

14.6.2 Channel Modes

Figure 14.5 shows a channel mode state transition chart. **Table 14.176** shows the channel mode transition time.



Note: CHMDC[1:0], CSLPR, BOM[1:0]: Bits in the RSCFDn(CFD)CmCTR register (m = 0 to 3)
BOSTS, TRMSTS, RECSTS, COMSTS: Bits in the RSCFDn(CFD)CmSTS register

Note 1. Timing of transition from bus off state to channel halt mode
- When BOM[1:0] = 01_B: Transition to channel halt mode when TEC exceeds 255
- When BOM[1:0] = 10_B: Transition to channel halt mode when 11 consecutive recessive bits have been detected 128 times
- When BOM[1:0] = 11_B: Transition to channel halt mode when the CHMDC[1:0] bits are set to 10_B

Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode.

Note 3. When 11 consecutive recessive bits have been detected 128 times before the CHMDC[1:0] bits are set to 10_B

Figure 14.5 Channel Mode State Transition Chart

Table 14.176 Channel Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Channel stop	Channel reset	Three pclk cycles
Channel reset	Channel stop	Three pclk cycles
Channel reset	Channel halt	Three CANm bit times*1
Channel reset	Channel communication	Four CANm bit times*1
Channel halt	Channel reset	Two CANm bit times*1
Channel halt	Channel communication	Four CANm bit times*1
Channel communication	Channel reset	Two CANm bit times*1
Channel communication	Channel halt	Two CANm frames

Note 1. In CAN FD mode, this time value is the CANm bit time of the nominal bit rate.

14.6.2.1 Channel Stop Mode

In channel stop mode, clocks are not supplied to channels and therefore power consumption is reduced. Channel-related registers can be read, but writing data to them is prohibited (except for the CSLPR bit). Register values are retained.

Each channel enters channel stop mode after the MCU is reset. Channels also transition to channel stop mode when the CSLPR bit in the RSCFDn(CFD)CmCTR register (m = 0 to 3) is set to 1 (channel stop mode) in channel reset mode. The CSLPR bit should not be modified in channel communication mode and channel halt mode.

14.6.2.2 Channel Reset Mode

In channel reset mode, channel settings are performed. When a channel transitions to channel reset mode, some channel-related registers are initialized. For registers to be initialized, see **Table 14.178, Registers Initialized in Global Reset Mode or Channel Reset Mode**.

When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 01_B (channel reset mode) during CAN communication, communication is terminated before it is completed and the channel transitions to channel reset mode. **Table 14.177** shows the operation when the CHMDC[1:0] bits are set to 01_B (channel reset mode) during CAN communication.

14.6.2.3 Channel Halt Mode

In channel halt mode, settings for test-related registers of channels are performed. When a channel transitions to channel halt mode, CAN communication of the channel stops.

Table 14.177 shows operation when the CHMDC[1:0] bits are set to 10_B (channel halt mode) during CAN communication.

Table 14.177 Operation a Channel Transitions to Channel Reset Mode/Channel Halt Mode

Mode	During Reception	During Transmission	Bus Off State
Channel reset (CHMDC[1:0] = 01_B)	Transitions to channel reset mode before reception is completed.*1	Transitions to channel reset mode before transmission is completed.*1	Transitions to channel reset mode before bus off recovery.
Channel halt*3 (CHMDC[1:0] = 10_B)	Transitions to channel halt mode after reception is completed.*2	Transitions to channel halt mode after transmission is completed.	[When BOM[1:0] = 00_B] Transitions to channel halt mode (CHMDC[1:0] = 10_B) only after bus off recovery. [When BOM[1:0] = 01_B] Transitions to channel halt mode automatically when the condition for transition to bus off state is met. [When BOM[1:0] = 10_B] Transitions to channel halt mode automatically after bus off recovery. [When BOM[1:0] = 11_B] Transitions to channel halt mode immediately after the CHMDC[1:0] bits are set to 10_B before bus off recovery.

Note 1. To allow transition to channel reset mode after communication is completed, set the CHMDC[1:0] bits to 10_B and confirm that communication has been completed and transition to channel halt mode has been made, and then set the CHMDC[1:0] bits to 01_B.

Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode. The CAN bus status can be confirmed with the BLF flag of the RSCFDn(CFD)CmERFL register that becomes 1 when dominant lock is detected.

Note 3. In classical CAN mode, when the transition from channel reset mode to channel halt mode is to be made, set the RSCANnCmCFG register in channel reset mode and then shift to channel halt mode. In CAN FD mode, set the RSCFDnCFDCmNCFG register and the RSCFDnCFDCmDCFG register, and then make a transition.

14.6.2.4 Channel Communication Mode

In channel communication mode, CAN communication is performed. Each channel has the following communication states during CAN communication.

- Idle: Neither reception nor transmission is in progress.
- Reception: Receiving a message sent from another node.
- Transmission: Transmitting a message.
- Bus off: Isolated from CAN communication.

When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 00_B, the channel transitions to channel communication mode. After that, once 11 consecutive recessive bits have been detected, the COMSTS flag in the RSCFDn(CFD)CmSTS register (m = 0 to 3) is set to 1 (communication is ready) and transmission and reception are enabled on the CAN network as an active node. At this time, transmission and reception of messages can be started.

14.6.2.5 Bus Off State

A channel transitions to the bus off state according to the transmit/receive error counter increment/decrement rules of the CAN specifications.

The conditions for returning from the bus off state are determined by the BOM[1:0] bits in the RSCFDn(CFD)CmCTR register.

- When BOM[1:0] = 00_B:
  Bus off recovery is compliant with the CAN specifications. After 11 consecutive recessive bits have been detected 128 times, a channel returns from the bus off state to the CAN communication ready state (error active state). At that time, the TEC[7:0] and REC[7:0] bits in the RSCFDn(CFD)CmSTS register are initialized to 00_H, the BORF flag in the RSCFDn(CFD)CmERFL register is set to 1 (bus off recovery is detected), and a bus off recovery interrupt request is generated. When the CHMDC[1:0] bits in the RSCFDn(CFD)CmCTR register are set to 10_B (channel halt mode) in the bus off state, the channel transitions to channel halt mode after bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times).
- When BOM[1:0] = 01_B:
  When a channel transitions to the bus off state, the CHMDC[1:0] bits are set to 10_B and the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H. The BORF flag is not set to 1, and bus off recovery interrupt request is not generated.
- When BOM[1:0] = 10_B:
  When a channel has transitioned to the bus off state, the CHMDC[1:0] bits are set to 10_B. After bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times), the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H, the BORF flag is set to 1, and a bus off recovery interrupt request is generated.
- When BOM[1:0] = 11_B:
  When the CHMDC[1:0] bits are set to 10_B in the bus off state, the channel transitions to channel halt mode before bus off recovery is completed. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H, but the BORF flag is not set to 1. Also, a bus off recovery interrupt is not generated.
  However, the BORF flag becomes 1 and a bus off recovery interrupt request is generated if a RS-CANFD module transitions to error active state (by detecting 128 times of 11 consecutive recessive bits) before CHMDC[1:0] bits are set to 10_B.

If the RS-CANFD module causes the channel to transition to channel halt mode simultaneously with a program write to the CHMDC[1:0] bits, the program write takes precedence. An automatic transition to channel halt mode when the BOM[1:0] bits are set to 01_B or 10_B is made only when the CHMDC[1:0] bits are 00_B (channel communication mode).

Furthermore, setting the RTBO bit in the RSCFDn(CFD)CmCTR register to 1 allows a forced return from the bus off state. As soon as the RTBO bit is set to 1, the state changes to the error active state. After 11 consecutive recessive bits have been detected, the RS-CANFD module becomes ready for communication. In this case, the BORF flag is not set to 1 and the TEC[7:0] and REC[7:0] bits are initialized to 00_H. Write 1 to the RTBO bit only when the BOM[1:0] value is 00_B. Writing the RTBO bit to 1 in a state other than the bus off state is ignored, and the RTBO bit is immediately set to 0.

14.6.3 Initializing Registers by Transition to CAN Mode

Table 14.178 lists bits and flags to be initialized by a transition to channel reset mode. These bits and flags are also initialized by a transition to global reset mode. Furthermore, **Table 14.179** lists bits and flags to be initialized only by a transition to global reset mode.

Table 14.178 Registers Initialized in Global Reset Mode or Channel Reset Mode

Register	Bit / Flag
RSCFDn(CFD)CmCTR register	(ROM), CRCT, CTMS[1:0], CTME, CHMDC[1:0]
RSCFDn(CFD)CmSTS register	CHLTSTS, EPSTS, BOSTS, TRMSTS, RECSTS, COMSTS, (ESIF), REC[7:0], TEC[7:0]
RSCFDn(CFD)CmERFL register	CRCREG[14:0], ADERR, B0ERR, B1ERR, CERR, AERR, FERR, SERR, ALF, BLF, OVLF, BORF, BOEF, EPF, EWF, BEF
RSCFDnCFDCmFDCTR register	EOCCLR, SOCCLR
RSCFDnCFDCmFDSTS register	SOC[7:0], EOC[7:0], SOCO, EOCO, TDCVF, TDCR[6:0]
RSCFDnCFDCmFDCRC register	CRCREG[20:0], SCNT[3:0]
RSCFDn(CFD)CFCCk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFE
RSCFDn(CFD)CFSTSk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFMC[7:0], CFFLL, CFEMP, CFMLT, CFRXIF, CFTXIF
RSCFDn(CFD)CFTISTS register	CFkTXIF
RSCFDn(CFD)TMCp register	TMOM, TMTAR, TMTR
RSCFDn(CFD)TMSTSp register	TMTARM, TMTRM, TMTRF[1:0], TMTSTS
RSCFDn(CFD)TMTRSTSy register	TMTRSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTARSTSy register	TMTARSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTCSTSy register	TMTCSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TMTASTSy register	TMTASTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCFDn(CFD)TXQCCm register	TXQE
RSCFDn(CFD)TXQSTSm register	TXQIF, TXQFLL, TXQEMP
RSCFDn(CFD)THLCCm register	THLE
RSCFDn(CFD)THLSTSm register	THLMC[4:0], THLIF, THLELT, THLFLL, THLEMP
RSCFDn(CFD)GTINTSTS0 register	TSIFm, TAIFm, TQIFm, CFTIFm, THIFm (m = 0 to 3)

Note: Bits and flags in parentheses exist only in registers in CAN FD mode.

Table 14.179 Registers Initialized Only in Global Reset Mode

Register	Bit / Flag
RSCFDn(CFD)GSTS register	GHLTSTS
RSCFDn(CFD)GERFL register	EEF0, EEF1, EEF2, EEF3, (CMPOF), THLES, MES, DEF
RSCFDn(CFD)GTSC register	TS[15:0]
RSCFDn(CFD)RMNDy register	RMNSq
RSCFDn(CFD)RFCCx register	RFE
RSCFDn(CFD)RFSTSx register	RFMC[7:0], RFIF, RFMLT, RFFLL, RFEMP
RSCFDn(CFD)CFCCk register	When transmit/receive FIFO buffer is in receive mode: CFE
RSCFDn(CFD)CFSTSk register	When transmit/receive FIFO buffer is in receive mode: CFMC[7:0], CFFLL, CFEMP, CFTXIF, CFRXIF, CFMLT
RSCFDn(CFD)FESTS register	CFkEMP, RFxEMP
RSCFDn(CFD)FFSTS register	CFkFLL, RFxFLL
RSCFDn(CFD)FMSTS register	CFkMLT, RFxMLT
RSCFDn(CFD)RFISTS register	RFxIF
RSCFDn(CFD)CFRISTS register	CFkRXIF
RSCFDnCFDCDTCT register	CFDMAEm, RFDMAEx
RSCFDnCFDCDTSTS register	CFDMASTSm, RFDMASTSx
RSCFDn(CFD)GTSTCFG register	RTMPS[6:0], C0ICBCE, C1ICBCE, C2ICBCE, C3ICBCE
RSCFDn(CFD)GTSTCTR register	RTME, ICBCTME

Note: Bits and flags in parentheses exist only in registers in CAN FD mode.

14.7 Reception Function

There are two reception types.

- Reception by receive buffers:
  0 to 64 receive buffers can be shared by all channels. Since messages stored in receive buffers are overwritten at each reception, the latest receive data can always be read.
- Reception by receive FIFO buffers and transmit/receive FIFO buffers (receive mode):
  Eight receive FIFO buffers can be shared by all channels and three dedicated transmit/receive FIFO buffers are provided for each channel. Messages of up to the number of buffer stages specified with the RFDC[2:0] and CFDC[2:0] bits can be stored in FIFO buffers and can be read sequentially from the oldest.

14.7.1 Data Processing Using the Receive Rule Table

Data processing using the receive rule table allows dispatching of selected messages to the specified buffer. Data processing includes acceptance filter processing, DLC filter processing, routing processing, label addition processing, and mirror function processing.

Up to 128 receive rules can be registered per channel and up to (64 × number of channels) total receive rules can be registered in the entire module. (Up to 256 receive rules can be registered in this module that has 4 channels.) Set receive rules for each channel. Receive rules cannot be shared with other channels. If receive rules are not set, no messages can be received. **Figure 14.6** illustrates how receive rules are registered.

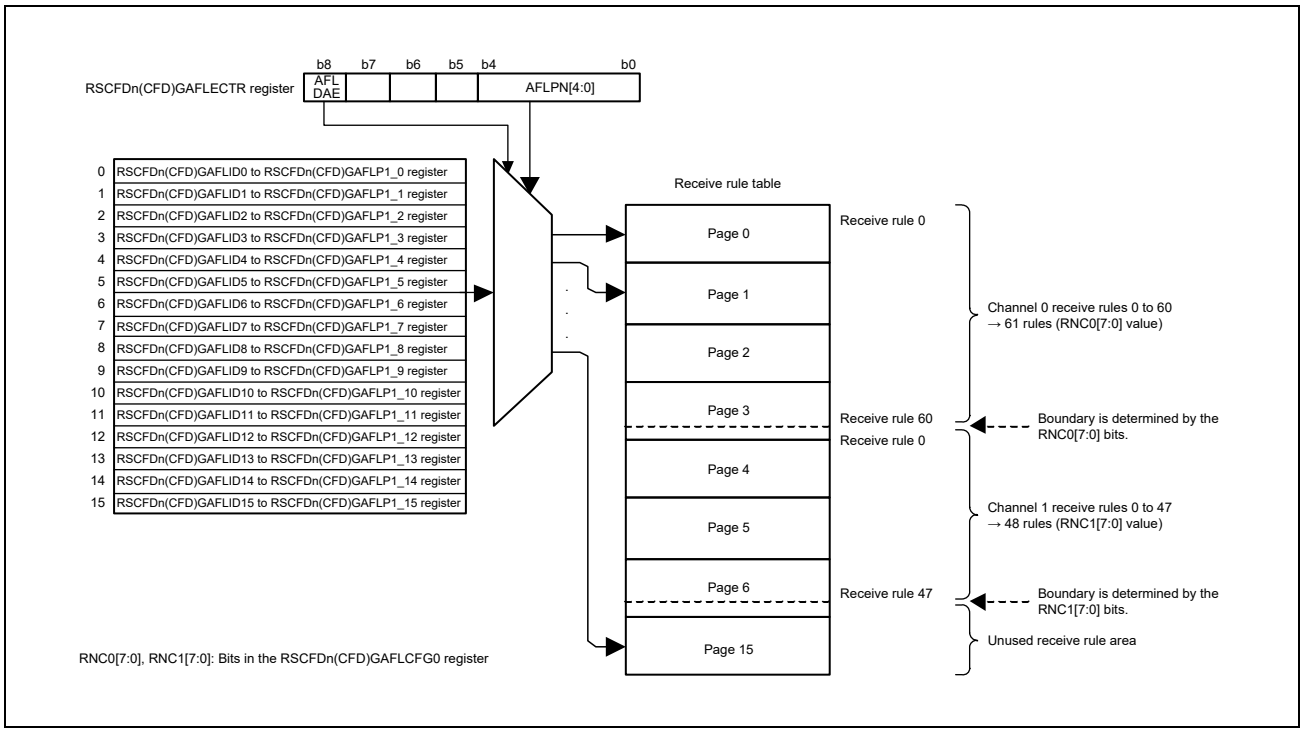


Figure 14.6 Entry of Receive Rules (for Setting Channel 0 and 1)

CAUTION

Receive rules for each channel must be set in contiguous blocks. Channel 1 rules and channel 0 rules must be set separately.

Each receive rule consists of 16 bytes in the RSCFDn(CFD)GAFLIDj, RSCFDn(CFD)GAFLMj, RSCFDn(CFD)GAFLP0_j, and RSCFDn(CFD)GAFLP1_j registers (j = 0 to 15). The RSCFDn(CFD)GAFLIDj register is used to set GAFLID, GAFLIDE bit, GAFLRTR bit, and the mirror function, the RSCFDn(CFD)GAFLMj register is used to set mask, the RSCFDn(CFD)GAFLP0_j register is used to set label information to be added, DLC value, and storage receive buffer, and the RSCFDn(CFD)GAFLP1_j register is used to set storage FIFO buffer. Up to 16 receive rules can be set per page.

14.7.1.1 Acceptance Filter Processing

In the acceptance filter processing, the ID data, IDE bit, and RTR bit in a received message are compared with the ID data, IDE bit, and RTR bit set in the receive rule of the corresponding channel. When all these bits match, the message passes through the acceptance filter processing. The ID data, IDE bit, and RTR bit in the received message which correspond to the bits set to 0 (bits are not compared) in the RSCFDn(CFD)GAFLMj register are not compared and are regarded as matched.

Check begins with the receive rule of the minimum number for the corresponding channel. When all the bits to be compared in a received message match the bits set in the receive rule or when all the receive rules are compared without any match, filter processing stops. If there is no matching receive rule, the received message is not stored in the receive buffer or FIFO buffer.

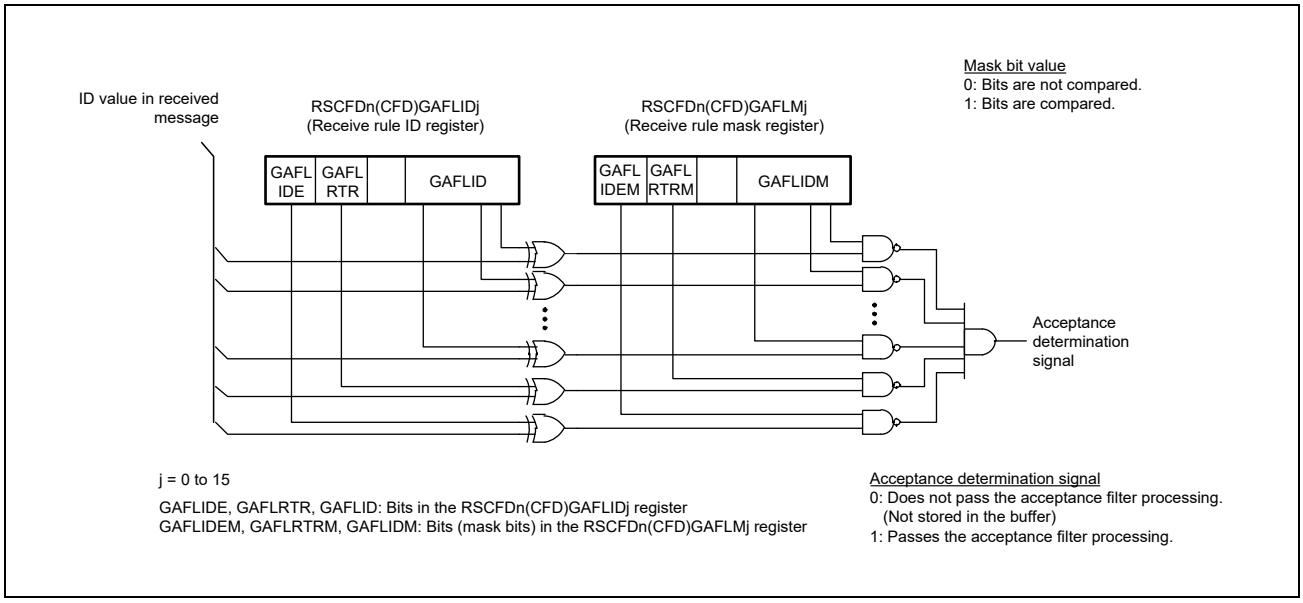


Figure 14.7 Acceptance Filter Function

14.7.1.2 DLC Filter Processing

When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), DLC filter processing is added to messages that passed through the acceptance filter processing. When the DLC value in a message is equal to or larger than the DLC value set in the receive rule, the message passes through the DLC filter processing.

When a message has passed through the DLC filter processing with the DRE bit in the RSCFDn(CFD)GCFG register set to 0 (DLC replacement is disabled), the DLC value in the received message is stored in the buffer. In this case, all the data bytes in the received message are stored in the buffer.

When a message has passed through the DLC filter processing with the DRE bit in the RSCFDn(CFD)GCFG register set to 1 (DLC replacement is enabled), the DLC value in the receive rule is stored in the buffer instead of the DLC value in the received message. In this case, a value of 00_H is stored in each data byte beyond the number of bytes which is indicated by the DLC value in the receive rule.

When the DLC value in the received message is smaller than that in the receive rule, the message does not pass through the DLC filter processing. In this case, the message is not stored in the receive buffer or the FIFO buffer and the DEF flag in the RSCFDn(CFD)GERFL register is set to 1 (a DLC error is present).

14.7.1.3 Routing Processing

Messages that passed through the acceptance filter processing and the DLC filter processing are stored in receive buffers, receive FIFO buffers, or transmit/receive FIFO buffers (set to receive mode or gateway mode). Message storage destination is set by the GAFLRMV and GAFLRMDP[6:0] bits in the RSCFDn(CFD)GAFLP0_j register (j = 0 to 15) and by the RSCFDn(CFD)GAFLP1_j register. Messages that passed through the acceptance filter processing and the DLC filter processing can be stored in up to eight buffers.

In CAN FD mode, if the payload length of the received message exceeds the payload storage size of the storage buffer, the CMPOF flag in the RSCFDnCFDGERFL register is set to 1 (payload overflow) and the processing is handled according to the CMPOC bit in the RSCFDnCFDGCFG register. When the CMPOC bit is 0, a received message with a payload exceeding the available space for payload storage is not stored in the buffer. When the CMPOC bit is 1, the received message with the payload truncated to fit the available space and the value of the DLC field, either that in the received message or from the reception rule table, depending on the value of the DRE bit in the RSCFDnCFDGCFG register, are stored in the buffer.

14.7.1.4 Label Addition Processing

It is possible to add 12-bit label information to messages that passed through the filter processing and store them in buffers. This label information is set in the GAFLPTR[11:0] bits in the RSCFDn(CFD)GAFLP0_j register.

14.7.1.5 Mirror Function Processing

The mirror function allows the CAN node to receive its own transmitted messages. The mirror function is made available by setting the MME bit in the RSCFDn(CFD)GCFG register to 1 (mirror function is enabled).

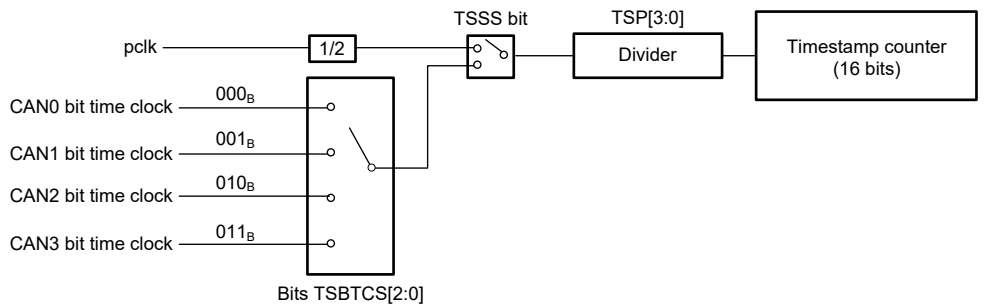
When the mirror function is in use, receive rules for which the GAFLLB bit in the RSCFDn(CFD)GAFLIDj register is set to 0 are used for data processing when receiving messages transmitted from other CAN nodes. When the CAN node is receiving its own transmitted messages, receive rules for which the GAFLLB bit is set to 1 are used for data processing.

14.7.1.6 Timestamp

The timestamp counter is a 16-bit free-running counter used for recording message reception time and transmission time. The timestamp counter value is fetched at the timing set with the TSCCFG[1:0] bits in the RSCFDn(CFD)GFDCFG register and is then stored in a receive buffer or a FIFO buffer together with the message ID and data during data reception. The clock source of the timestamp counter is selected by the TSBTCS[2:0] and TSSS bits in the RSCFDn(CFD)GCFG register. In classical CAN mode, either pclk/2 or the CANm bit time clock (m = 0 to 3). In CAN FD mode, the clock source is selectable from pclk/2 or nominal CANm bit time clock. However, do not select the nominal CANm bit time clock of channels that handle CAN FD frames. The timestamp counter count source is obtained by dividing the selected clock source by the TSP[3:0] value in the RSCFDn(CFD)GCFG register.

When the CANm bit time clock or nominal CANm bit time clock is used as a clock source, the timestamp counter stops when the corresponding channel transitions to channel reset mode or channel halt mode. When the pclk/2 is used as a clock source, the timestamp function is not affected by channel mode.

The timestamp counter value is reset to 0000_H by setting the TSRST bit in the RSCFDn(CFD)GCTR register to 1.



TSBTCS[2:0], TSSS, TSP[3:0]: Bits in the RSCFDn(CFD)GCFG register

Note: When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000_B.

Figure 14.8 Timestamp Function Block Diagram

14.8 Transmission Functions

There are three types of transmission. In classical CAN mode, transmittable payload length is 8 bytes in every transmission type. In CAN FD mode, transmittable payload length varies with transmission types.

- Transmission using transmit buffers:
  Each channel has 16 buffers. Transmittable payload length in CAN FD mode is 20 bytes. However, when transmit buffer merge mode is used, four buffers out of 16 buffers are allocated as a payload-only storage area and two buffers are able to transmit payloads with a length of more than 20 bytes.
- Transmission using transmit/receive FIFO buffers (transmit mode):
  Each channel has three FIFO buffers. Up to 128 messages can be contained in a single FIFO buffer. Transmittable payload length in CAN FD mode is 64 bytes. Each FIFO buffer is used with a link to a transmit buffer. Only the message to be transmitted next in a FIFO buffer becomes the target of transmit priority determination. Messages are transmitted sequentially on a first-in, first-out basis.
- Transmission using transmit queues:
  Up to 16 transmit buffers per channel can be allocated to the transmit queues. Transmittable payload length in CAN FD mode is 20 bytes. Transmit buffer ((16 × m) + 15) is used as an access window of a corresponding channel. Transmit buffers are allocated to transmit queues in descending order of buffer number. All messages in transmit queues, which are targets of priority determination, are transmitted in the order of ID number.

Figure 14.9 shows the allocation of transmit queues and transmit/receive FIFO buffer link.

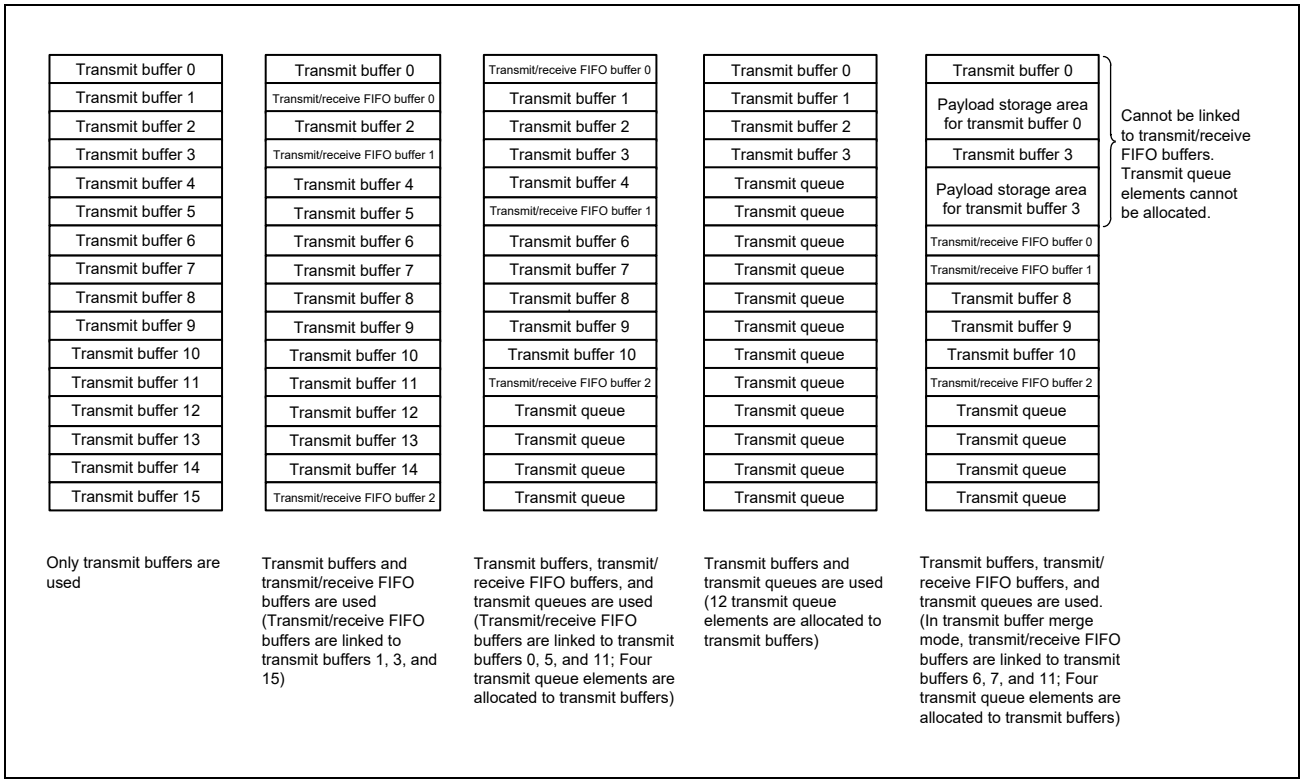


Figure 14.9 Allocation of Transmit Queues and Transmit/Receive FIFO Buffer Links

14.8.1 Transmit Priority Determination

If transmit requests are issued from multiple buffers or from the queue on the same channel, transmit priority is determined using one of the following methods.

The priority is determined by using one of the following methods.

- ID priority (TPRI bit = 0)
- Transmit buffer number priority (TPRI bit = 1)

All CAN channels use the setting of the TPRI bit in the RSCFDn(CFD)GCFG register.

When the TPRI bit is set to 0, messages are transmitted according to the priority of stored message IDs. ID priority conforms to the CAN bus arbitration specification defined in the CAN specifications. All IDs of pending transmit messages are targets of priority determination, regardless of whether they are stored in transmit buffers, transmit/receive FIFO buffers (set to transmit mode or gateway mode), or the transmit queue. If even a single transmit queue is used, select ID priority. When transmit/receive FIFO buffers are used, the oldest message in a FIFO buffer becomes the target of priority determination. When a message is being transmitted from a transmit/receive FIFO buffer, the next message in the FIFO buffer becomes the target of priority determination. When a transmit queue is used, all messages in the transmit queue are targets of priority determination. If the same ID is set for two or more buffers, the buffer with the smaller buffer number takes precedence.

When the TPRI bit is set to 1, the message in the transmit buffer with the minimum buffer number among all buffers with a transmit request is transmitted first. When transmit/receive FIFO buffers are linked to transmit buffers, transmit priority is determined according to linked transmit buffer numbers.

When messages are retransmitted due to an arbitration-lost or an error, transmit priority determination is made again regardless of the TPRI bit. When a 2-bit ECC error is detected in the priority determination processing, no message is transmitted (only when the EEFE bit in the RSCANnGCFG register is 1 in classical CAN mode).

14.8.2 Transmission Using Transmit Buffers

Setting the transmit request bit (TMTR bit in the RSCFDn(CFD)TMCp register) in a transmit buffer to 1 (transmission is requested) allows transmission of data frames or remote frames.

The transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCFDn(CFD)TMSTSp register (p = 0 to 63). When transmit completes successfully, the TMTRF[1:0] flag is set to 10_B (transmission has been completed (without transmit abort request)) or 11_B (transmission has been completed (with transmit abort request)).

14.8.2.1 Transmit Abort Function

With respect to transmit buffers for which the TMTRM bit in the RSCFDn(CFD)TMSTSp register is set to 1 (a transmit request is present), when the TMTAR bit in the RSCFDn(CFD)TMCp register is set to 1 (transmit abort is requested), the transmit request is canceled. When transmit abort is completed, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSp register is set to 01_B (transmit abort has been completed) and the transmit request is canceled (clearing the TMTRM bit to 0).

A message that is being transmitted or a message to be transmitted next according to the transmit priority determination cannot be aborted. However, when an arbitration-lost or an error occurs during transmission of a message for which the TMTAR bit is set to 1, retransmission is not performed.

14.8.2.2 One-Shot Transmission Function (Retransmission Disabling Function)

When the TMOM bit in the RSCFDn(CFD)TMCp register is set to 1 (one-shot transmission is enabled), transmission is performed only once. Even if an arbitration-lost or an error occurs, retransmission is not performed.

The one-shot transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCFDn(CFD)TMSTSp register. When one-shot transmission completes successfully, the TMTRF[1:0] flag is set to 10_B or 11_B. When an arbitration-lost or an error occurs, the TMTRF[1:0] flag is set to 01_B (transmit abort has been completed).

14.8.2.3 Transmit Buffer Merge Mode (Only in CAN FD Mode)

Transmit buffers can transmit messages with a payload length of 20 bytes, but can transmit messages with a payload length of up to 64 bytes by merging three transmit buffers in transmit buffer merge mode.

Setting the TMME bit to 1 in the RSCFDnCFDCmFDCFG register enables transmit buffer merge mode. In this mode, six buffers per channel become a merge area and two sets of transmit buffers (16 × m) + 0 to (16 × m) + 2 and transmit buffers (16 × m) + 3 to (16 × m) + 5 are merged. A transmission request is made by the first transmit buffer, and subsequent two buffers are used as a payload storage area. Do not set the transmission request bit (TMTR bit in the RSCFDnCFDTMCp register) and the transmission abort request bit (TMTAR bit in the RSCFDnCFDTMCp register) to 1 for transmit buffers except for the first buffer.

While transmit buffer merge mode is enabled, do not link the transmit/receive FIFO buffer to six merged buffers or allocate it to the transmit queue.

14.8.3 Transmission Using FIFO Buffers

Multiple messages can be stored in a single transmit/receive FIFO buffers, up to the number specified by the FIFO buffer depth, which is set by the CFDC[2:0] bits in the RSCFDn(CFD)CFCCk register (k = 0 to 11). Messages are transmitted sequentially on a first-in, first-out basis.

Each transmit/receive FIFO buffer is linked to a transmit buffer selected by the CFTML[3:0] bits in the RSCFDn(CFD)CFCCk register. When the CFE bit in the RSCFDn(CFD)CFCCk register is set to 1 (transmit/receive FIFO buffers are used), transmit/receive FIFO buffers become targets of transmit priority determination. Priority of only the next transmit message is determined in the FIFO buffer.

When the CFE bit is set to 0 (no transmit/receive FIFO buffer is used), the CFEMP flag is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) at the timing below.

- The transmit/receive FIFO buffer becomes empty immediately if the message in it is not being transmitted or is not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, arbitration-lost or the transition to channel halt mode in the case that a message in it is being transmitted or to be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages cannot be stored in FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

14.8.3.1 Interval Transmission Function

A message transmission interval time can be set to space the transmission of messages from the same FIFO buffer when using a transmit/receive FIFO buffer set to transmit mode or gateway mode.

Immediately after the first message has been transmitted successfully from the FIFO buffer with the CFE bit in the RSCFDn(CFD)CFCCk register set to 1, the interval timer starts counting (after EOF7 of the CAN protocol). After that, when the interval time has passed, the next message is transmitted. The interval timer stops in channel reset mode or by clearing the CFE bit to 0.

The interval time is set by the CFITT[7:0] bits in the RSCFDn(CFD)CFCCk register. When the interval timer is not used, set the CFITT[7:0] bits to 00_H.

Select an interval timer count source using the CFITR and CFITSS bits in the RSCFDn(CFD)CFCCk register. When the CFITR and CFITSS bits are set to 00_B, the count source is obtained by dividing pclk/2 by the value of the ITRCP[15:0] bits. When the CFITR and CFITSS bits are set to 10_B, the count source is obtained by dividing pclk/2 by (the value of the ITRCP[15:0] bits in the RSCFDn(CFD)GCFG register × 10). When the CFITR and CFITSS bits are set to $x1_B$, the CANm bit time clock becomes a count source in classical CAN mode and the nominal CANm bit time clock becomes a count source in CAN FD mode.

The interval time is calculated by the following equations where M is the value of ITRCP[15:0] and N is the value of CFITT[7:0].

- When CFITR and CFITSS = 00_B:

$$\frac{1}{\text{pclk frequency}} \times 2 \times M \times N$$

- When CFITR and CFITSS = 10_B:

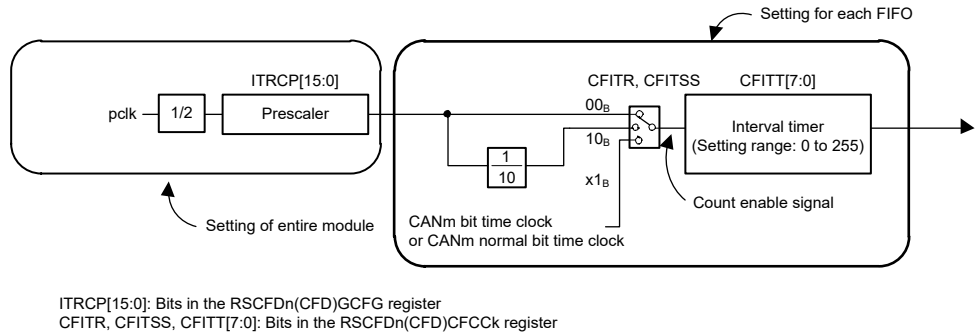
$$\frac{1}{\text{pclk frequency}} \times 2 \times M \times 10 \times N$$

- When CFITR and CFITSS = $x1_B$:

$$\text{Classical CAN mode} : \frac{1}{\text{CANm bit time clock frequency}} \times N$$

$$\text{CAN FD mode} : \frac{1}{\text{Nominal CANm bit time clock frequency}} \times N$$

Figure 14.10 shows the interval timer block diagram.



ITRCP[15:0]: Bits in the RSCFDn(CFD)GCFG register
CFITR, CFITSS, CFITT[7:0]: Bits in the RSCFDn(CFD)CFCCk register

Note: m = 0 to 3, k = 0 to 11

Figure 14.10 Interval Timer Block Diagram

Figure 14.11 shows the interval timer timing diagram.

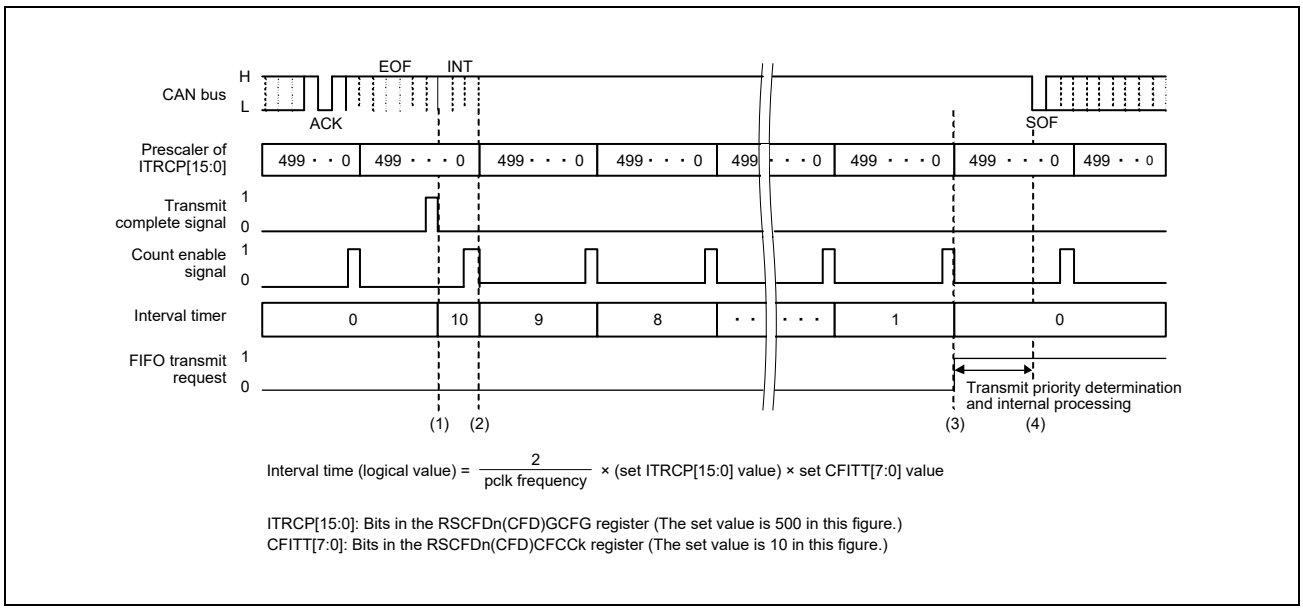


Figure 14.11 Interval Timer Timing Chart

(1) The interval timer starts counting upon completion of transmission. Since the prescaler is not initialized at the time of transmission completion, the first interval time contains an error of up to one count of the interval timer.

(2) The interval timer is decremented by the next count enable signal.

(3) When the interval timer has decreased to 0, the transmit/receive FIFO buffer issues a transmit request.

(4) The transmit/receive FIFO buffer is determined for the next transmission by the priority determination, it starts transmitting data. Transmission starts usually with a delay of three CANm bit time clock cycles or less from the issue of transmit request. If multiple internal processes (such as receive filter processing, message routing, and transmit priority determination) take place in all channels, a delay of up to 1376 cycles of the pclk may be generated.

14.8.4 Transmission Using Transmit Queues

Three to sixteen buffers (up to 10 buffers in transmit buffer merge mode) are allocated to a transmit queue for each channel, and transmit buffer ((16 × m) + 15) is used as an access window of a corresponding channel.

All messages in a transmit queue are targets of transmit priority determination and are transmitted in the ID priority order regardless of storage sequence. If two messages having the same ID are stored in a transmit queue, these messages are not always transmitted in the order of their storage in the transmit queue.

Setting the TXQE bit in the RSCFDn(CFD)TXQCCm register to 0 disables transmit queues. When the TXQE bit is set to 0, the TXQEMP flag in the RSCFDn(CFD)TXQSTSm register is set to 1 (the transmit queue contains no messages (transmit queue empty)) at the timing below.

- The transmit queue becomes empty immediately when no message in it is being transmitted or will be transmitted next.
- The transmit queue becomes empty after transmission completion, CAN bus error detection, arbitration-lost, or the transition to channel halt mode when a message in it is being transmitted or will be transmitted next.

When the TXQE bit is cleared to 0, all messages in transmit queues are lost and messages cannot be stored in transmit queues. Confirm that the TXQEMP flag is set to 1 before setting the TXQE bit to 1 again.

14.8.5 Transmit Data Padding (Only in CAN FD Mode)

When the payload length indicated by the set DLC value in a transmit message exceeds the payload storage area size of a buffer to be used for transmission, excessive payloads are padded by CC_H.

This processing is performed in the following cases when the transmit buffer merge mode is disabled (TMME bit in the RSCFDnCFDCmFDCFG register is 0).

- Transmit/receive FIFO set to transmission or gateway mode:
  When the payload length of the transmit DLC exceeds the transmit/receive FIFO payload storage area size set by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register
- Transmit buffer (including transmit queue):
  When the payload length of the transmit DLC exceeds 20 bytes

When the transmit buffer merge mode is enabled, no transmit data is padded in any transmission using a transmit buffer, transmit/receive FIFO buffer, or transmit queue. At this time, do not set a payload length more than the payload storage size of the buffer for transmitting as the DLC value in the transmit message.

14.8.6 Transmit History Function

Information about transmission-completed messages can be stored in the transmit history buffer. Each channel has a single transmit history buffer that can contain 16 sets of transmit history data.

A message transmit source buffer type can be selected by the THLDTE bit in the RSCFDn(CFD)THLCCm register. The THLEN bit in the RSCFDn(CFD)CFIDk register (k = 0 to 11) determines whether transmit history data is stored for each message. In classical CAN mode, the TMTSCE bit in the RSCANnGCFG register can be used to set whether to include a timestamp value in the transmit history data. In CAN FD mode, a timestamp value is always included.

The following information on a transmitted message will be stored in the transmit history buffer after the successful completion of transmission.

Storage of the transmit history data after the successful completion of transmission may take up to 152 cycles of pclk in classical CAN mode or 420 cycles of pclk in CAN FD mode.

- Buffer type
  001_B: Transmit buffer
  010_B: Transmit/receive FIFO buffer
  100_B: Transmit queue
- Buffer number
  Number of source transmit buffer, transmit queue, or transmit/receive FIFO buffer. This number depends on buffer types. See **Table 14.180**.
- Label data
  Label information of the transmit message
- Timestamp
  Timestamp value of the transmit message
  (When the TMTSCE bit is 1 in classical CAN mode)

Table 14.180 Transmit History Data Buffer Numbers

Buffer type / Buffer No.	001_B	010_B	100_B
0000_B	Transmit buffer 16 × m + 0	Buffer numbers of the transmit buffer linked to the transmit/receive FIFO buffer by the CFTML[3:0] bits in the RSCFDn(CFD)CFCCk register (k = 0 to 11)	Buffer numbers of the transmit buffer allocated to the transmit queue that performed transmission
0001_B	Transmit buffer 16 × m + 1		
0010_B	Transmit buffer 16 × m + 2		
0011_B	Transmit buffer 16 × m + 3		
0100_B	Transmit buffer 16 × m + 4		
0101_B	Transmit buffer 16 × m + 5		
0110_B	Transmit buffer 16 × m + 6		
0111_B	Transmit buffer 16 × m + 7		
1000_B	Transmit buffer 16 × m + 8		
1001_B	Transmit buffer 16 × m + 9		
1010_B	Transmit buffer 16 × m + 10		
1011_B	Transmit buffer 16 × m + 11		
1100_B	Transmit buffer 16 × m + 12		
1101_B	Transmit buffer 16 × m + 13		
1110_B	Transmit buffer 16 × m + 14		
1111_B	Transmit buffer 16 × m + 15		

Label data is used to identify each message. Unique label data can be added to each message transmitted from a transmit buffer, transmit queue, or transmit/receive FIFO buffer.

The timestamp value is fetched from the timestamp counter at the SOF (start of frame) timing of the message. For details about the timestamp counter, see **Section 14.7.1.6, Timestamp**.

Transmit history data can be read from the RSCFDn(CFD)THLACCm register. If an attempt is made to store new transmit history data while the buffer is full, the buffer overflows and the new data is discarded.

14.9 Gateway Function

When a transmit/receive FIFO buffer is set to gateway mode, receive messages can be transmitted from an arbitrary channel without CPU intervention.

When the CFM[1:0] bits in the RSCFDn(CFD)CFCCk register are set to 10_B (gateway mode) for the transmit/receive FIFO buffer selected by the RSCFDn(CFD)GAFLP1_j register of a channel being used for transmission, messages that pass through filter processing according to the receive rule are stored in the specified transmit/receive FIFO buffer and are automatically transmitted from the buffer.

Messages stored in a transmit/receive FIFO buffer are transmitted sequentially on a first-in, first-out basis. Only the message to be transmitted next becomes the target of transmit priority determination.

Transmit/receive FIFO buffers in the gateway mode are disabled by setting the CFE bit in the RSCFDn(CFD)CFCCk register to 0 and the CFEMP flag becomes 1 according to the timing below.

- The transmit/receive FIFO buffer becomes empty immediately when the message in it is not being transmitted and will not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when the message in it is being transmitted or will be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages can no longer be stored in transmit/receive FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

14.9.1 CAN-CAN FD Gateway (Only in CAN FD Mode)

When the gateway function is used in CAN FD mode, a frame to be transmitted can be replaced with a classical CAN frame or a CAN FD frame.

Setting the GWEN bit in the RSCFDnCFDCmFDCFG register to 1 enables the CAN-CAN FD gateway. The FDF and BRS bits in the transmit frame can be selected by the GWFDF and GWBRS bits in the RSCFDnCFDCmFDCFG register. When the DLC value of the received CAN frame is 1001_B or more and the GWFDF bit is 1 (CAN FD frame), the DLC value is replaced with 1000_B.

When the CAN-CAN FD gateway is enabled, do not perform routing for the following frames.

- CAN FD frames with a payload length of more than 8 bytes
- Remote frames

When the CAN-CAN FD gateway is enabled, the following frame shoudl be transmitted in the channel by setting of GWFDF.

- When GWFDF bit is set to 0, only classical CAN frame should be transmitted.
- When GWFDF bit is set to 1, only CAN FD frame should be transmitted.

14.10 Test Function

The test function is classified into communication tests and global tests.

- Communication tests: Performed for each channel.
  - Standard test mode
  - Listen-only mode
  - Self-test mode 0 (external loopback mode)
  - Self-test mode 1 (internal loopback mode)
  - Restricted operation mode (only in CAN FD mode)
- Global tests: Performed for the entire module
  - RAM test (read/write test)
  - Inter-channel communication test [CRC error test enabled]

14.10.1 Standard Test Mode

CRC tests are enabled in standard test mode. The CRC value calculated by the RS-CANFD module based on the transmit message or receive message is stored in the register. This CRC value is stored in the CRCREG[14:0] bits in the RSCFDn(CFD)CmERFL register when the message is a classical CAN frame (CRC length = 15 bits) or in the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register when the message is a CAN FD frame (CRC length = 17 or 21 bits). Use the inter-channel communication test function for CRC error tests. For details, see **Section 14.10.6.1, CRC Error Test**.

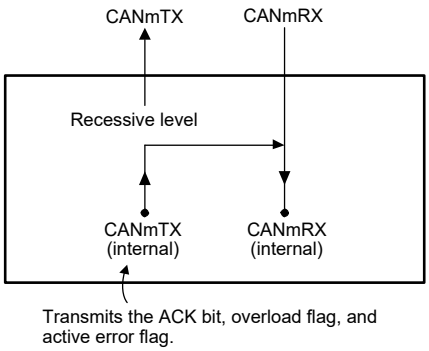
14.10.2 Listen-Only Mode

Listen-only mode allows reception of data frames and remote frames. Only recessive bits are transmitted on the CAN bus, and the ACK bit, overload flag, and active error flag are not transmitted.

Listen-only mode is available for detecting the communication speed.

Do not make a transmit request from any buffer or queue in listen-only mode.

Figure 14.12 shows the connection when listen-only mode is selected.



Note: m = 0 to 3

Figure 14.12 Connection when Listen-Only Mode is Selected

14.10.3 Self-Test Mode (Loopback Mode)

In self-test mode, transmitted messages are compared with the receive rule of the own channel and the messages are stored in a buffer if they have passed through the filter processing. Messages transmitted from other CAN nodes are compared only with the receive rule for which the GAFLLB bit in the RSCFDn(CFD)GAFLIDj register (j = 0 to 15) is set to 0 (when a message transmitted from another CAN node is received).

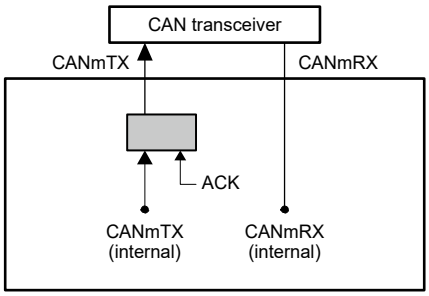
If the mirror function and self-test mode are both enabled, the self-test mode setting takes precedence.

14.10.3.1 Self-Test Mode 0 (External Loopback Mode)

Self-test mode 0 is used to perform a loopback test within a channel including the CAN transceiver.

In self-test mode 0, transmitted messages are handled as messages received through the CAN transceiver and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

Figure 14.13 shows the connection when self-test mode 0 is selected.



Note: m = 0 to 3

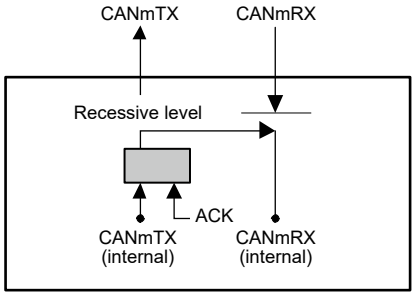
Figure 14.13 Connection when Self-Test Mode 0 is Selected

14.10.3.2 Self-Test Mode 1 (Internal Loopback Mode)

In self-test mode 1, transmitted messages are handled as received messages and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

In self-test mode 1, internal feedback from the internal CANmTX pin (m = 0 to 3) to the internal CANmRX pin is performed. The external CANmRX pin input is isolated. The external CANmTX pin outputs only recessive bits.

Figure 14.14 shows the connection when self-test mode 1 is selected.



Note: m = 0 to 3

Figure 14.14 Connection when Self-Test Mode 1 is Selected

14.10.4 Restricted Operation Mode (Only in CAN FD Mode)

In restricted operation mode, an ACK bit is generated when a valid data frame and a remote frame have been received, but these frames are not transmitted even if an error frame or an overload frame transmit condition is detected. When a condition is detected, operation is suspended until the bus idle state comes for resynchronization with the CAN communication. The receive error counter (REC) and the transmit error counter (TEC) do not change due to an error.

A desired transmission request can be made for transmission without restrictions.

14.10.5 RAM Test

The RAM test function allows accesses to all CAN RAM addresses.

When the RAM test function is used, the RAM is divided into pages of 256 bytes each. RAM test page is set by the RTMPS[6:0] bits in the RSCFDn(CFD)GTSTCFG register. Data in the set page can be read from and written to the RSCFDn(CFD)RPGACCr register (r = 0 to 63). The available total RAM size is 10112 bytes (2780_H) in classical CAN mode or 14208 bytes (3780_H) in CAN FD mode. Do not access more than 128 bytes in the last page (classical CAN mode: RTMPS[6:0] = 27_H, CAN FD mode: RTMPS[6:0] = 37_H) during RAM test.

14.10.6 Inter-Channel Communication Test

The inter-channel communication test function allows communication test by internally connecting CAN channels to each other. During this test, channels are isolated from the external CAN bus.

Before starting data transmission/reception in channel communication mode, make transmission/reception settings for each channel.

Figure 14.15 shows the connection for inter-channel communication test.

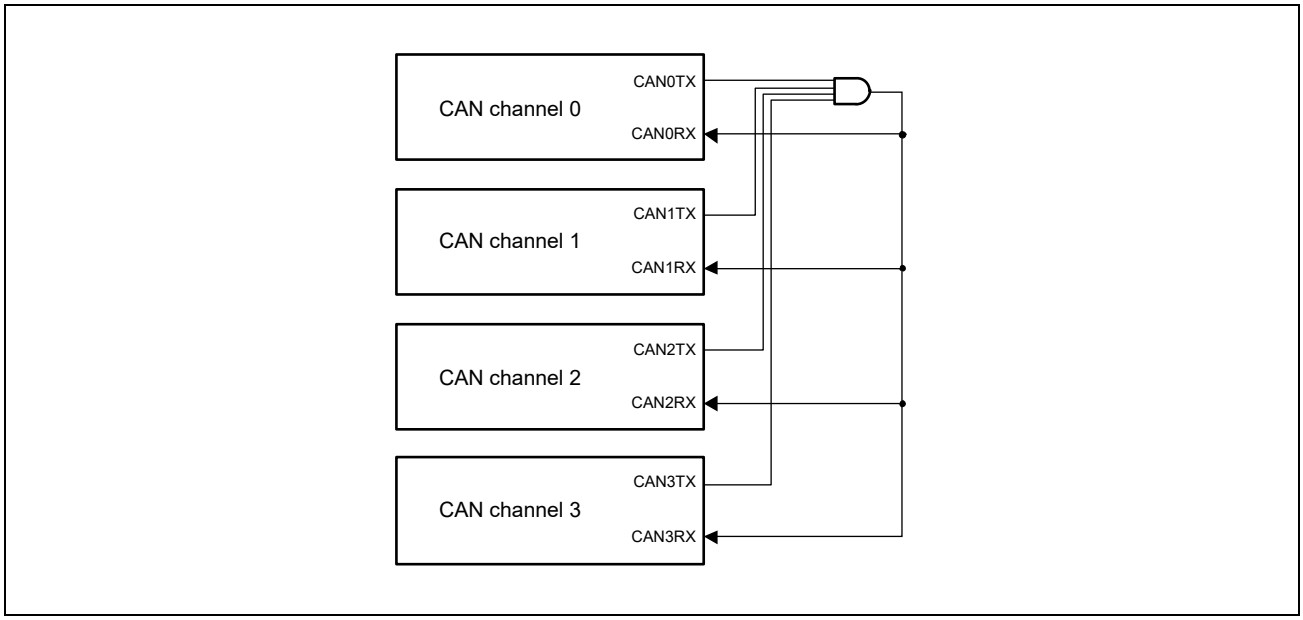


Figure 14.15 Connection for Inter-Channel Communication Test

14.10.6.1 CRC Error Test

A CRC error test is enabled during an inter-channel communication test. The following shows an example of channel 0 CRC error test procedure during a communication test between channel 0 and channel 1.

Preconditions

- Inter-channel communication test is enabled.
- Channel 0 and channel 1 are in standard test mode.

Procedure

1. Make a setting to send a message from the transmit buffer p of channel 1.
2. Set the CRCT bit in the RSCFDn(CFD)C0CTR register to 1 (to enable inversion of the first bit in the received ID field).
3. Set the TMTR bit in the RSCFDn(CFD)TMCp register to 1 (to issue a transmission request to the transmit buffer p of channel 1).
4. Wait for occurrence of a CAN0 error interrupt due to a channel bus error.
5. Read the CRCREG[14:0] bits in the RSCFDn(CFD)CmERFL register or the CRCREG[20:0] bits in the RSCFDnCFDCmFDCRC register of channel 0 and channel 1, and confirm that the CRC values are different on the transmission and the reception side.
6. Confirm that the CERR bit in RSCFDm(CFD)C0ERFL is 1 (CRC error detected).

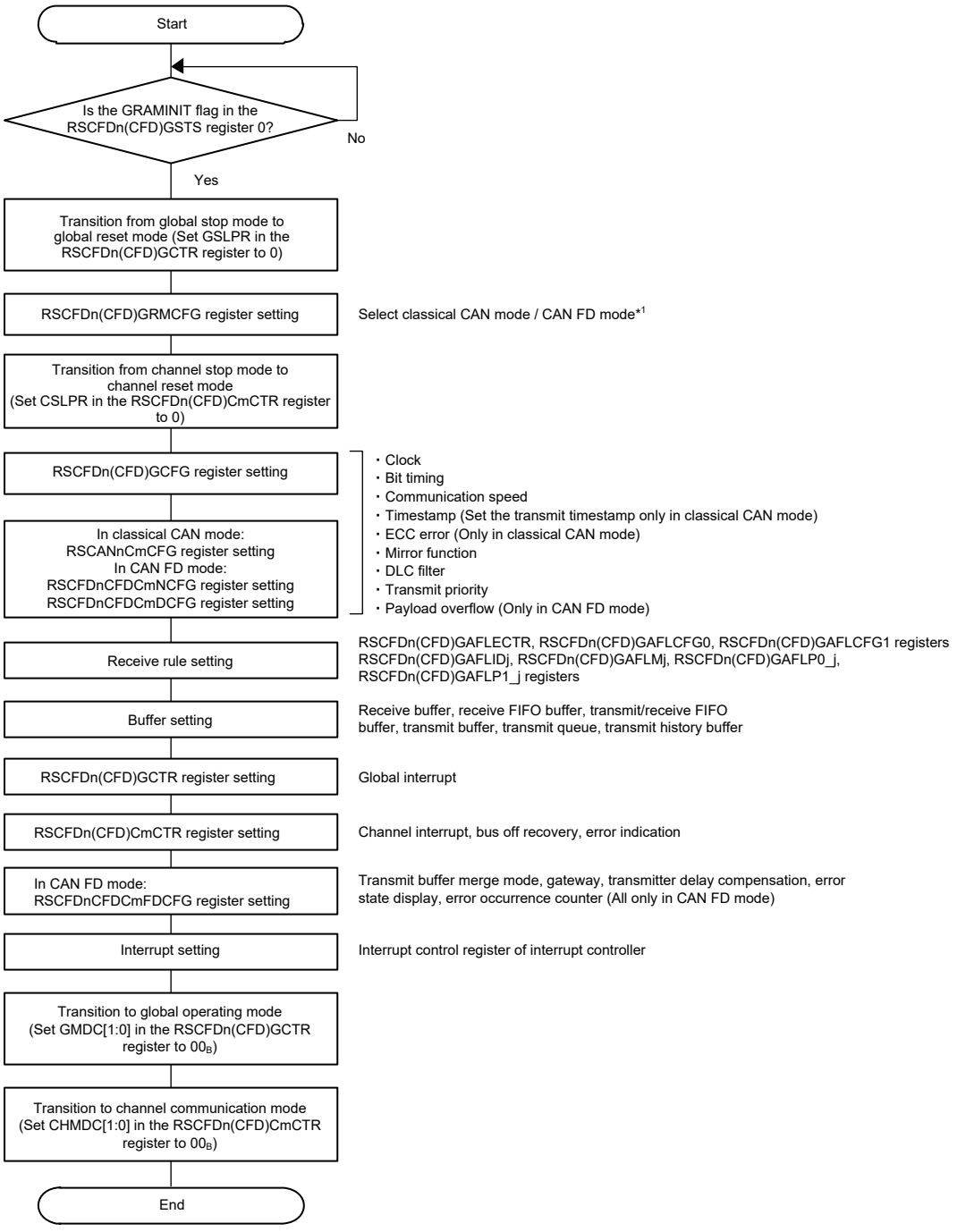
The CRC error test function generates an incorrect CRC value by inverting the first bit in the received ID field. Therefore, note that not a CRC error but a stuff error (continuous 6-bit data of the same level) is detected when a message in which ID's upper 5-bit value is 10000_B or ID's upper 6-bit value is 011111_B is received.

The CRC generation circuit of the RS-CANFD module is contained in the protocol controller of each channel. Another CRC calculation test is not necessary during transmission because the same circuit is used for both transmission and reception.

14.11 RS-CANFD Setting Procedure

14.11.1 Initial Settings

The RS-CANFD module initializes the CAN RAM after the MCU is reset. The RAM initialization time is 7586 cycles of the pclk. The GRAMINIT flag in the RSCFDn(CFD)GSTS register is set to 1 (CAN RAM initialization is ongoing) during the RAM initialization and is cleared to 0 (CAN RAM initialization is finished) when the initialization is completed. Make CAN settings after the GRAMINIT flag is cleared to 0. **Figure 14.16** shows the CAN setting procedure after the MCU is reset.



Note 1. RSCANnGRMCFG and RSCFDnCFDGRMCFG are the same register. Set either of these registers.

Note: m = 0 to 3, j = 0 to 15

Figure 14.16 CAN Setting Procedure after the MCU is Reset

14.11.1.1 Clock Setting

Set the CAN clock (fCAN) as a clock source of the RS-CANFD module. Select the clk_xincan or clkc using the DCS bit in the RSCFDn(CFD)GCFG register.

14.11.1.2 Bit Timing Setting

In the CAN protocol, one bit of a communication frame consists of three segments SS, TSEG1, and TSEG2, of which two segments TSEG1 and TSEG2 can be set by the corresponding registers for each channel. In classical CAN mode, set these two segments in the RSCANnCmCFG register. Two bit rates (nominal bit rate and data bit rate) are provided for CAN FD mode. Set the nominal bit rate in the RSCFDnCFDCmNCFG register and set the data bit rate in the RSCFDnCFDCmDCFG register. Sample point timing can be determined by setting these two segments. This timing can be adjusted in units of 1 Time Quantum (hereafter referred to as Tq). A single Tq is the cycle of clock obtained by dividing the clock selected by the DCS bit in the RSCFDn(CFD)GCFG register. Set a division ratio by the BRP[9:0] bits in the RSCANnCmCFG register in classical CAN mode (CANmTq clock), and by the NBRP[9:0] bits in the RSCFDnCFDCmNCFG register and the DBRP[7:0] bits in the RSCFDnCFDCmDCFG register in CAN FD mode (CANmTq(N) clock and CANmTq(D) clock). ISO11898-1:2015 grants different frequency division ratios for the ordinary bit rate and the data bit rate. If different values are set for these rates, however, it becomes likely that CAN nodes become out-of-synchronization when the active bit rate switches from the ordinary bit rate to the data bit rate. Thus, the NBRP[9:0] value and the DBRP[7:0] value should be equal and the two bit rate values be different according to the respective segment values.

Figure 14.17 shows the bit timing chart. **Table 14.181** shows an example of bit timing setting.

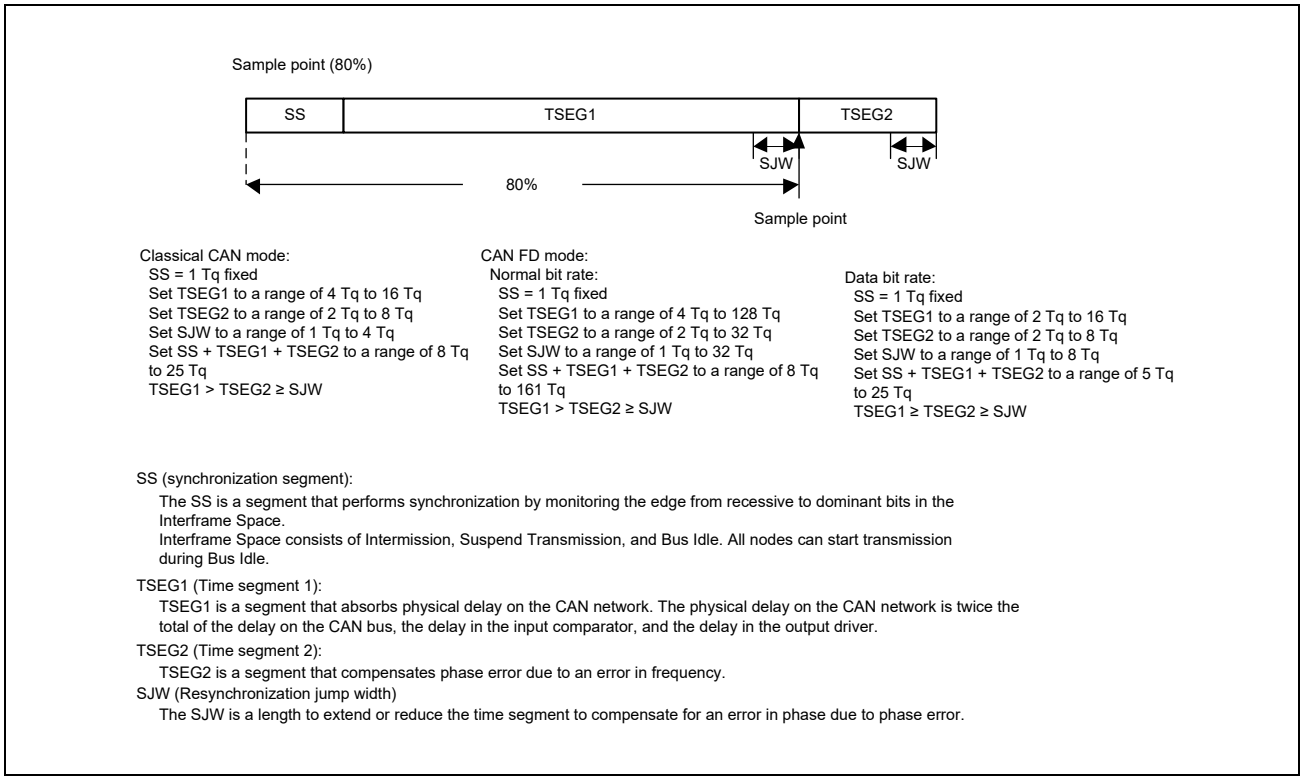


Figure 14.17 Bit Timing Chart

Table 14.181 Example of Bit Timing Setting

	Set Value (Tq)				Sample Point (%)
1 Bit	SS	TSEG1	TSEG2	SJW	**Note:** See **Figure 14.17**.
5Tq*1	1	2	2	1	60.00
8Tq	1	4	3	1	62.50
	1	5	2	1	75.00
10Tq	1	6	3	1	70.00
	1	7	2	1	80.00
16Tq	1	10	5	1	68.75
	1	11	4	1	75.00
20Tq	1	12	7	1	65.00
	1	13	6	1	70.00
50Tq*1	1	39	10	4	80.00

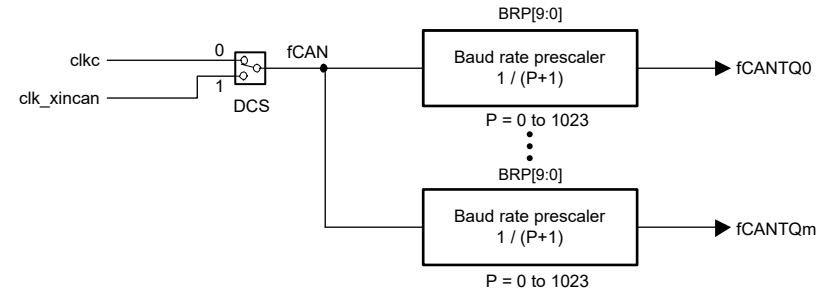
Note 1. Only in CAN FD mode

14.11.1.3 Communication Speed Setting

Set the CAN communication speed for each channel using the fCAN, baud rate prescaler division value, and Tq count per bit time. For CAN FD mode, set two types of transmission rate (arbitration phase and data phase) for each channel.

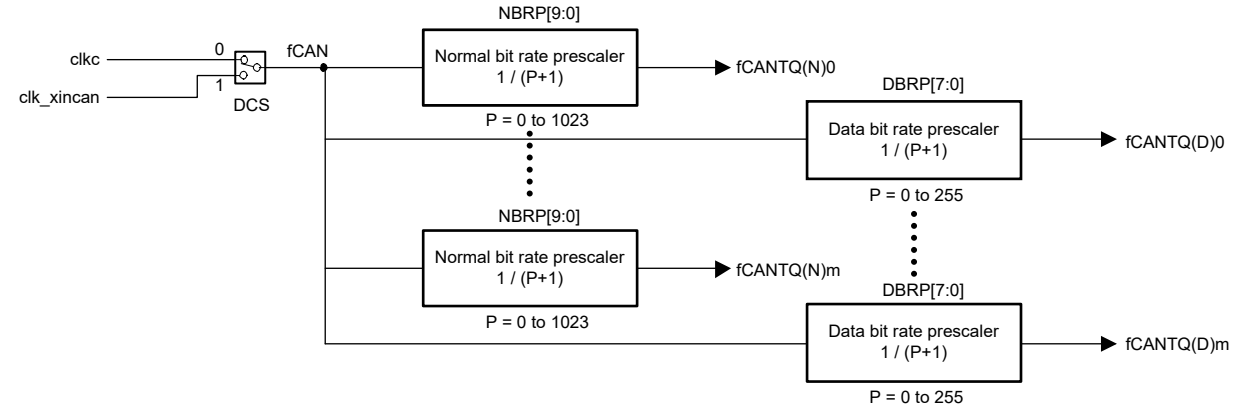
Figure 14.18 shows the CAN clock control block diagram, and **Table 14.182**, **Table 14.183** shows an example of the communication speed setting.

Classical CAN mode:



$$\text{Communication speed} = \frac{\text{fCAN}}{\text{Baud rate prescaler division value} \times (\text{Tq count of 1 bit time})}$$

CAN FD mode:



$$\text{Normal bit rate} = \frac{\text{fCAN}}{\text{Normal bit rate prescaler division value x (Tq count of 1 normal bit time)}}$$

$$\text{Data bit rate} = \frac{\text{fCAN}}{\text{Data bit rate prescaler division value x (Tq count of 1 data bit time)}}$$

Note: m = 0 to 3
DCS: Bit in the RSCFDn(CFD)GCFG register
BRP[9:0]: Bits in the RSCANnCmCFG register
NBRP[9:0]: Bits in the RSCFDnCFDCmNCFG register
DBRP[7:0]: Bits in the RSCFDnCFDCmDCFG register
fCAN: CAN clock
fCANTQm: CANmTq clock
fCANTQ(N)m: CANmTq(N) clock
fCANTQ(D)m: CANmTq(D) clock

Figure 14.18 CAN Clock Control Block Diagram

Table 14.182 Example of Communication Speed Setting (Classical CAN mode)

fCAN / Communication Speed	40 MHz	20 MHz
1 Mbps	8 Tq (5) 20 Tq (2)	10 Tq (2) 20 Tq (1)
500 Kbps	8 Tq (10) 20 Tq (4)	10 Tq (4) 20T Tq (2)
250 Kbps	8 Tq (20) 20 Tq (8)	10 Tq (8) 20 Tq (4)
125 Kbps	8 Tq (40) 20 Tq (16)	10 Tq (16) 20 Tq (8)

Table 14.183 Example of Transmission Rate Setting (Nominal Bit Rate and Data Bit Rate in CAN FD Mode)

fCAN / Communication Rate	40 MHz	20 MHz
Nominal bit rate 1 Mbps Data bit rate 5 Mbps	Nominal bit rate 40 Tq (1) Data bit rate 8 Tq (1)	-
Nominal bit rate 500 Kbps Data bit rate 2 Mbps	Nominal bit rate 80 Tq (1) Data bit rate 20 Tq (1)	Nominal bit rate 40 Tq (1) Data bit rate 10 Tq (1)

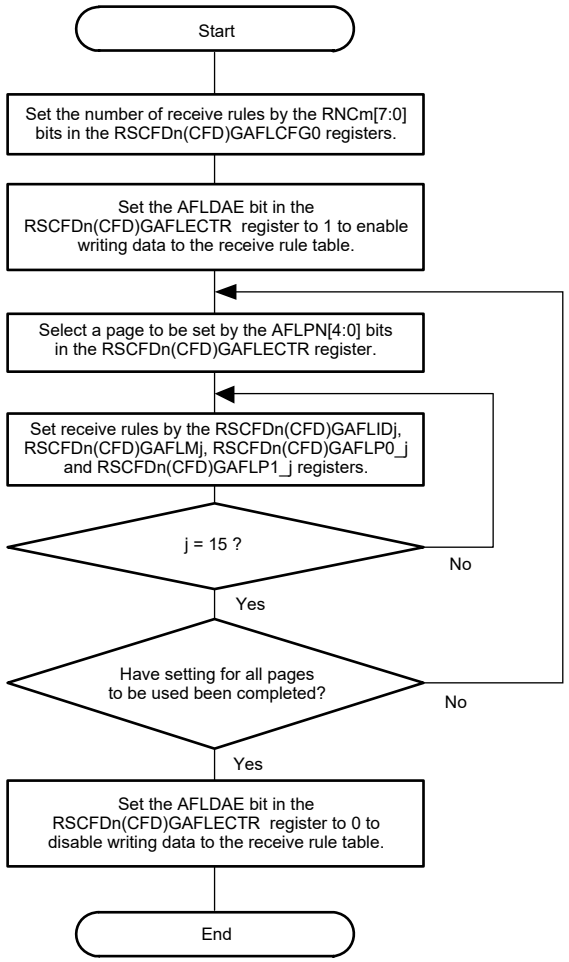
Note: Values in ( ) are baud rate prescaler division values.

14.11.1.4 Receive Rule Setting

Receive rules can be set using receive rule-related registers.

Up to 16 receive rules can be registered per page. Specify pages 0 to 23 (for 6-channel unit) by the AFLPN[4:0] bits in the RSCFDn(CFD)GAFLECTR register. Set receive rule table write enable/disable using the AFLDAE bit.

Figure 14.19 shows the receive rule setting procedure.



Note: m = 0 to 3, j = 0 to 15

Figure 14.19 Receive Rule Setting Procedure

14.11.1.5 Buffer Setting

Set the number of buffers to be used (number of messages to be stored) and interrupt sources. Also set the payload storage size for CAN FD mode. For transmit/receive FIFO buffers that are set to transmit mode, set transmit buffers to be linked.

In classical CAN mode, up to 4096 bytes of the RAM can be used in receive buffers and FIFO buffers. Up to 256 buffers are available, and 16 bytes are used per buffer. Configure the buffers so that the following conditions are met.

Number of receive buffers
+ total number of depth of receive FIFO buffers x
+ total number of depth of transmit/receive FIFO buffers k ≤ 256 buffers

In CAN FD mode, up to 7168 bytes of the RAM can be used in receive buffers and FIFO buffers. Configure the buffers so that the following conditions are met.

Number of receive buffers × (12 + payload storage size)
+ total of (number of depth × (12 + payload storage size)) of receive FIFO buffers x
+ total of (number of depth × (12 + payload storage size)) of transmit/receive FIFO buffers k ≤ 7168 bytes

Figure 14.20 shows the buffer configuration. **Figure 14.21** shows the buffer setting procedure.

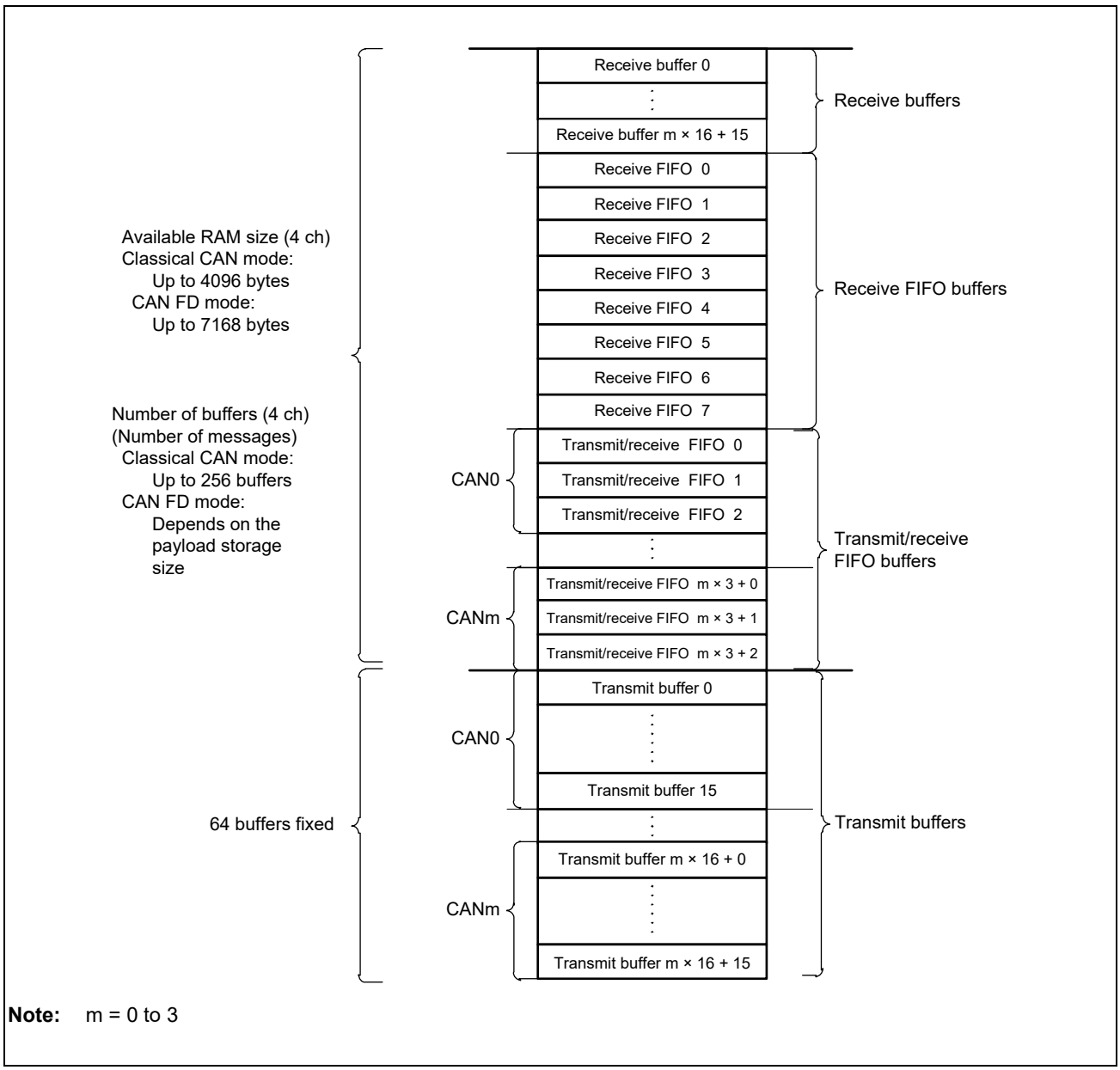
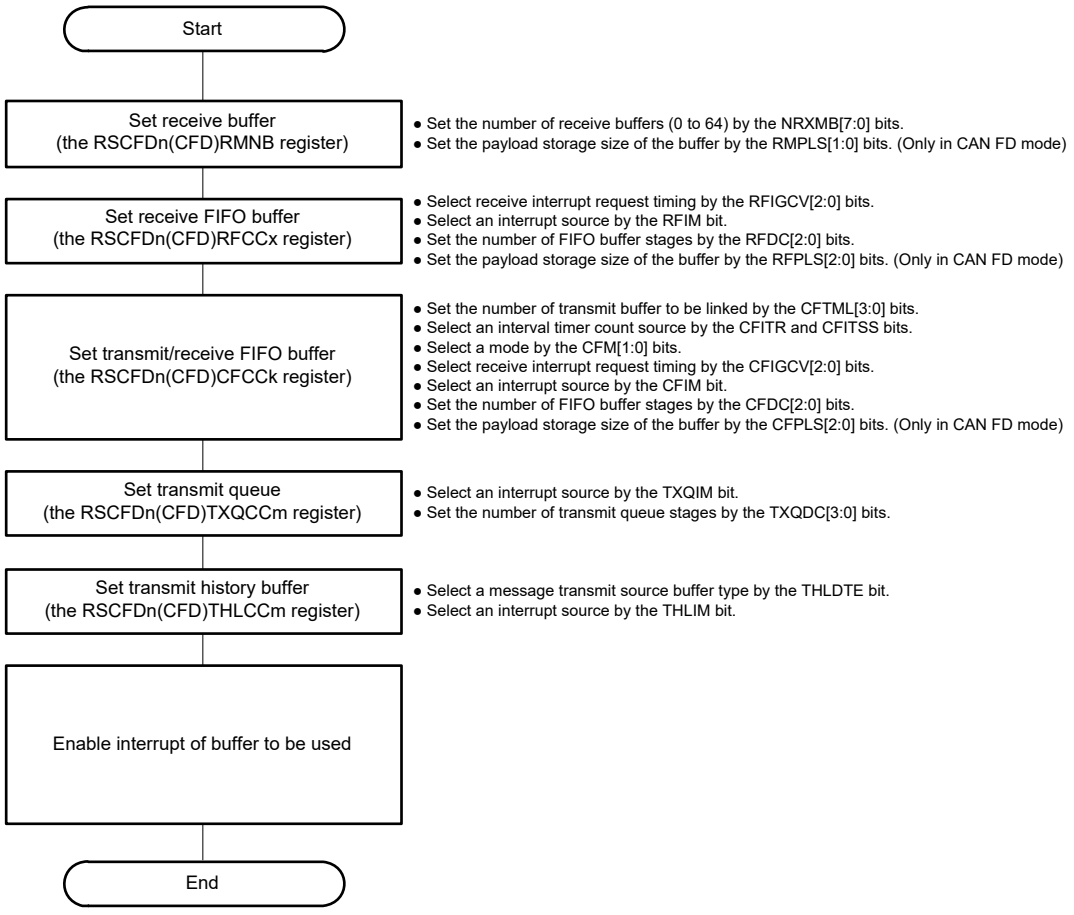


Figure 14.20 Buffer Configuration

CAUTION

Receive buffers, receive FIFO buffers, transmit/receive FIFO buffers, and transmit buffers are located in succession.



Note: k = 0 to 11, x = 0 to 7

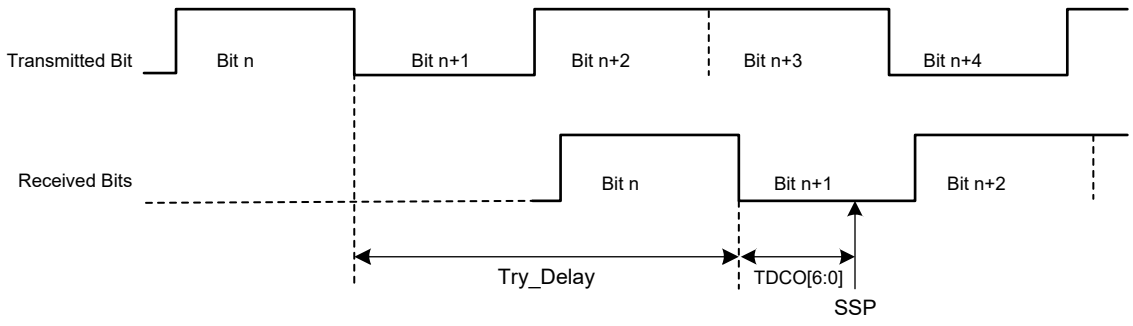
Figure 14.21 Buffer Setting Procedure

14.11.1.6 Transmitter Delay Compensation (Only in CAN FD Mode)

A high baud rate is used in CAN FD mode. Transmitter delay compensation is provided as a function to accept propagation delay in this case.

To use this function, set the TDCE bit in the RSCFDnCFDCmFDCFG register to 1. Also set the secondary sample point (SSP) timing used in the data phase by the TDCOC bit and TDCO[6:0] bits in the RSCFDnCFDCmFDCFG register.

When the TDCOC bit is 0, the SSP timing equals the total value of the delay measured by the RS-CANFD module and the TDCO[6:0] value. (This value is rounded off to the nearest integer of Tq.) Usually, the TDCO[6:0] value must be equal to SS + TSEG1, the sample point timing



Note: Bit n: Desired bit of the data phase
TDCO[6:0]: Bits of the RSCFDnCFDCmFDCFG register

Figure 14.22 SSP Timing

When the TDCOC bit is 1, the SSP timing is determined only by the TDCO[6:0] value. (When the DBRP[7:0] value in the RSCFDnCFDCmDCFG register is larger than 0, the TDCO[6:0] value is also rounded off to the nearest integer of Tq.)

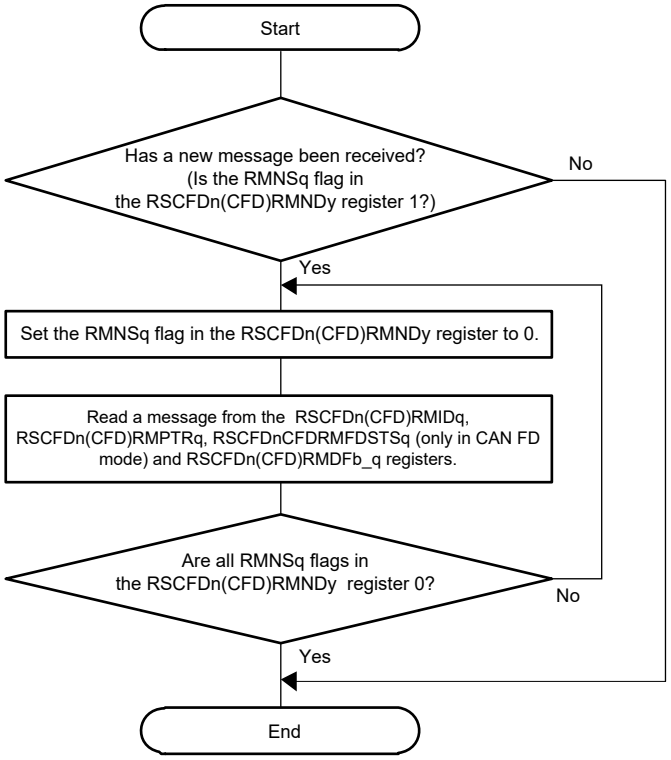
The RS-CANFD module compensates a delay up to (3 CANm bit time - 2 fCAN). (CANm bit time is data bit rate value.)

If the TDCE bit of the RSCFDnCFDCmFDCFG register is 1 (enabling transmitter delay compensation), set the NBRP[9:0] and DBRP[7:0] bits to the same value equal to or less than 1.

14.11.2 Reception Procedure

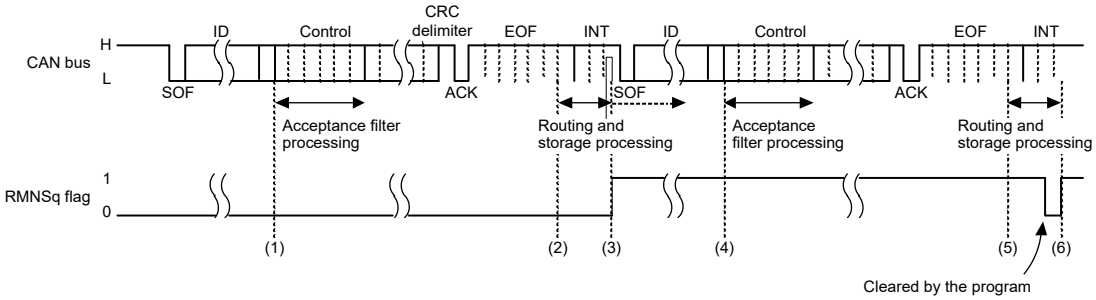
14.11.2.1 Receive Buffer Reading Procedure

When the processing to store received messages in a receive buffer starts, the RMNSq flag in the RSCFDn(CFD)RMNDy register (y = 0, 1, q = 0 to 63) is set to 1 (receive buffer q contains a new message). Messages can be read from registers RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq (only in CAN FD mode), and RSCFDn(CFD)RMDFb_q (b = 0 or 1 in classical CAN mode, b = 0 to 4 in CAN FD mode). **Figure 14.23** shows the receive buffer reading procedure. This procedure ensures the consistency of messages read from registers RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q.



Note: y = 0, 1, q = 0 to 63, b = 0 to 1 (classical CAN mode), 0 to 4 (CAN FD mode)

Figure 14.23 Receive Buffer Reading Procedure



Note: y = 0, 1, q = 0 to 63
RMNSq: Flag in the RSCFDn(CFD)RMNDy register

Figure 14.24 Receive Buffer Reception Timing Chart

(1) When the ID field in a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing, the processing to store the message in the specified receive buffer starts.
When the message storage processing starts, the RMNSq flag in the corresponding RSCFDn(CFD)RMNDy register is set to 1 (the receive buffer contains a new message). If other channels are performing filter processing or transmit priority determination processing, the routing processing and the storage processing may be delayed.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(6) When the corresponding RMNSq flag is cleared to 0 (the receive buffer contains no new message), this flag is set to 1 again when the message storage processing starts. Even if the RMNSq flag remains 1, a new message is overwritten to the receive buffer. The RMNSq flag should not be cleared to 0 during storage of messages.

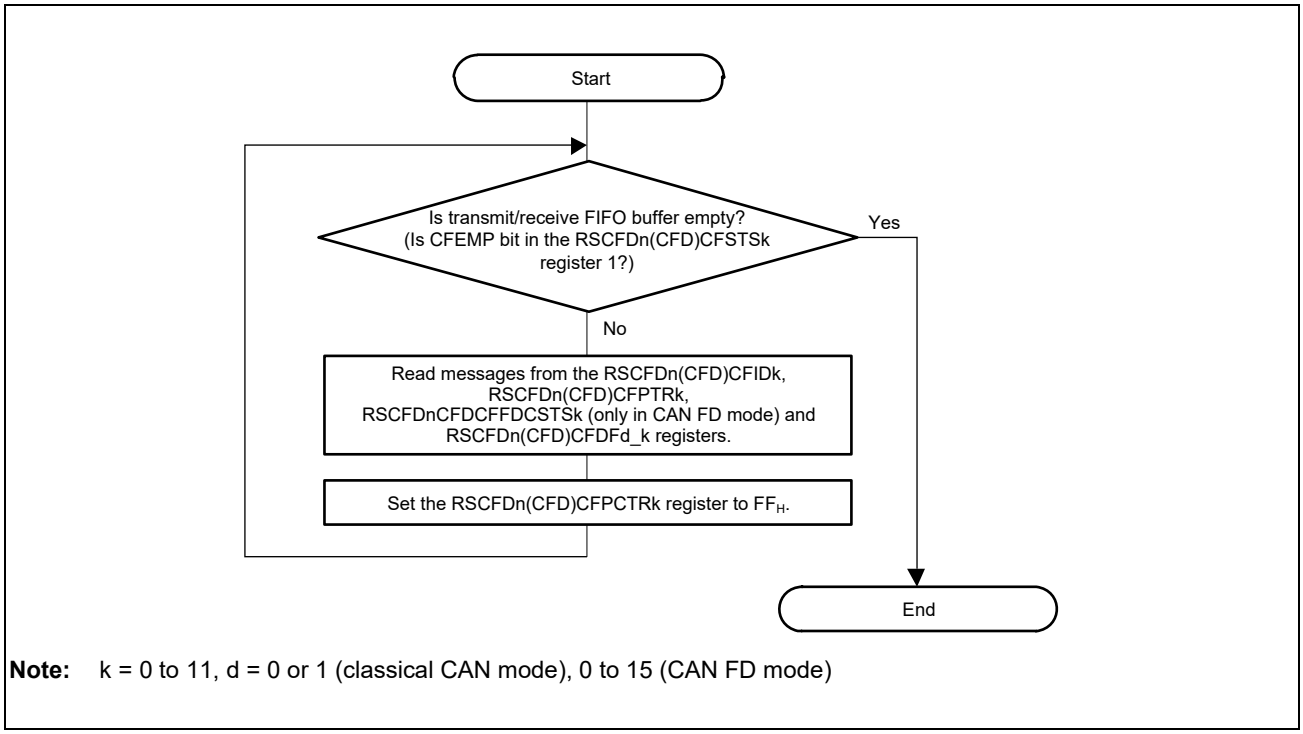
14.11.2.2 FIFO Buffer Reading Procedure

When received messages have been stored in one or more receive FIFO buffers or a transmit/receive FIFO buffer that is set to receive mode or gateway mode, the corresponding message count display counter (RFMC[7:0] bits in the RSCFDn(CFD)RFSTSx register (x = 0 to 7) or CFMC[7:0] bits in the RSCFDn(CFD)CFSTSk register (k = 0 to 11)) is incremented. At this time, when the RFIE bit (receive FIFO interrupt is enabled) in the RSCFDn(CFD)RFCCx register or the CFRXIE bit (transmit/receive FIFO receive interrupt is enabled) in the RSCFDn(CFD)CFCCk register is set to 1, an interrupt request is generated. Received messages can be read from the RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx (only in CAN FD mode), and RSCFDn(CFD)RFDFd_x (d = 0 or 1 in classical CAN mode, d = 0 to 15 in CAN FD mode) registers for receive FIFO buffers, or from the RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk (only in CAN FD mode), and RSCFDn(CFD)CFDFd_k registers for transmit/receive FIFO buffers. Messages in FIFO buffers can be read sequentially on a first-in, first-out basis.

When the message count display counter value matches the FIFO buffer depth (a value set by the RFDC[2:0] bits in the RSCFDn(CFD)RFCCx register or the CFDC[2:0] bits in the RSCFDn(CFD)CFCCk register), the RFFLL or CFFLL flag is set to 1 (the receive FIFO buffer is full).

When all messages have been read out of the FIFO buffer, the RFEMP flag in the RSCFDn(CFD)RFSTSx register or the CFEMP flag in the RSCFDn(CFD)CFSTSk register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)).

If the RFE bit or the CFE bit is cleared to 0 (no receive FIFO buffer is used) with the interrupt request flag (RFIF flag in the RSCFDn(CFD)RFSTSx register or CFRXIF flag in the RSCFDn(CFD)CFSTSk register) set to 1 (a receive FIFO interrupt request is present), the interrupt request flag is not automatically cleared to 0. The program must clear the interrupt request flag to 0.



Note: k = 0 to 11, d = 0 or 1 (classical CAN mode), 0 to 15 (CAN FD mode)

Figure 14.25 Transmit/Receive FIFO Buffer Reading Procedure

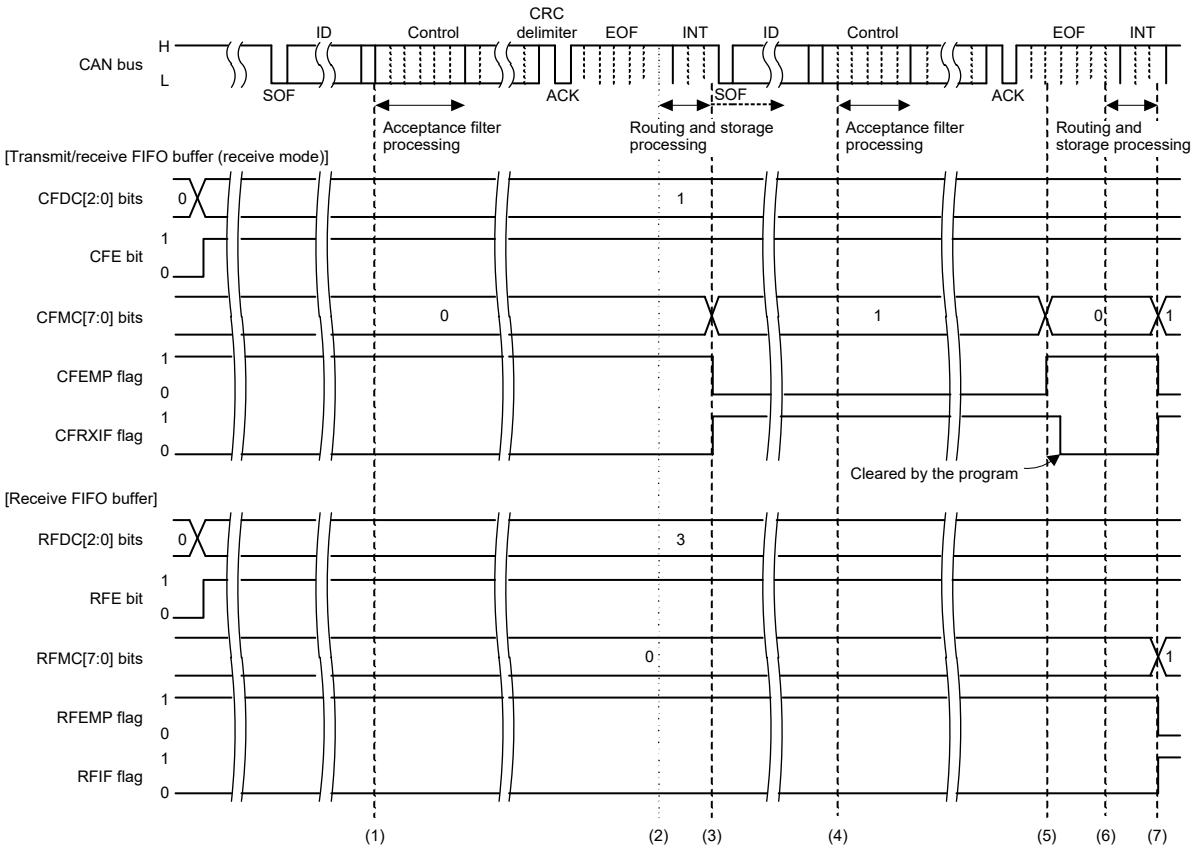
When reading a message in CAN FD mode, do not read the RSCFDnCFDRFDFd_x or RSCFDnCFDCFDFd_k register corresponding to the area exceeding the payload storage size specified by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register or the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

Table 14.184 Payload Storage Area of Receive FIFO Buffer

Set RFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000_B	8 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF1_x
001_B	12 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF2_x
010_B	16 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF3_x
011_B	20 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF4_x
100_B	24 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF5_x
101_B	32 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF7_x
110_B	48 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF11_x
111_B	64 bytes	RSCFDnCFDRFDF0_x to RSCFDnCFDRFDF15_x

Table 14.185 Payload Storage Area of Transmit/Receive FIFO Buffer

Set CFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000_B	8 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF1_k
001_B	12 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF2_k
010_B	16 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF3_k
011_B	20 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF4_k
100_B	24 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF5_k
101_B	32 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF7_k
110_B	48 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF11_k
111_B	64 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF15_k



Note: k = 0 to 11, x = 0 to 7
CFDC[2:0], CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0], CFEMP, CFRXIF: Flags in the RSCFDn(CFD)CFSTSk register
RFDC[2:0], RFE: Bits in the RSCFDn(CFD)RFCCx register
RFMC[7:0], RFEMP, RFIF: Flags in the RSCFDn(CFD)RFSTSx register

Figure 14.26 FIFO Buffer Reception Timing Chart

(1) When the ID field in a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing and the CFE bit in the RSCFDn(CFD)CFCCk register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCk register is 001_B or more, the message is stored in the transmit/receive FIFO buffer that is set to receive mode. The CFMC[7:0] value in the RSCFDn(CFD)CFSTSk register is incremented and becomes 01_H. When the CFIM bit in the RSCFDn(CFD)CFCCk register is set to 1 (a FIFO receive interrupt request is generated each time a message has been received), the CFRXIF flag in the RSCFDn(CFD)CFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). The CFRXIF flag can be reset to 0 by the program.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) Read received messages from the RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, and RSCFDn(CFD)CFDFd_k registers and write FF_H to the RSCFDn(CFD)CFPCTRk register. This causes the CFMC[7:0] bits in the RSCFDn(CFD)CFSTSk register to be decremented. When CFMC[7:0] becomes 00_H, the CFEMP flag in the RSCFDn(CFD)CFSTSk register becomes 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

(6) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCFDn(CFD)GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(7) The message is stored in the transmit/receive FIFO buffer set in receive mode when the message has passed through the DLC filter process if the CFE bit is set to 1 (transmit/receive FIFO buffers are used), and the CFDC[2:0] bits are set to 001_B or more. The CFMC[7:0] bit value is incremented by 1 to be 01_H. When the CFIM bit is set to 1 (an interrupt occurs each time a message has been received), the CFRXIF flag is set to 1 (a transmit/receive FIFO receive interrupt request is present).
The message is stored in the receive FIFO buffer if the RFE bit in the RSCFDn(CFD)RFCCx register is set to 1 (receive FIFO buffers are used), and the RFDC[2:0] bits in the RSCFDn(CFD)RFCCx register are set to 001_B or more. The RFMC[7:0] bits in the RSCFDn(CFD)RFSTSx register are set to 01_H by being incremented by 1. When the RFIM bit in the RSCFDn(CFD)RFCCx register is set to 1 (an interrupt occurs each time a message has been received), the RFIF flag in the RSCFDn(CFD)RFSTSx register is set to 1 (a receive FIFO interrupt request is present).

14.11.2.3 FIFO Buffer Reading Procedure by DMA Transfer

In CAN FD mode, the following FIFO buffers can be read by DMA transfer.

- All receive FIFO buffers x (x = 0 to 7)
- The first transmit/receive FIFO buffer k allocated to channel m (k = 3 × m, m = 0 to 3)

The DMA enable bit (RFDMAEx or CFDMAEm bit in the RSCFDnCFDCDTCT register) can be set at any time. However, before setting the DMA enable bit to 1 (to enable DMA transfer requests), set the receive interrupt enable bit (RFIE bit in the RSCFDnCFDRFCCx register or CFRXIE bit in the RSCFDnCFDCFCCk register) of related FIFO buffers to 0 (to disable interrupts). When DMA transfer requests are enabled, do not write a value to the FIFO control register (RSCFDnCFDRFCCx register or RSCFDnCFDCFCCk register).

When an unread message is remaining in a DMA transfer-enabled FIFO buffer, a DMA transfer request trigger is generated. Specify the FIFO access register*[1] address for the transfer source address, and adjust the transfer size so that data can be read to the end of the payload storage area with a single trigger. The end of the payload storage area depends on the payload storage size specified by the RFPLS[2:0] bits in the RSCFDnCFDRFCCx register or the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

After the end of the payload stored in the FIFO buffer has been read, the RFMC[7:0] value in the RSCFDnCFDRFSTSx register or the CFMC[7:0] value in the RSCFDnCFDCFSTSk register is automatically decremented. After that, if an unread message is remaining in the FIFO buffer, a trigger is generated again.

When the RFDMAEx or CFDMAEm bit is set to 0 (to disable DMA transfer requests) during DMA transfer, wait until the DMA transfer status (RFDMASTSx or CFDMASTSm bit in the RSCFDnCFDCDTSTS register) is cleared to 0 (DMA transfer disabled), and then start the next processing (enabling DMA transfer again etc.). When disabling DMA transfer, examine how to process a message remaining in the FIFO buffer and a newly arriving message. When the FIFO buffer is enabled, it continues to receive messages.

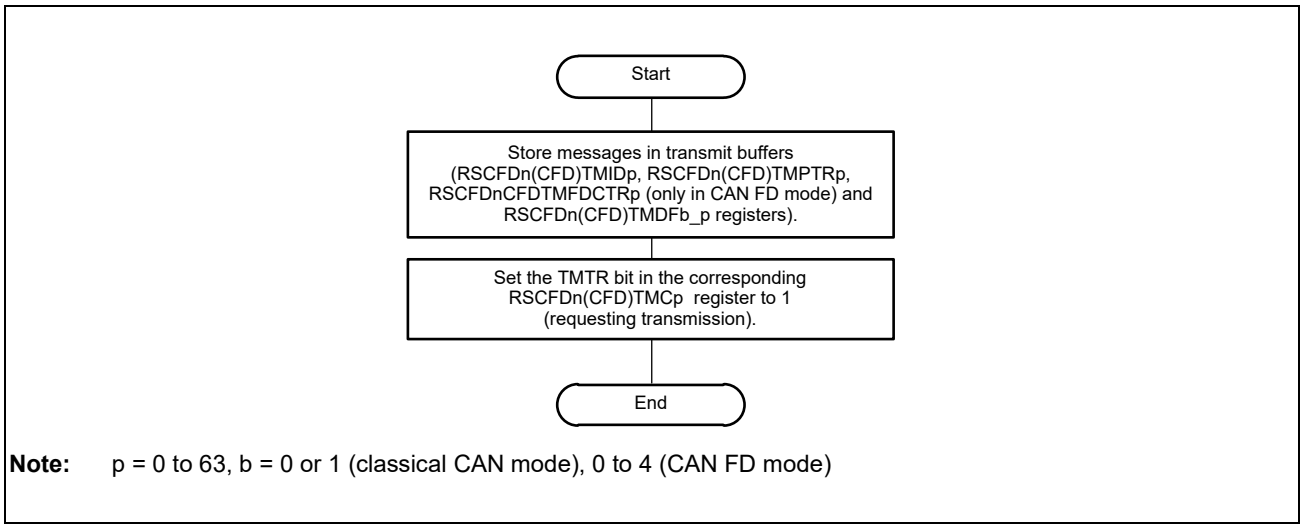
Note 1. • For receive FIFO buffers
RCFDCnCFDRFIDx, RCFDCnCFDRFPTRx, RCFDCnCFDRFFDSTSx, RCFDCnCFDRFDFd_x
• For transmit/receive FIFO buffers
RCFDCnCFDCFIDk, RCFDCnCFDCFPTRk, RCFDCnCFDCFFDCSTSk, RCFDCnCFDCFDFd_k

14.11.3 Transmission Procedure

14.11.3.1 Procedure for Transmission from Transmit Buffers

Figure 14.27 shows the procedure for transmission from transmit buffers.

Figure 14.28 shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmission has been successfully completed. **Figure 14.29** shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmit abort has been completed.



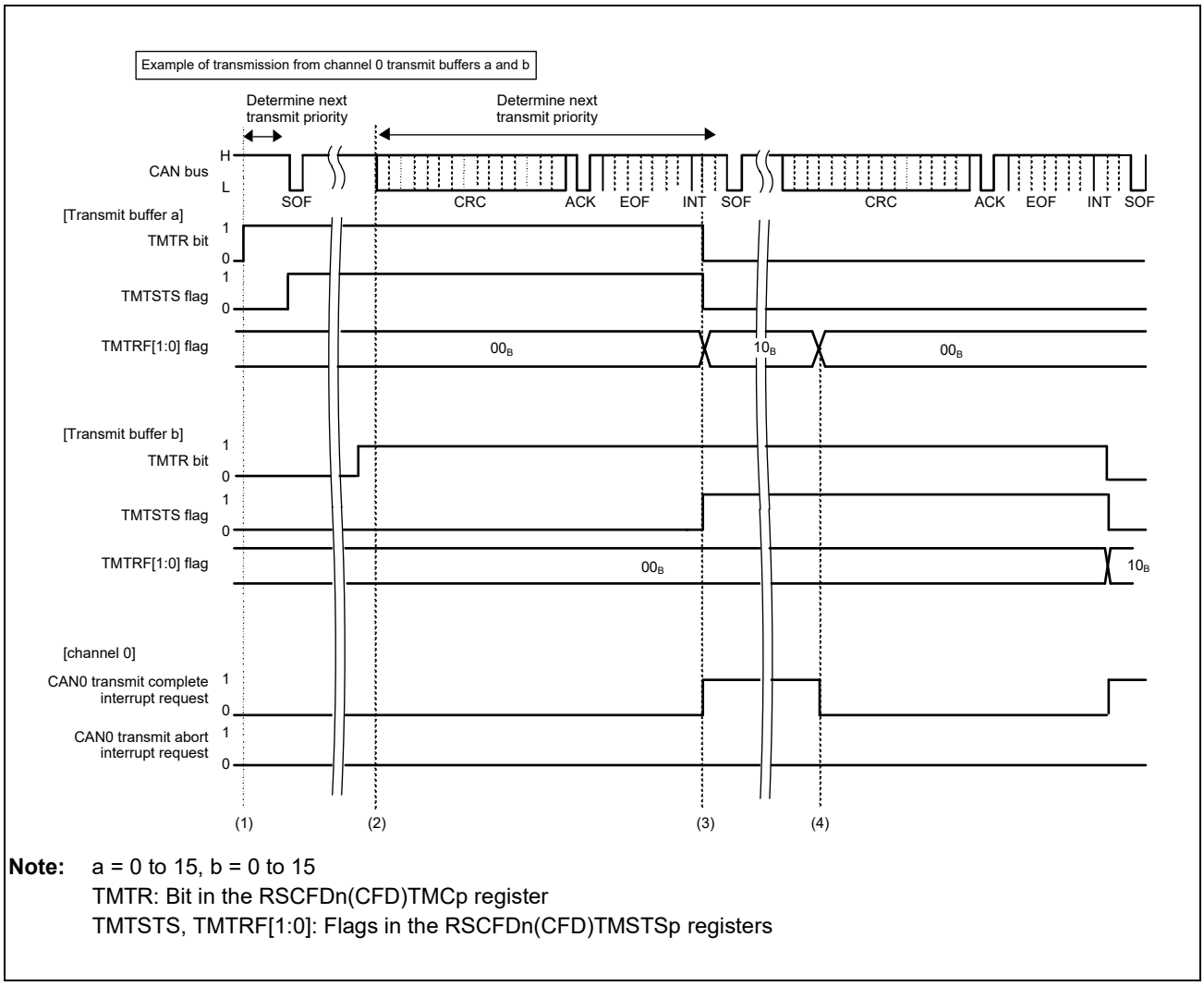
Note: p = 0 to 63, b = 0 or 1 (classical CAN mode), 0 to 4 (CAN FD mode)

Figure 14.27 Procedure for Transmission from Transmit Buffers

In CAN FD mode and transmit buffer merge mode, messages with a payload size of more than 20 bytes can be transmitted from transmit buffers (16 × m) + 0 and transmit buffers (16 × m) + 3. At this time, transmit buffers (16 × m) + 1 to (16 × m) + 2 and transmit buffers (16 × m) + 4 to (16 × m) + 5 are allocated as a payload storage area. Registers RSCFDnCFDTMIDp, RSCFDnCFDTMPTRp, and RSCFDnCFDTMFDCTRp corresponding to these buffers can be used as data field registers that can store 4-byte data bytes (payload) like the RSCFDnCFDTMDFb_p register. **Table 14.186** shows message storage registers when transmitting a message with a payload size of more than 20 bytes from transmit buffer 0.

Table 14.186 Message Storage Registers in Transmit Buffer Merge Mode (Example of Transmit Buffer 0)

Transmit Buffer	Offset from Base Address	Symbol	Register Function in Transmit Buffer Merge Mode
Transmit buffer 0	4000_H	RSCFDnCFDTMID0	Transmit buffer 0 ID data, transmit history data store enable bit, RTR bit,and IDE bit
	4004_H	RSCFDnCFDTMPTR0	Transmit buffer 0 label data and DLC data
	4008_H	RSCFDnCFDTMFDCTR0	Transmit buffer 0 ESI bit, BRS bit, and FDF bit
	$400C_H$ to $401C_H$	RSCFDnCFDTMDF0_0 to RSCFDnCFDTMDF4_0	Transmit buffer 0 data bytes 0, 1, 2, and 3 to transmit buffer 0 data bytes 16, 17, 18, and 19
Transmit buffer 1	4020_H	RSCFDnCFDTMID1	Transmit buffer 0 data bytes 20, 21, 22, and 23
	4024_H	RSCFDnCFDTMPTR1	Transmit buffer 0 data bytes 24, 25, 26, and 27
	4028_H	RSCFDnCFDTMFDCTR1	Transmit buffer 0 data bytes 28, 29, 30, and 31
	$402C_H$ to $403C_H$	RSCFDnCFDTMDF0_1 to RSCFDnCFDTMDF4_1	Transmit buffer 0 data bytes 32, 33, 34, and 35 to transmit buffer 0 data bytes 48, 49, 50, and 51
Transmit buffer 2	4040_H	RSCFDnCFDTMID2	Transmit buffer 0 data bytes 52, 53, 54, and 55
	4044_H	RSCFDnCFDTMPTR2	Transmit buffer 0 data bytes 56, 57, 58, and 59
	4048_H	RSCFDnCFDTMFDCTR2	Transmit buffer 0 data bytes 60, 61, 62, and 63
	$404C_H$ to $405C_H$	RSCFDnCFDTMDF0_2 to RSCFDnCFDTMDF4_2	Not used



Note: a = 0 to 15, b = 0 to 15
TMTR: Bit in the RSCFDn(CFD)TMCp register
TMTSTS, TMTRF[1:0]: Flags in the RSCFDn(CFD)TMSTSp registers

Figure 14.28 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) When the TMTR bit in the RSCFDn(CFD)TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCFDn(CFD)TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

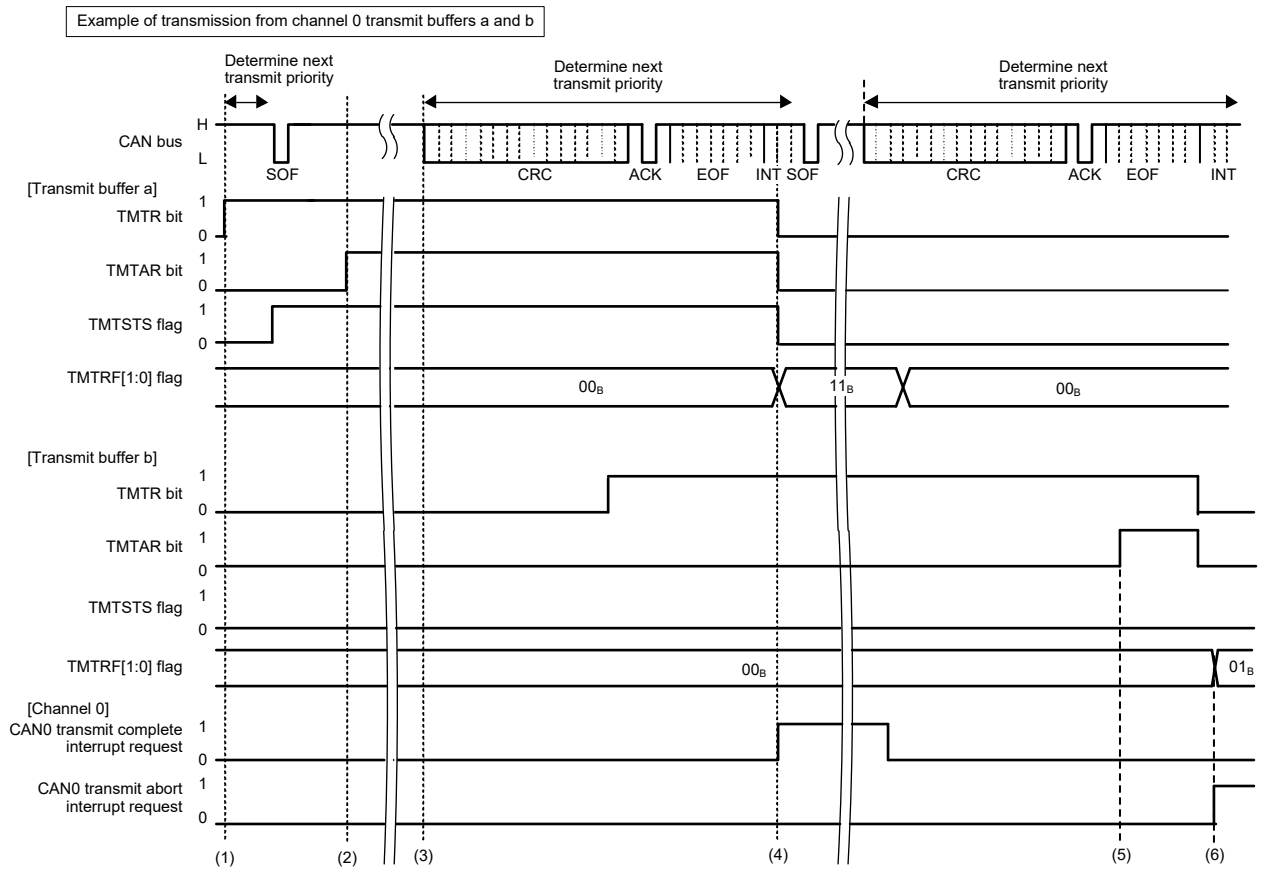
(2) When a transmit request from a buffer is present, the priority determination starts at the first bit of CRC field for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(3) When transmission completes successfully, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSa register is set to 10_B (transmission has been completed (without transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCFDn(CFD)TMCa register are cleared to 0. When the TMIEa bit in the RSCFDn(CFD)TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B (transmission is in progress or no transmit request is present).

(4) Before starting the next transmission, set the TMTRF[1:0] flag to 00_B. Write the next message to the transmit buffer, and then set the TMTR bit to 1 (transmission is requested). The TMTR bit can be set to 1 only when the TMTRF[1:0] flag value is 00_B.

If an arbitration-lost has occurred after transmission is started, the TMTSTS flag is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC field to search the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

When a 2-bit ECC error is detected during the priority determination processing, no data is transmitted (in classical CAN mode, only when the EEFE bit in the RSCANnGCFG register is 1).



Note: a = 0 to 15, b = 0 to 15
TMTR, TMTAR: Bits in the RSCFDn(CFD)TMCp register
TMTSTS, TMTRF[1:0]: Flags in the RSCFDn(CFD)TMSTSp register

Figure 14.29 Transmit Buffer Transmission Timing Chart (Transmit Abort Completed)

(1) When the TMTR bit in the RSCFDn(CFD)TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCFDn(CFD)TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

(2) When it is determined that the transmit buffer is used for the next transmission or transmission is in progress, message transmission is not aborted unless an error or arbitration loss occurs even if the TMTAR bit is set to 1 (transmit abort is requested).

(3) The priority determination starts at the first bit of the CRC field for the next transmission. In this timing chart, buffer b is not selected as the next transmit buffer. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(4) When transmission completes successfully, the TMTRF[1:0] flag in the RSCFDn(CFD)TMSTSa register is set to 11_B (transmission has been completed (with transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCFDn(CFD)TMCa register are cleared to 0. When the TMIEa value in the RSCFDn(CFD)TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B (transmission is in progress or no transmit request is present).

(5) While another CAN node is transmitting data on the CAN bus (TMTSTS flag = 0), if the TMTAR bit is set to 1 while the corresponding channel is determining transmit priority, the TMTR bit cannot be cleared to 0.

(6) After the internal processing time has passed, the transmission is terminated and the TMTRF[1:0] flag is set to 01_B. When the transmit buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not being made, an abort request is immediately accepted and the TMTRF[1:0] flag is set to 01_B. At this time, the TMTR and TMTAR bits are cleared to 0. When transmit abort is completed with the TAIE bit in the RSCFDn(CFD)CmCTR register set to 1 (transmit abort interrupt is enabled), an interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B.

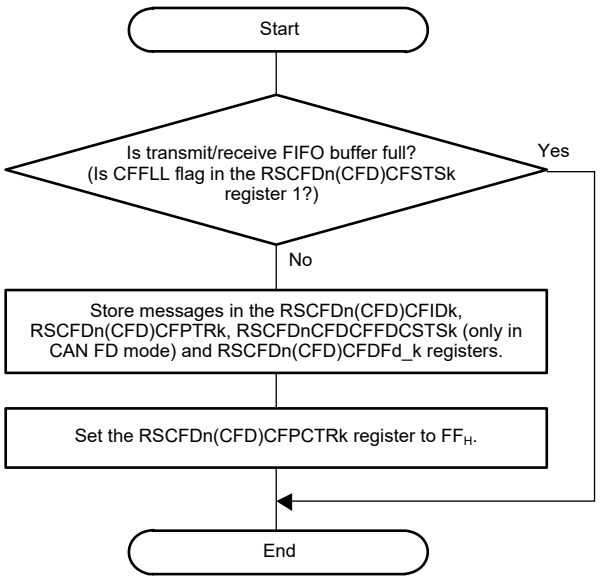
If an arbitration loss has occurred after the CAN channel started transmission, the TMTSTS bit is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC field to find the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

When a 2-bit ECC error is detected during the priority determination processing, no data is transmitted (in classical CAN mode, only when the EEFE bit in the RSCANnGCFG register is 1).

14.11.3.2 Procedure for Transmission from Transmit/Receive FIFO Buffers

Figure 14.30 shows the procedure for transmission from transmit/receive FIFO buffers.

Figure 14.31 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmission has been successfully completed. **Figure 14.32** shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmit abort has been completed.



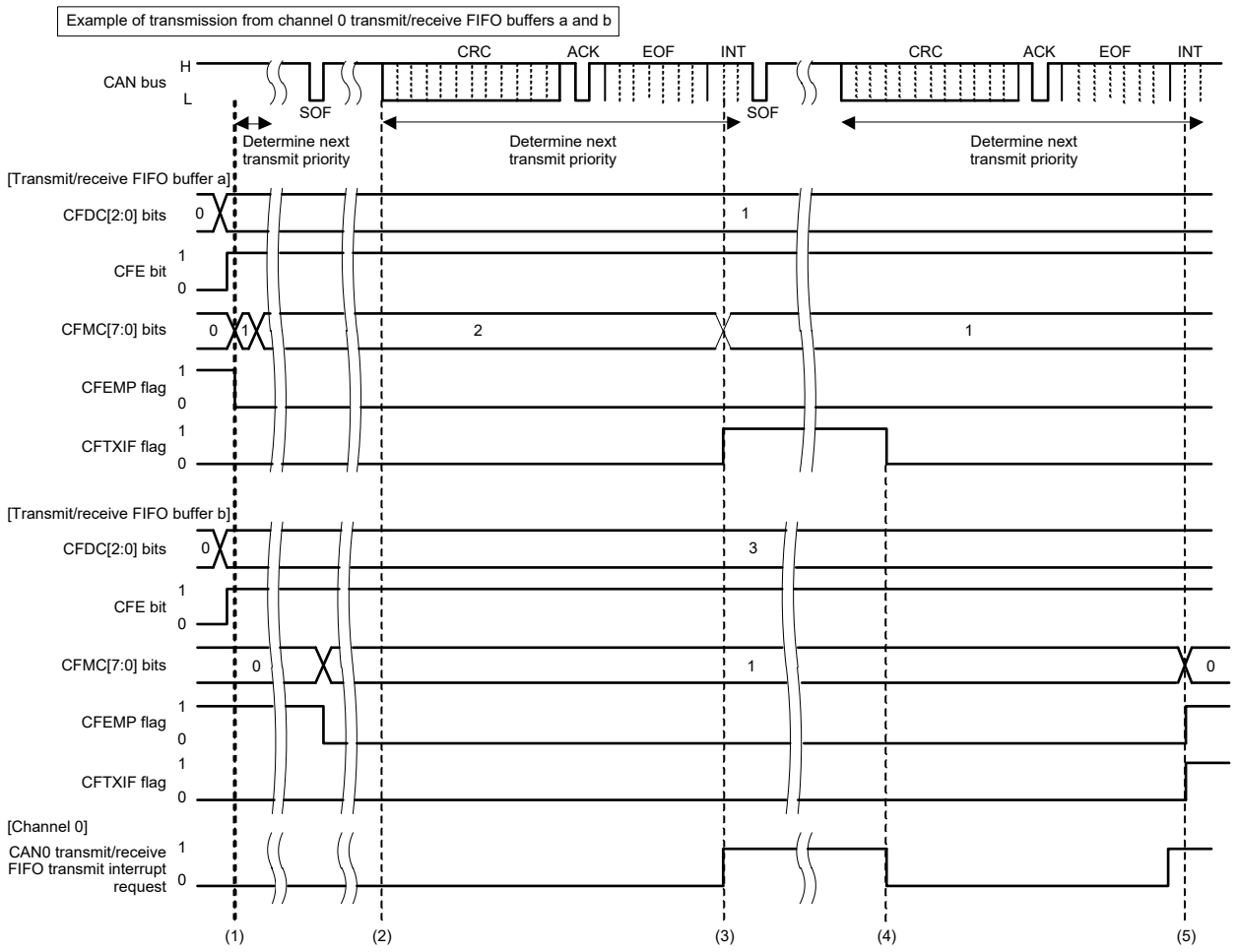
Note: k = 0 to 11, d = 0 or 1 (classical CAN mode), 0 to 15 (CAN FD mode)

Figure 14.30 Procedure for Transmission from Transmit/Receive FIFO Buffers

When storing a message, do not write a value to the RSCFDnCFDCFDFd_k register corresponding to the area exceeding the payload storage size specified by the CFPLS[2:0] bits in the RSCFDnCFDCFCCk register.

Table 14.187 Payload Storage Area of Transmit/Receive FIFO Buffer

Set CFPLS[2:0] Value	Payload Storage Size	Corresponding Data Field Registers
000_B	8 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF1_k
001_B	12 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF2_k
010_B	16 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF3_k
011_B	20 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF4_k
100_B	24 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF5_k
101_B	32 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF7_k
110_B	48 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF11_k
111_B	64 bytes	RSCFDnCFDCFDF0_k to RSCFDnCFDCFDF15_k



Note: a = 0 to 2, b = 0 to 2
CFDC[2:0], CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0], CFEMP, CFTXIF: Flags in the RSCFDn(CFD)CFSTSk register

Figure 14.31 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) While the CAN bus is idle, when the CFE bit in the RSCFDn(CFD)CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCa register is 001_B (4 messages) or more and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is 01_H or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

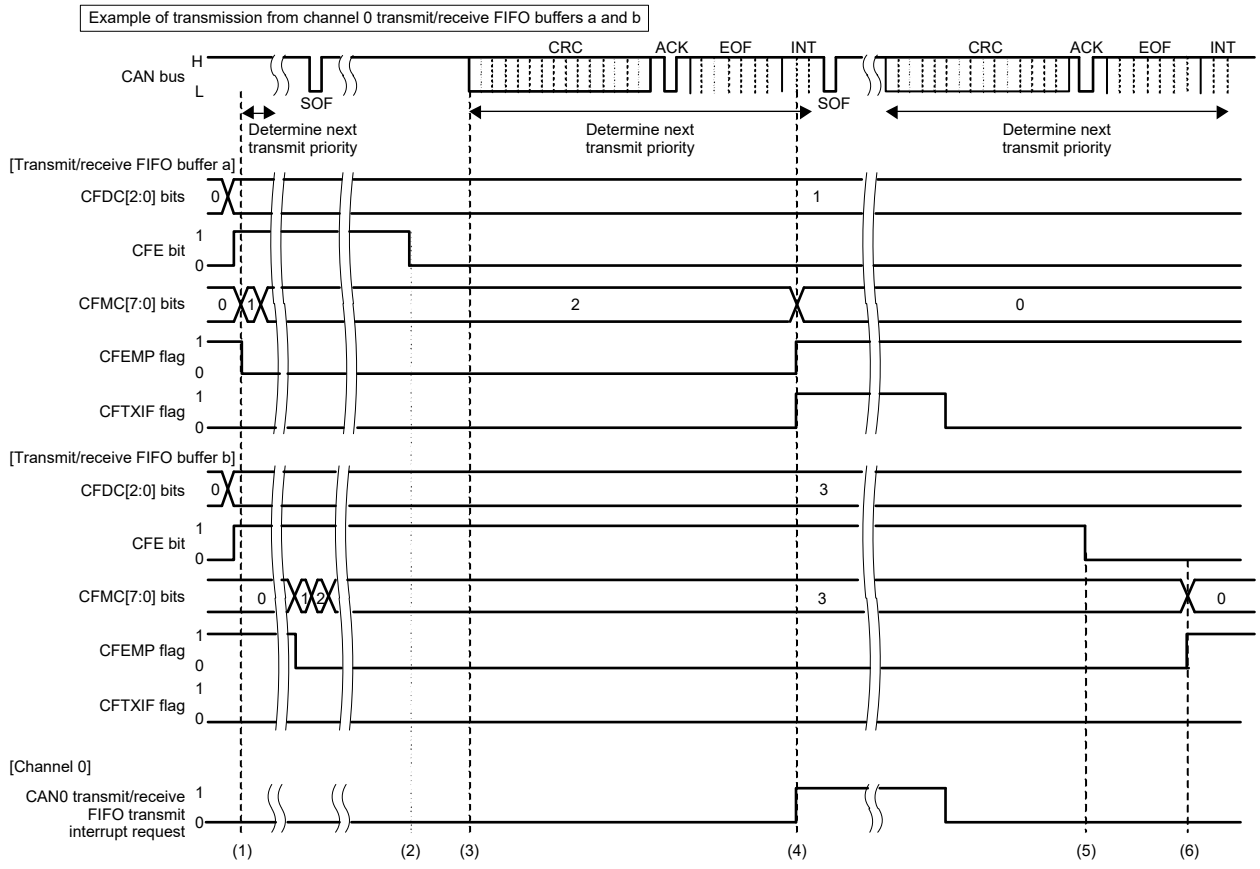
(2) When a transmit request from a buffer is present, the priority determination starts at the first bit of the CRC field for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(3) When transmission completes successfully, the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is decremented. Setting the CFIM bit in the RSCFDn(CFD)CFCCa register to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCFDn(CFD)CFSTSa register to 1 (a transmit/receive FIFO transmit interrupt request is present).

(4) The program can clear the CFTXIF flag.

(5) Message transmission from transmit/receive FIFO buffer b of channel 0 has been completed and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSb register is decremented. The CFMC[7:0] bits are cleared to 00_H and therefore the CFEMP flag in the RSCFDn(CFD)CFSTSb register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

Transmission is continued until the CFEMP flag is set to 1. It is possible to continuously store transmit messages in FIFO buffers until the CFFLL flag in the RSCFDn(CFD)CFSTSa and RSCFDn(CFD)CFSTSb register is set to 1 (the transmit/receive FIFO buffer is full).



Note: a = 0 to 2, b = 0 to 2
CFDC[2:0], CFE: Bits in the RSCFDn(CFD)CFCCk register
CFMC[7:0] , CFEMP, CFTXIF: Flags in the RSCFDn(CFD)CFSTSk register

Figure 14.32 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmit Abort Completed)

(1) While the CAN bus is idle, when the CFE bit in the RSCFDn(CFD)CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCFDn(CFD)CFCCa register is 001_B (4 messages) or more and the CFMC[7:0] value in the RSCFDn(CFD)CFSTSa register is 01_H or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

(2) When transmission is in progress or it is determined that the transmit/receive FIFO buffer is used for the next transmission, message transmission is not aborted unless an error or arbitration loss occurs even if the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

(3) When a transmit request from a buffer is present, the priority determination starts at the first bit of the CRC field for the next transmission. In this figure, transmit/receive FIFO buffer b is not selected as a buffer for the next transmission. The determination time may delay if the transmit priority determination processing is performed on another channel. However, the delay does not occur during transmission because the determination processing is completed by the third bit of the intermission.

(4) When transmit completes successfully, the CFMC[7:0] value is cleared to 00_H. Setting the CFIM bit to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCFDn(CFD)CFSTSa register to 1 (a transmit/receive FIFO transmit interrupt request is present). The program can clear the CFTXIF flag.

(5) If another CAN node on the CAN bus is transmitting data (not from transmit/receive FIFO buffer b), transmit/receive FIFO buffer b cannot be disabled immediately even if the CFE bit in the RSCFDn(CFD)CFCCb register is cleared to 0 (no transmit/receive FIFO buffer b is used) during transmit priority determination. (The CFEMP flag in the RSCFDn(CFD)CFSTSb register is not set to 1 (the transmit/receive FIFO buffer b contains no message (buffer empty)) immediately.)

(6) After the internal processing time has passed, transmit/receive FIFO buffer b are disabled and the CFMC[7:0] bits in the RSCFDn(CFD)CFSTSb register are cleared to 00_H and the CFEMP flag is set to 1. When the transmit/receive FIFO buffer b is not transmitting data and is not selected as the next transmit buffer and priority determination is not in progress, the transmit/receive FIFO buffer b is immediately disabled. (The CFMC[7:0] bits are cleared to 00_H and the CFEMP flag is set to 1.)

14.11.3.3 Procedure for Transmission from the Transmit Queue

Figure 14.33 shows the procedure for transmission from the transmit queue.

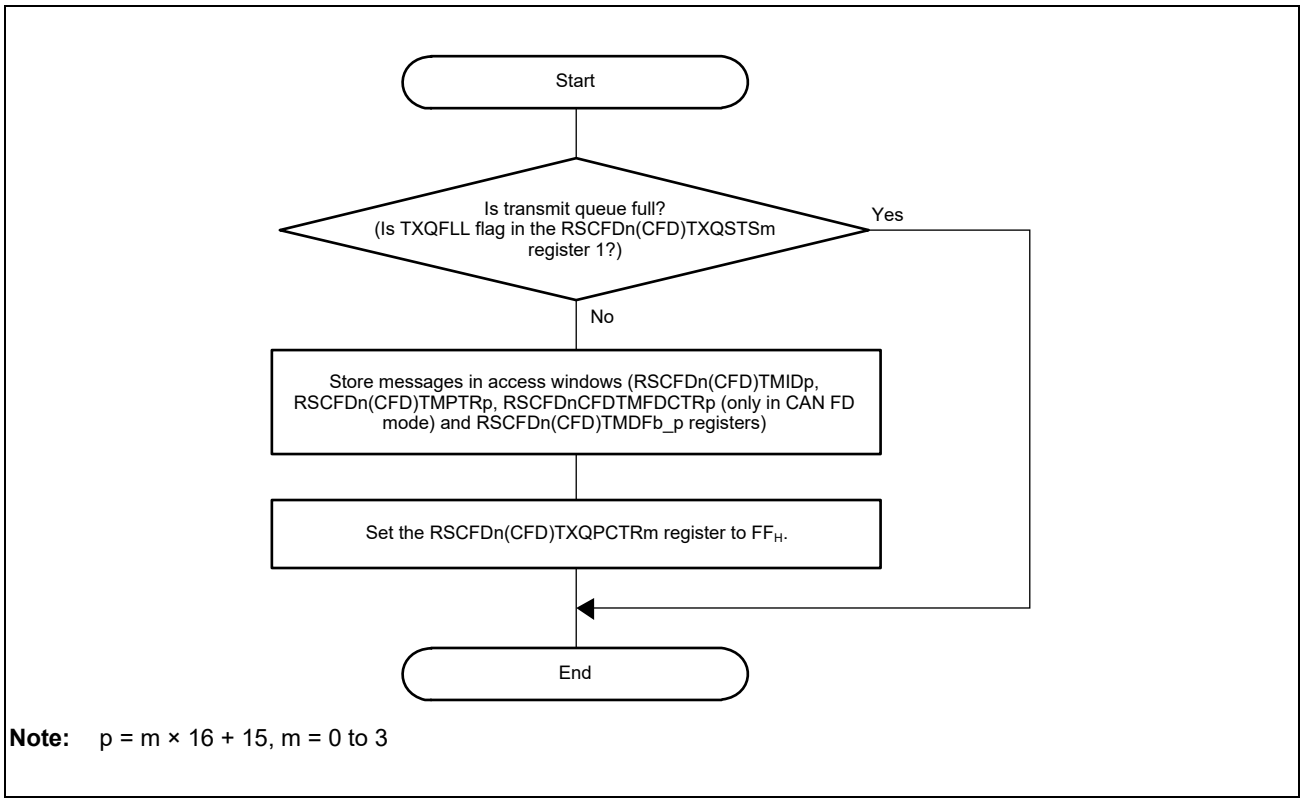
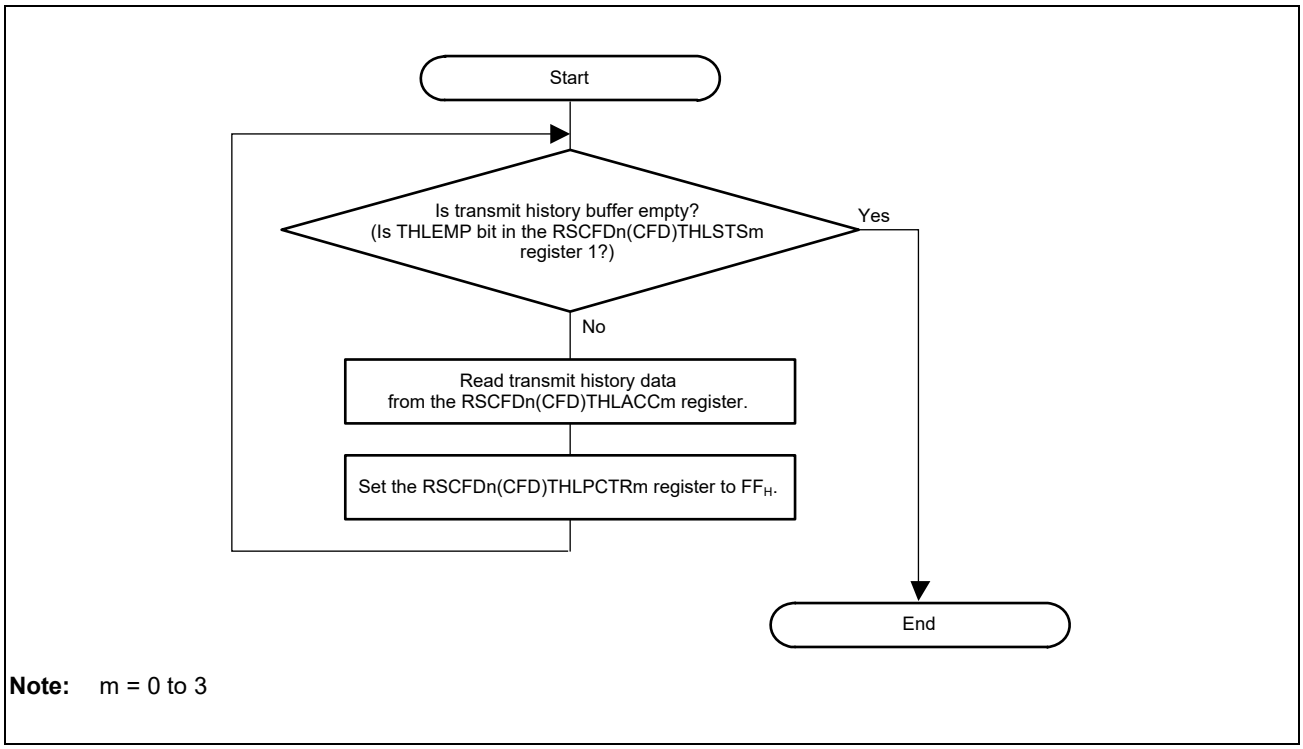


Figure 14.33 Procedure for Transmission from the Transmit Queue

14.11.3.4 Transmit History Buffer Reading Procedure

Transmit history data can be read from the RSCFDn(CFD)THLACCm register. The next data can be accessed by writing FF_H to the corresponding RSCFDn(CFD)THLPCTRm register (m = 0 to 3) after reading a set of data. **Figure 14.34** shows the transmit history buffer reading procedure.



Note: m = 0 to 3

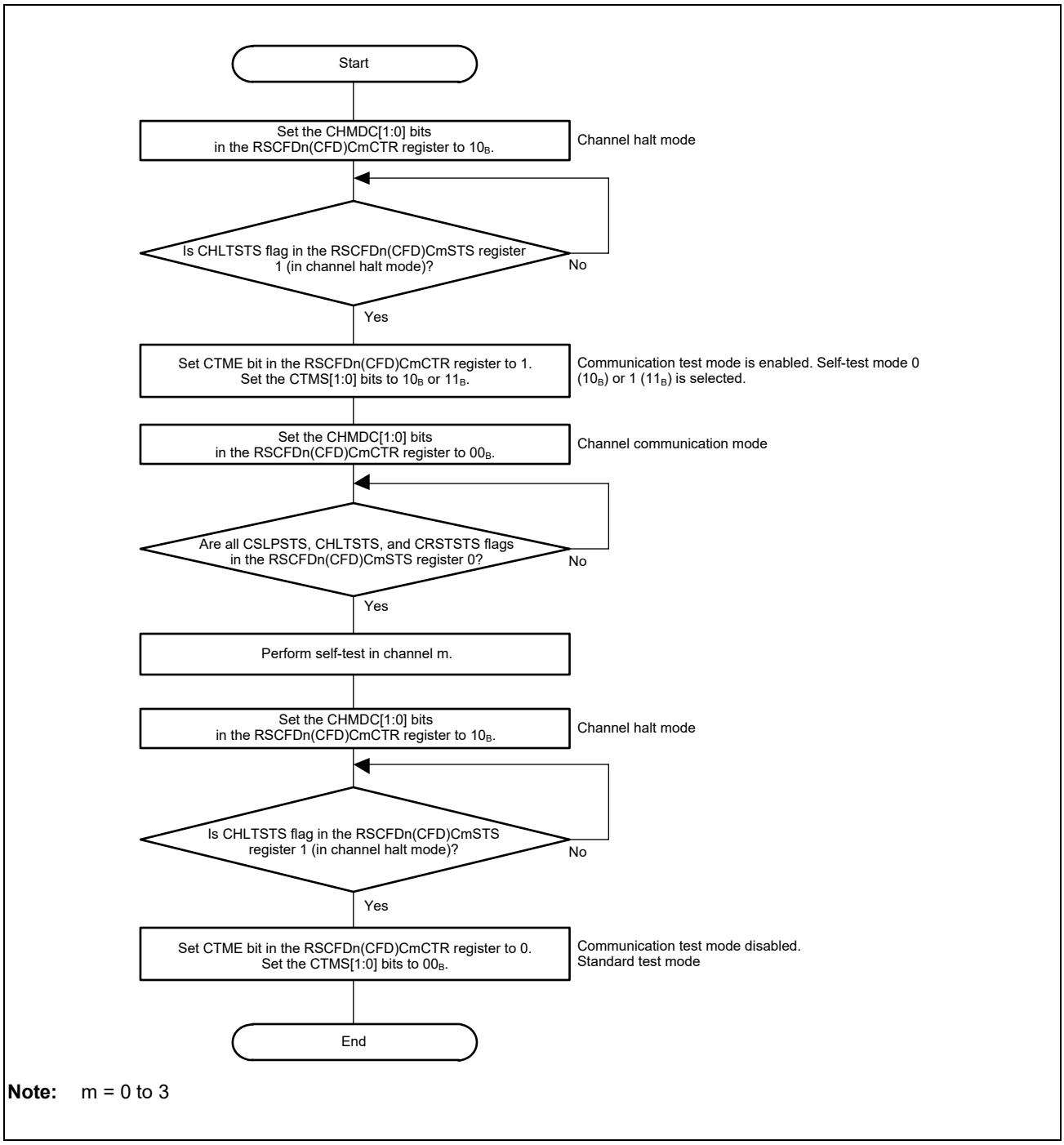
Figure 14.34 Transmit History Buffer Reading Procedure

14.11.4 Test Settings

14.11.4.1 Self-Test Mode Setting Procedure

Self-test mode allows communication test on a channel basis by enabling a CAN node to receive its own transmitted messages.

Figure 14.35 shows the self-test mode setting procedure.



Note: m = 0 to 3

Figure 14.35 Self-Test Mode Setting Procedure

14.11.4.2 Procedure for Releasing the Protection

Since the global test function in **Table 14.188** is protected, write the protection release data 1 and release data 2 in succession to the LOCK[15:0] bits in the RSCFDn(CFD)GLOCKK register, then set the target test bit to 1.

Table 14.188 Protection Release Data for Test Function

Test Function	Protection Release Data 1	Protection Release Data 2	Target Bit
RAM test	7575_H	$8A8A_H$	RTME bit in the RRSCFDn(CFD)GTSTCTR register

If an incorrect value is written to the LOCK[15:0] bits, restart from writing the protection release data 1. **Figure 14.36** shows the procedure for releasing the protection.

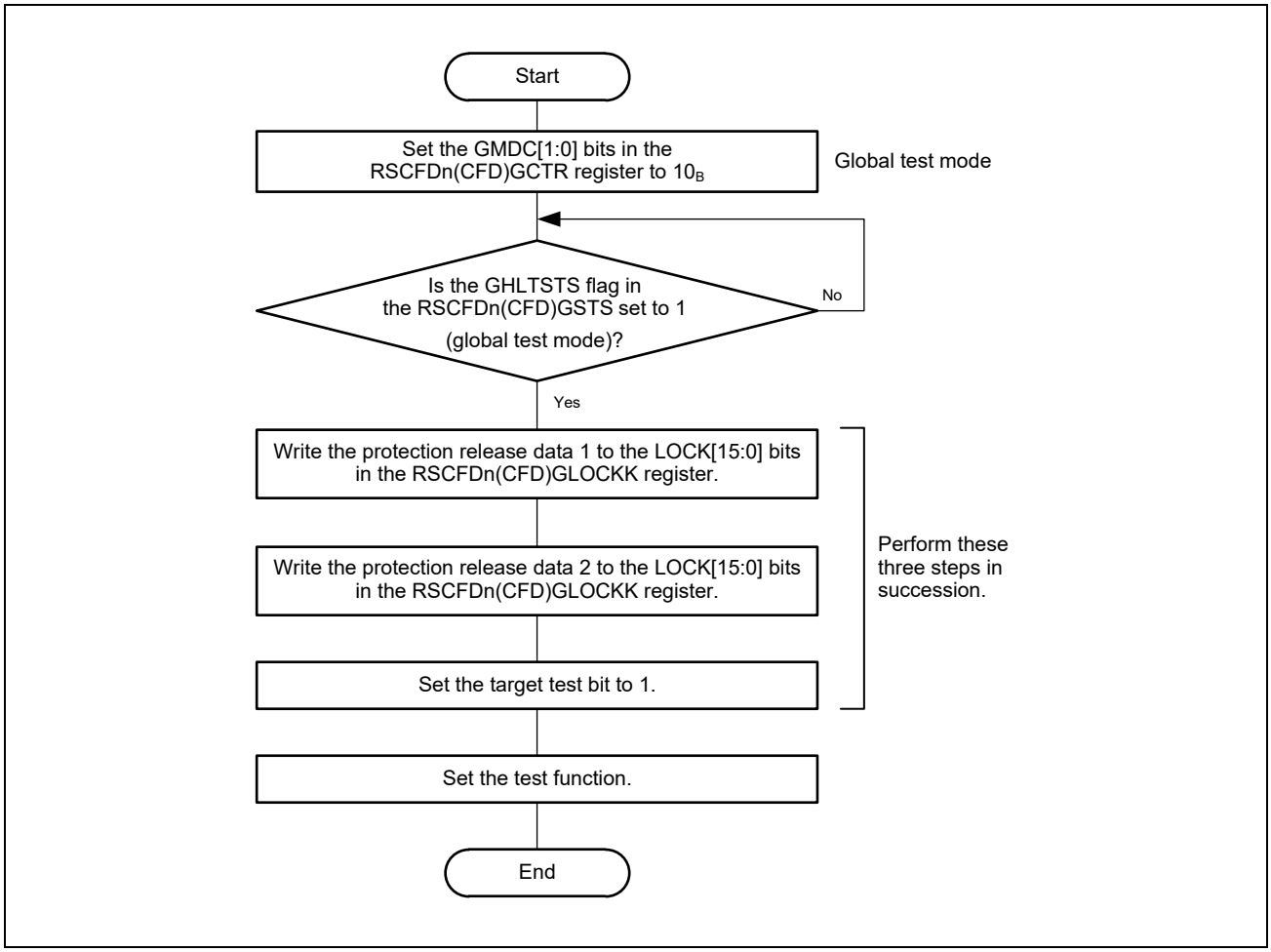
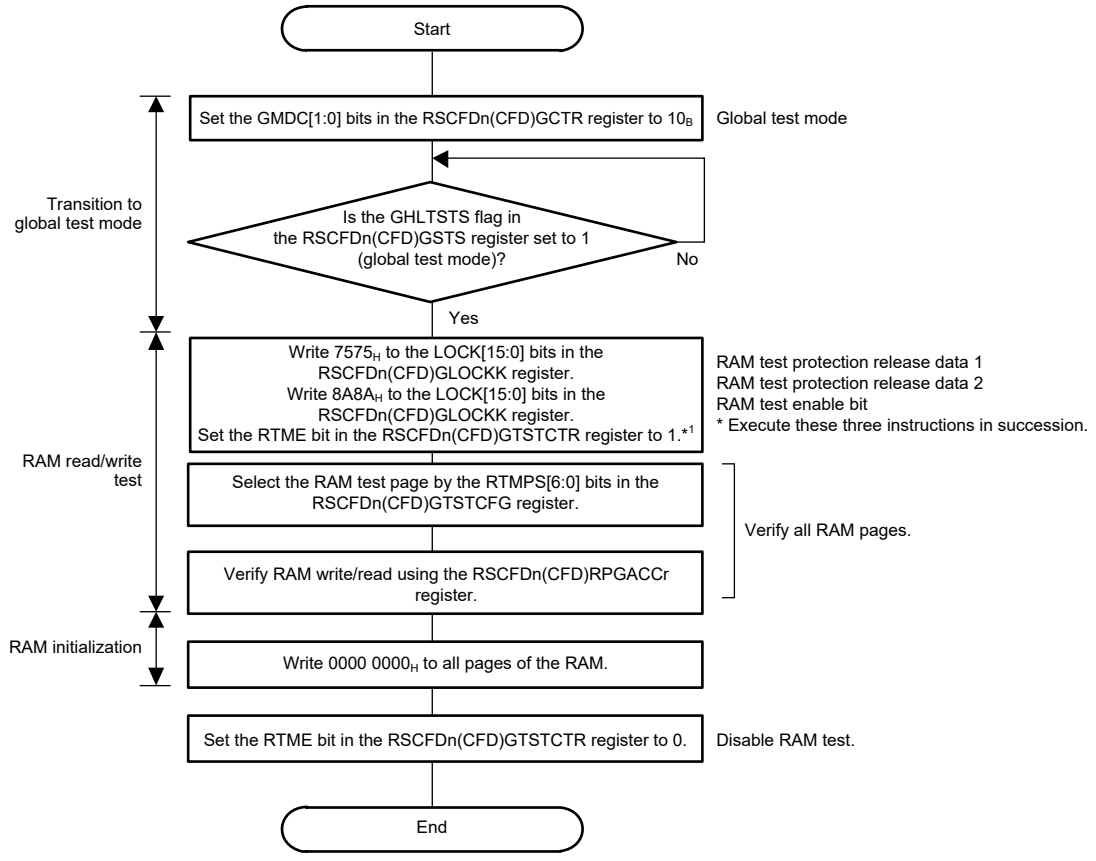


Figure 14.36 Protection Release Procedure

14.11.4.3 RAM Test Setting Procedure

RAM tests include CAN RAM read/write test. The read/write test verifies that data written to the RAM is read correctly. Before closing the RAM test, write 0000 0000_H to all pages of the CAN RAM.

Figure 14.37 shows the RAM test setting procedure.



Note 1. Perform the following operations before the transition to the RAM test.
- Canceling the transmit request
- Disabling the FIFO buffers and transmit queues
- Clearing the flags for the receive buffers

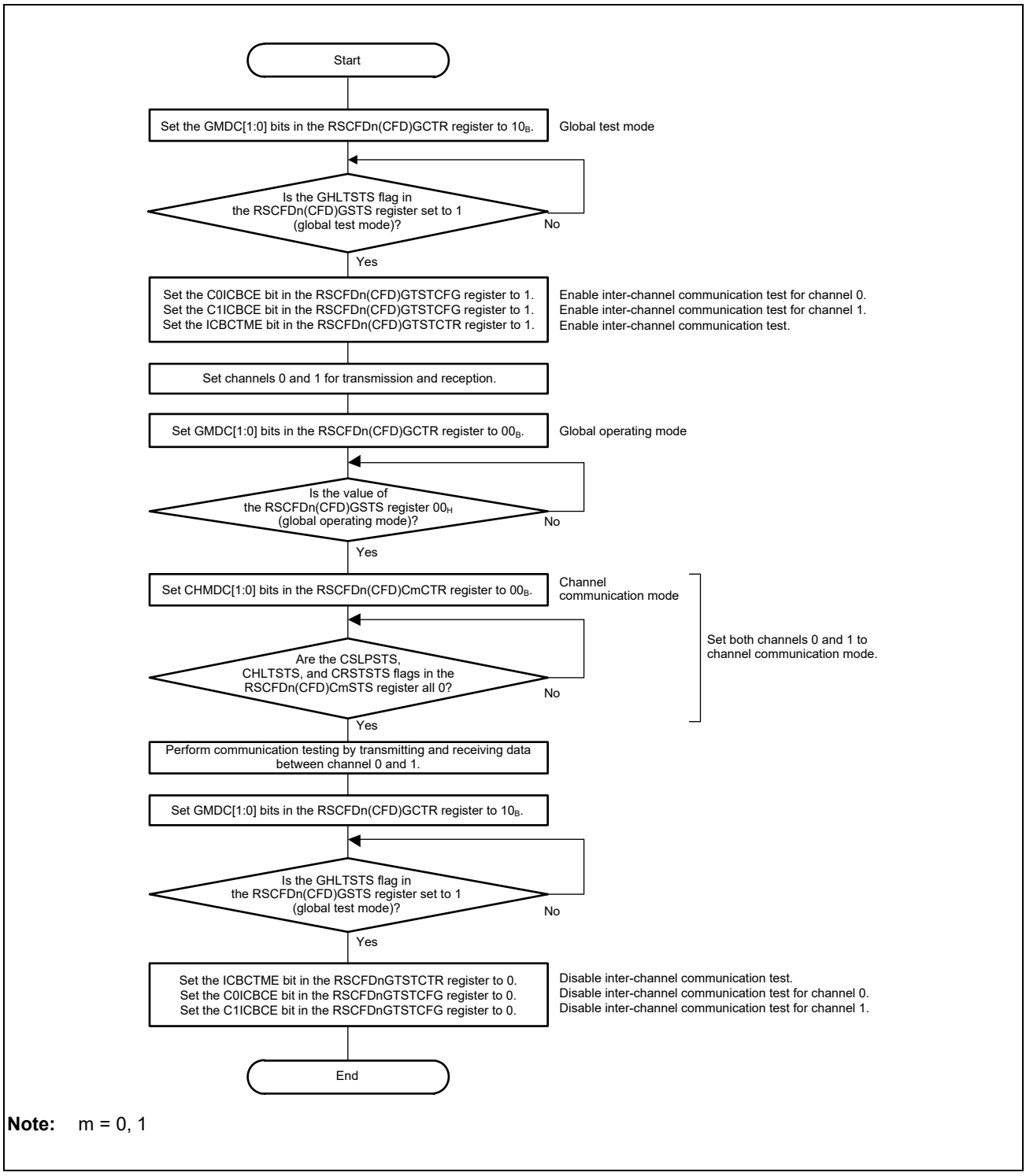
Note: r = 0 to 63

Figure 14.37 RAM Test Setting Procedure

14.11.4.4 Inter-Channel Communication Test Setting Procedure

Communication testing can be performed by transmitting and receiving data between different channels.

Figure 14.38 shows the inter-channel communication test setting procedure.



Note: m = 0, 1

Figure 14.38 Inter-Channel Communication Test Setting Procedure (Example of Communication Test between Channel 0 and Channel 1)

14.12 Notes on the RS-CANFD Module

- When changing interface mode without resetting the RS-CANFD, write the value after reset to all registers and bits that are not allocated to the register map after change and then modify the RSCFDn(CFD)GRMCFG register.
- When changing a global mode, check the GSLPSTS, GHLTSTS, and GRSTSTS flags in the RSCFDn(CFD)GSTS register for transitions. When changing a channel mode, check the CSLPSTS, CHLTSTS, and CRSTSTS flags in the RSCFDn(CFD)CmSTS register (m = 0 to 3) for transitions.
- When only classical CAN frames are used in CAN FD mode, set the RSCFDnCFDCmDCFG register to the value equal to the set RSCFDnCFDCmNCFG register value.
- The acceptance filter processing checks receive rules sequentially in ascending order from the minimum rule number. If the same ID, IDE bit, or RTR bit value is set for multiple receive rules, the minimum number of receive rule is used for the acceptance filter processing. If the message does not pass through the subsequent DLC filter processing, the data processing is terminated without returning to the acceptance filter processing and the message is not stored in the buffer.
- When linking transmit buffers to transmit/receive FIFO buffers or allocating transmit buffers to transmit queues, set the control register (RSCFDn(CFD)TMCp) of the corresponding transmit buffer to 00_H. The status register (RSCFDn(CFD)TMSTSp) of the corresponding transmit buffer should not be used. Flags in other status registers (registers RSCFDn(CFD)TMTRSTS0, RSCFDn(CFD)TMTRSTS1, RSCFDn(CFD)TMTARSTS0, RSCFDn(CFD)TMTARSTS1, RSCFDn(CFD)TMTCSTS0, RSCFDn(CFD)TMTCSTS1, RSCFDn(CFD)TMTASTS0 and RSCFDn(CFD)TMTASTS1), which correspond to transmit buffers linked to transmit/receive FIFO buffers or allocated to transmit queues remain unchanged. Set the enable bit in the corresponding interrupt enable register (registers RSCFDn(CFD)TMIEC0 and RSCFDn(CFD)TMIEC1) to 0 (transmit buffer interrupt is disabled).
- When using transmit buffer merge mode (in CAN FD mode), write 00_H to the control register (RSCFDn(CFD)TMCp) of the transmit buffer corresponding to the transmit buffer allocated as a payload storage area. Set the enable bit of corresponding interrupt enable registers (RSCFDn(CFD)TMIEC0 and RSCFDn(CFD)TMIEC1) to 0 (to disable interrupts).
- Transmit buffers that are linked to transmit/receive FIFO buffers must not bet allocated to transmit queues. Do not allocate a transmit buffer allocated as a payload storage area in transmit buffer merge mode (in CAN FD mode) to the transmit queue either.
- Only a single transmit/receive FIFO buffer can be linked to a transmit buffer. Do not link two or more transmit/receive FIFO buffers to transmit buffers of the same number.
- When the CANm bit time clock is selected as a timestamp counter clock source, the timestamp counter stops when the corresponding channel has transitioned to channel reset mode or channel halt mode.
- In case of an attempt to store a new received message when the receive FIFO buffer and the transmit/receive FIFO buffer are full, the new message is discarded. If you wish to store a new transmit message in the transmit/receive FIFO buffer or the transmit queue, check that the transmit/receive FIFO buffer or the transmit queue is not full.

- In the case of registers that access the RAM, the value after reset shown in **Section 14.3, Registers (Classical CAN Mode)** and **Section 14.4, Registers (CAN FD Mode)** indicate the values cleared by initialization of the CAN RAM. Values before clear are undefined. The following registers apply.
  - Receive rule (RSCFDn(CFD)GAFLIDj, RSCFDn(CFD)GAFLMj, RSCFDn(CFD)GAFLP0_j, RSCFDn(CFD)GAFLP1_j registers)
  - Receive buffers (RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, RSCFDn(CFD)RMDFb_q registers)
  - Receive FIFO buffer access registers (RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx, RSCFDn(CFD)RFDFd_x registers)
  - Transmit/receive FIFO buffer access registers (RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk, RSCFDn(CFD)CFDFd_k registers)
  - Transmit buffers (RSCFDn(CFD)TMIDp, RSCFDn(CFD)TMPTRp, RSCFDnCFDTMFDCTRp, RSCFDn(CFD)TMDFb_p registers)
  - Transmit history access register (RSCFDn(CFD)THLACCm registers)
  - RAM test page access register (RSCFDn(CFD)RPGACCr registers)
- The values of unused receive buffers (RSCFDn(CFD)RMIDq, RSCFDn(CFD)RMPTRq, RSCFDnCFDRMFDSTSq, and RSCFDn(CFD)RMDFb_q registers), receive FIFO buffer access registers (RSCFDn(CFD)RFIDx, RSCFDn(CFD)RFPTRx, RSCFDnCFDRFFDSTSx, and RSCFDn(CFD)RFDFd_x registers) and transmit/receive FIFO buffer access registers (RSCFDn(CFD)CFIDk, RSCFDn(CFD)CFPTRk, RSCFDnCFDCFFDCSTSk, and RSCFDn(CFD)CFDFd_k registers) are undefined when the RS-CANFD module transitions to global operation mode or global test mode after exiting from global reset mode.

Section 15 Single Edge Nibble Transmission (RSENT)

This section describes the Single Edge Nibble Transmission (RSENT) module.

The first subsection describes the RH850/C1M-A-specific features such as the number of units and register base address. The subsequent subsections describe the functions and registers of the RSENT.

15.1 Features of the RH850/C1M-A RSENT

15.1.1 Number of Units

This product contains the number of RSENT units listed below.

Table 15.1 Number of Units

Product Name	RH850/C1M-A
Number of units	4
Name	RSENTn (n = 0 to 3)

Index n

This section identifies each RSENT units by “n” (n = 0 to 3). For example, the RSENT timestamp register is described as RSENTnTSPC.

15.1.2 Register Base Addresses

RSENT register base addresses are represented by an offset from the base address.

The following table shows the base address of each RSENT module.

Table 15.2 Register Base Addresses

Base Address Name	Base Address
<RSENT0_base>	FFDB 0000_H
<RSENT1_base>	FF7B 0080_H
<RSENT2_base>	FFDB 0100_H
<RSENT3_base>	FF7B 0180_H

15.1.3 Clock Supply

The following clock input is supplied for the RSENT module.

Table 15.3 RSENT Clock Source

Unit Name	Name of Unit Clock	Name of Supplied Clock
RSENTn	clkc	Non-modulated low-speed clock CLKC_LSB

15.1.4 Interrupt Requests

The RSENT module can generate the following interrupt requests.

Table 15.4 RSENT Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt No.	DMA Trigger No.*1		DTS Trigger No.*1	
			1st	2nd	1st	2nd
RSENT0						
INT_SENT_TX	RSENT status interrupt	322	—	—	—	—
INT_SENT_RX	RSENT receive interrupt	321	—	45	—	45
RSENT1						
INT_SENT_TX	RSENT status interrupt	324	—	—	—	—
INT_SENT_RX	RSENT receive interrupt	323	—	46	—	46
RSENT2						
INT_SENT_TX	RSENT status interrupt	326	—	—	—	—
INT_SENT_RX	RSENT receive interrupt	325	—	47	—	47
RSENT3						
INT_SENT_TX	RSENT status interrupt	328	—	—	—	—
INT_SENT_RX	RSENT receive interrupt	327	—	48	—	48

—: Not assigned.

Note 1. 1st: Primary Channel, 2nd: Secondary Channel

15.1.5 Reset Factor

The reset factors of the RSENT module are listed below. The RSENT module is initialized by the following reset factor.

Table 15.5 Reset Factor

Unit Name	Reset Factor
RSENTn	All reset factors

15.1.6 External Input/Output Signals

The following table shows the RSENT external input/output signals.

Table 15.6 External Input/Output Signals

Unit Signal Name	Outline	Port Pin Name
RSENT0		
sent_rx	RSENT data input	RSENT0RX
sent_spc	RSENT SPC control output	RSENT0SPCO
RSENT1		
sent_rx	RSENT data input	RSENT1RX
sent_spc	RSENT SPC control output	RSENT1SPCO
RSENT2		
sent_rx	RSENT data input	RSENT2RX
sent_spc	RSENT SPC control output	RSENT2SPCO
RSENT3		
sent_rx	RSENT data input	RSENT3RX
sent_spc	RSENT SPC control output	RSENT3SPCO

15.2 Outline

15.2.1 Function Outline

The RSENT interface supports the following standard specification (SAE J2716 version JAN2010) functions:

- Triple speed expansion Tick Time: Clock cycle (1 µs to 90 µs)
- Variable data transmission rate
  - 24.7 kbps to 64.9 kbps: 3 clock rate 6 nibble data
  - 74.1 kbps to 194.7 kbps: 1 clock rate 6 nibble data
- Unidirectional communication: Between the sensor and MCU
- Bidirectional communication: Between the sensor and MCU (supported in SPC mode)
- Single edge data transmission: Coded by the temporal distance of two serially-detected falling edges on a data line
- Transmission frame with up to 6 data nibbles + status and communication nibbles
- Data transmission protected with CRC is available.
  - CRC data can be read with the RSENTnSRXD.SCRC bits.
- Calibration phrase in each data frame (RSENTnCPL.CPLV bits)
- Compatible with 3 systems, i.e., SENT standard (1-wire system), SPC extension (1-wire system), and SPC extension (2-wire system).
- Multiple sensors can connect to the RSENT channel that has the standard expansion function. Received data from sensors is detected by software or DMA.
- Timestamp function: Master can only be set for RSENT0.
  For others, slave can only be set (RSENTnTSPC.TMS bit)

15.2.2 Block Diagram

The following figure shows a block diagram of the RSENT module.

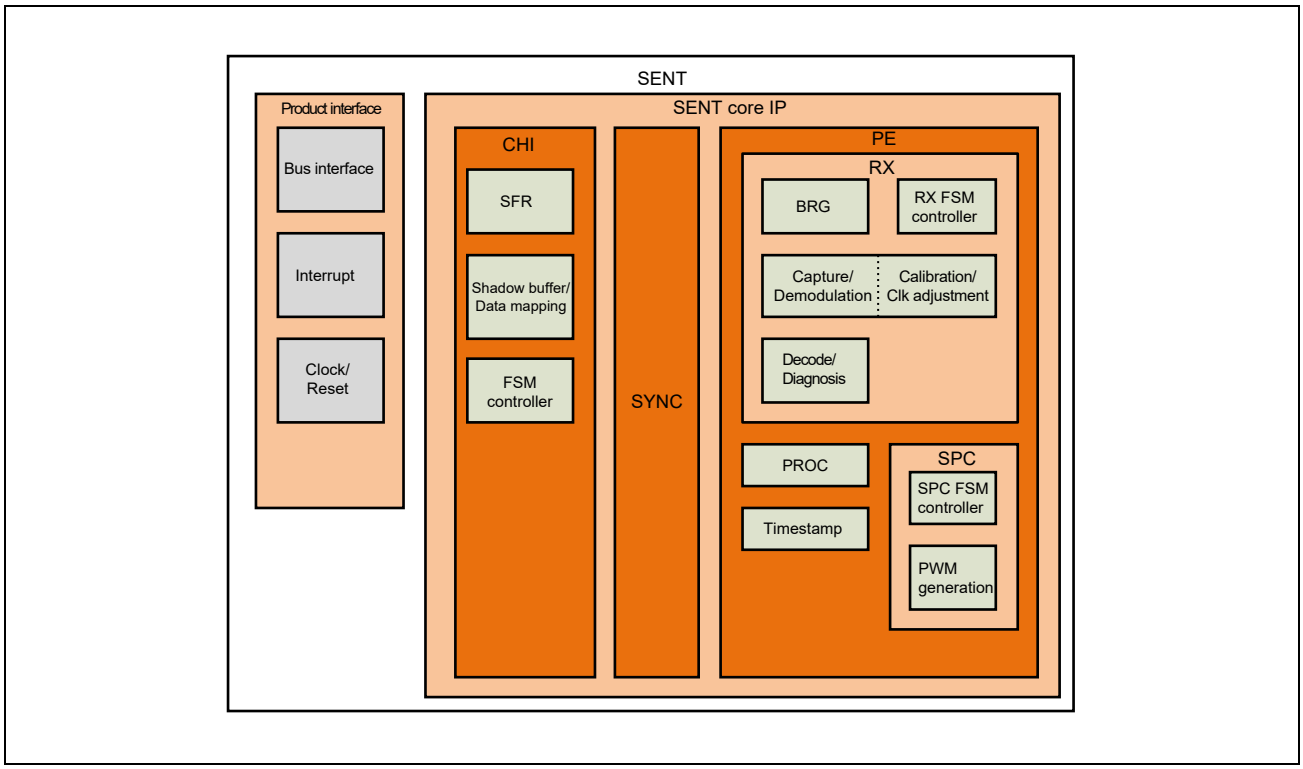


Figure 15.1 Block Diagram of RSENT

15.3 Registers

15.3.1 List of Registers

RSENTn is controlled and operated by the following registers.

For <RSENTn_base>, see **Section 15.1.2, Register Base Addresses.**

Table 15.7 Overview of RSENTn Registers

Module Name	Register Name	Abbreviation	Address
RSENTn	RSENT timestamp register	RSENTnTSPC	<RSENTn_base> + 0000_H
RSENTn	RSENT timestamp counter register	RSENTnTSC	<RSENTn_base> + 0004_H
RSENTn	RSENT communication configuration register	RSENTnCC	<RSENTn_base> + 0010_H
RSENTn	RSENT baud rate prescaler register	RSENTnBRP	<RSENTn_base> + 0014_H
RSENTn	RSENT interrupt/DMA enable register	RSENTnIDE	<RSENTn_base> + 0018_H
RSENTn	RSENT mode control register	RSENTnMDC	<RSENTn_base> + $001C_H$
RSENTn	RSENT SPC transmission register	RSENTnSPCT	<RSENTn_base> + 0020_H
RSENTn	RSENT mode status register	RSENTnMST	<RSENTn_base> + 0024_H
RSENTn	RSENT communication status register	RSENTnCS	<RSENTn_base> + 0028_H
RSENTn	RSENT communication status clear register	RSENTnCSC	<RSENTn_base> + $002C_H$
RSENTn	RSENT slow channel receive timestamp register	RSENTnSRTS	<RSENTn_base> + 0030_H
RSENTn	RSENT slow channel receive data register	RSENTnSRXD	<RSENTn_base> + 0034_H
RSENTn	RSENT calibration pulse length register	RSENTnCPL	<RSENTn_base> + 0038_H
RSENTn	RSENT message length register	RSENTnML	<RSENTn_base> + $003C_H$
RSENTn	RSENT fast channel receive timestamp register	RSENTnFRTS	<RSENTn_base> + 0040_H
RSENTn	RSENT fast channel receive data register	RSENTnFRXD	<RSENTn_base> + 0044_H

15.3.2 RSENTnTSPC — RSENT Timestamp Register

Access: This register can be read/written in 32-bit units.

Address: <RSENTn_base> + 0000_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TMS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TTM[6:0]							—	TTPV[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.8 RSENTnTSPC Register Contents

Bit Position	Bit Name	Function
31 to 17	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
16	TMS	Timestamp Mode Selection 0: Master mode 1: Slave mode
15	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
14 to 8	TTM[6:0]	Timestamp Tick Multiplier 0000000_B: 1 0000001_B: 2 0000010_B: 3 : 1111111_B: 128
7	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
6 to 0	TTPV[6:0]	Timestamp Tick Prescaler Value 0000000_B: 1 0000001_B: 2 0000010_B: 3 : 1111111_B: 128

RSENTnTSPC.TMS (Timestamp Mode Selection)

When this bit is set to 0, the timestamp counter operates in master mode.

Writing 0000 0000_H to RSENTnTSC by the module set as the master leads to clearing of the timestamps in that module and in the channels set as slaves.

The CPU can only write to this bit if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

A RSENT module running in the slave mode needs to have the same timestamp counter prescaler value as the RSENT module running as its master.

NOTE

In this product series, RSENT0 is only the channel which can be set as a master.

Before synchronizing the timestamp counter, be sure to set RSENT0 and other channels as slave mode and slave mode, respectively.

Synchronization of the timestamp counter when the above setting is not made is not guaranteed.

RSENTnTSPC.TTM (Timestamp Tick Multiplier)

These bits define the multiplication value of the 1-µs time tick used for the timestamp counter. For timestamp clock configuration, see **Section 15.4.2.1, Timestamp**.

The CPU can only write to these bits if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTnTSPC.TTPV (Timestamp Tick Prescaler Value)

These bits define the prescaler value to generate a 1-µs clock tick.

For timestamp clock configuration, see **Section 15.4.2.1, Timestamp.**

The CPU can only write to these bits if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

The CPU should configure this value in such a way that, based on the supplied communication clock, a 1-µs clock tick is generated.

15.3.3 RSENTnTSC — RSENT Timestamp Counter Register

Access: This register can be read/written in 32-bit units.

Address: <RSENTn_base> + 0004_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TS[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.9 RSENTnTSC Register Contents

Bit Position	Bit Name	Function
31 to 0	TS[31:0]	Timestamp counter value

RSENTnTSC.TS (Timestamp)

These bits indicate the current timestamp counter value.

The CPU can set this bit to a value other than $0000\ 0000_H$ only when the RSENT module is in CONFIGURATION mode (RSENTnMST.OMS = 001_B).

When the timestamp counter is configured to operate in slave mode (RSENTnTSPC.TMS = 1), writing to this register has no effect when the RSENT module is in OPERATION IDLE or OPERATION ACTIVE mode (the RSENTnMST.OMS bits are either 011_B or 101_B).

The timestamp counter is incremented on every timestamp counter tick (as configured in the RSENTnTSPC.TTPV and RSENTnTSPC.TTM bits) when the RSENT module is in OPERATION IDLE or OPERATION ACTIVE mode (the RSENTnMST.OMS bits are either 011_B or 101_B).

When the timestamp counter is configured to operate in master mode (RSENTnTSPC.TMS = 0), the CPU writes $0000\ 0000_H$ to these bits and RSENTnTSC.TS is set to $0000\ 0000_H$.

When the slave mode setting is made for the timestamp counter (RSENTnTSPC.TMS = 1), writing $0000\ 0000_H$ to the RSENT0TSC.TS bit leads to the RSENTnTSC.TS bits being set to $0000\ 0000_H$.

For timestamp mode selection, see **Section 15.4.2.1, Timestamp**.

15.3.4 RSENTnCC — RSENT Communication Configuration Register

Access: This register can be read/written in 32-bit units.

Address: <RSENTn_base> + 0010_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SOPC	FCM	SCCD	FCCD	DCF	SMF[1:0]		PPTC	PPC	NDN[2:0]			SPCE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.10 RSENTnCC Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 13	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
12	SOPC	SPC Output Polarity Control 0: SPC pulse active high 1: SPC pulse active low
11	FCM	Frame Check Method0: 0: Check against next calibration pulse 1: Check against previous calibration pulse
10	SCCD	Slow Channel CRC Check 0: Slow channel CRC check enabled 1: Slow channel CRC check disabled
9	FCCD	Fast Channel CRC Check 0: Fast channel CRC check enabled 1: Fast channel CRC check disabled
8	DCF	Data Nibble CRC Format Select 0: CRC format defined in SAE J2716 JAN2010 is selected 1: CRC format prior to SAE J2716 FEB2008 is selected
7, 6	SMF[1:0]	Serial Message Format 00_B: No serial message extraction 01_B: Short serial message format 10_B: Enhanced serial message format 11_B: Setting prohibited
5	PPTC	Pause Pulse Type Configuration 0: Pause pulse for variable message length 1: Pause pulse for fixed message length
4	PPC	Pause Pulse Configuration 0: Without pause pulse 1: With pause pulse

Table 15.10 RSENTnCC Register Contents (2/2)

Bit Position	Bit Name	Function
3 to 1	NDN[2:0]	Number of Data Nibbles 000_B: 1 data nibble 001_B: 2 data nibbles 010_B: 3 data nibbles 011_B: 4 data nibbles 100_B: 5 data nibbles 101_B: 6 data nibbles Other than above: Setting prohibited
0	SPCE	SPC Mode Enable 0: SPC mode disabled 1: SPC mode enabled

RSENTnCC.SOPC (SPC Output Polarity Control)

When this bit is set to 0, the SPC pulse is sent as an active high signal. The default output value is low level.

When this bit is set to 1, the SPC pulse is sent as an active low signal. The default output value is high level.

For SPC operation, see **Section 15.4.4, SPC Function**.

The CPU can write to this bit only when the RSENT module is in CONFIGURATION mode (RSENTnMST.OMS = 001_B).

NOTE

- In this product series, be sure to set this bit to select active low for use with a single-wire SPC system.
- Any change to this bit from the default value becomes effective on the output value when entering the OPERATION_ACTIVE mode (MST.OMS is 101_B). When entering RESET mode (MST.OMS is 000_B), the output level is set to the default value (low level).

RSENTnCC.FCM (Frame Check Method)

When this bit is set to 0, the current calibration pulse is compared to the next received calibration pulse.

The buffer update mechanism is operating according to the preferred option as described in SAE J2716 2010.

When this bit is set to 1, the current calibration pulse is compared to the previously received calibration pulse.

The buffer update mechanism is operating according to the second option as described in SAE J2716 2010 which should be only used if extra latency to process the second calibration pulse cannot be tolerated.

For buffer update timings, see also **Section 15.4.3.5, Fast Channel Message Reception.**

The CPU can only write to this bit if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTnCC.SCCD (Slow Channel CRC Check Disable)

When this bit is set to 1, the CRC check for the slow channel is disabled. In this case, messages are stored in the slow channel message reception buffer with the received CRC.

When this bit is set to 1, the RSENTnCS.SCS bit is not set.

The CPU can only write to this bit if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTnCC.FCCD (Fast Channel CRC Check Disable)

When this bit is set to 1, the CRC check for the fast channel is disabled. In this case, messages are stored in the fast channel message receive buffer with the received CRC.

When this bit is set to 1, the RSENTnCS.FCS bit is not set.

The CPU can only write to this bit if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTn.DCF (Data Nibble CRC Format Select)

This bit selects the data nibble CRC format from the format defined in SAE J2716 JAN2010 or an earlier format.

When this bit is set to 0, the CRC format defined in SAE J2716 JAN2010 is set. When this bit is set to 1, the CRC format defined prior to SAE J2716 FEB2008 is set.

For details of the CRC for data nibbles, see Sections *5.4.2.1* and *5.4.2.2* of *SAE J2716 JAN2010.*

The CPU can write to this bit only when the RSENT module is in CONFIGURATION mode (RSENTnMST.OMS = 001_B).

RSENTnCC.SMF (Serial Message Format)

These bits define the serial message format expected to be received for automatic extraction.

When these bits are set to 00_B, no serial message is extracted and the status and communication nibble are provided in the RSENTnSRXD register.

The CPU can only write to these bits if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

The CPU shall set these bits to 00_B when RSENTnCC.SPCE is set to 1 and more than one sensor is connected to the RSENT module.

RSENTnCC.PPTC (Pause Pulse Type Configuration)

This bit defines the pause pulse type.

The CPU can only write to this bit if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

The CPU must not set this bit to 1 when the RSENTnCC.PPC bit is set to 0.

RSENTnCC.PPC (Pause Pulse Configuration)

This bit defines the presence or absence of the pause pulse.

The CPU can only write to this bit if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTnCC.NDN (Number of Data Nibbles)

These bits define the number of data nibbles included in an RSENT message.

The CPU can only write to these bits if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTnCC.SPCE (SPC Mode Enable)

This bit enables SPC mode.

For details about SPC mode operation, see **Section 15.4.4, SPC Function**.

The CPU can only write to this bit if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

15.3.5 RSENTnBRP — RSENT Baud Rate Prescaler Register

Access: This register can be read/written in 32-bit units.

Address: <RSENTn_base> + 0014_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	TTF[3:0]				—	TTI[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SCDV[6:0]							—	—	—	SCMV[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 15.11 RSENTnBPR Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
27 to 24	TTF[3:0]	Time Tick Decimal Fraction 0000_B: 0.0 µs 0001_B: 0.1 µs 0010_B: 0.2 µs : 1000_B: 0.8 µs 1001_B: 0.9 µs Other than above: Setting prohibited
23	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
22 to 16	TTI[6:0]	Time Tick Integer 0000000_B: 1 µs 0000001_B: 2 µs 0000010_B: 3 µs : 1011000_B: 89 µs 1011001_B: 90 µs Other than above: Setting prohibited
15	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
14 to 8	SCDV[6:0]	Sample Clock Division Value 0000000_B: 1 0000001_B: 2 0000010_B: 3 : 1111110_B: 127 1111111_B: 128
7 to 5	Reserved	When read, these bits are read as 0. When writing, write the value after reset.

Table 15.11 RSENTnBPR Register Contents (2/2)

Bit Position	Bit Name	Function
4 to 0	SCMV[4:0]	Sample Clock Multiplication Value 00000_B: 1 00001_B: 2 00010_B: 3 : 11110_B: 31 11111_B: 32

RSENTnBRP.TTF (Time Tick Decimal Fraction)

These bits define the decimal part of the tick length in 0.1-µs granularity.

For tick length configuration, see **Section 15.4.2.2(2), RX and SPC Tick Settings**.

The CPU can only write to these bits if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTnBRP.TTI (Time Tick Integer)

These bits define the integer part of the tick length.

For tick length configuration, see **Section 15.4.2.2(2), RX and SPC Tick Settings**.

The CPU can only write to these bits if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTnBRP.SCDV (Sample Clock Division Value)

These bits define the division value for the sample clock generation logic.

For RSENTnBRP configuration, see **Section 15.4.2.2(1), RX BRP Setting**.

The CPU can only write to these bits if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

RSENTnBRP.SCMV (Sample Clock Multiplication Value)

These bits define the multiplication value for the sample clock generation logic.

For RSENTnBRP configuration, see **Section 15.4.2.2(1), RX BRP Setting**.

The CPU can only write to these bits if the RSENT module is in CONFIGURATION mode (the RSENTnMST.OMS bits are 001_B).

15.3.6 RSENTnIDE — RSENT Interrupt/DMA Enable Register

Access: This register can be read/written in 32-bit units.

Address: <RSENTn_base> + 0018_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SEIE	SMIE	SCIE	NRIE	CVIE	CLIE	FNIE	FEIE	FMIE	FCIE	FRIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.12 RSENTnIDE Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 11	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
10	SEIE	Slow Channel Encoding Error Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
9	SMIE	Slow Channel Message Lost Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
8	SCIE	Slow Channel CRC Error Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
7	NRIE	No Response Error Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
6	CVIE	Calibration Pulse Length Variation Error Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
5	CLIE	Calibration Pulse Length Error Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
4	FNIE	Fast Channel Nibble Count Error Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
3	FEIE	Fast Channel Nibble Encoding Error Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
2	FMIE	Fast Channel Message Lost Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
1	FCIE	Fast Channel CRC Error Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled

Table 15.12 RSENTnIDE Register Contents (2/2)

Bit Position	Bit Name	Function
0	FRIE	Fast Channel Receive Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled

RSENTnIDE.SEIE (Slow Channel Encoding Error Interrupt Enable)

This bit enables the generation of the slow channel encoding error interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.SMIE (Slow Channel Message Lost Interrupt Enable)

This bit enables the generation of the slow channel message lost interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.SCIE (Slow Channel CRC Error Interrupt Enable)

This bit enables the generation of the slow channel CRC error interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.NRIE (No Response Error Interrupt Enable)

This bit enables the generation of the no response error interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

The CPU should not set this bit when SPC mode is disabled (RSENTnCC.SPCE set to 0).

RSENTnIDE.CVIE (Calibration Pulse Length Variation Error Interrupt Enable)

This bit enables the generation of the calibration pulse length variation error interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.CLIE (Calibration Pulse Length Error Interrupt Enable)

This bit enables the generation of the calibration pulse length error interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.FNIE (Fast Channel Nibble Count Error Interrupt Enable)

This bit enables the generation of the fast channel nibble count error interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.FEIE (Fast Channel Nibble Encoding Error Interrupt Enable)

This bit enables the generation of the fast channel nibble encoding error interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.FMIE (Fast Channel Message Lost Interrupt Enable)

This bit enables the generation of the fast channel message lost interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.FCIE (Fast Channel CRC Error Interrupt Enable)

This bit enables the generation of the fast channel CRC error interrupt.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

RSENTnIDE.FRIE (Fast Channel Receive Interrupt Enable)

This bit enables the generation of the fast channel receive interrupt.

The fast channel receive interrupt can be also used to notify a DMA request.

The CPU cannot write to this bit if the RSENT module is in RESET mode (the RSENTnMST.OMS bits are 000_B).

15.3.7 RSENTnMDC — RSENT Mode Control Register

Access: This register can be read/written in 32-bit units.

Address: <RSENTn_base> + $001C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	OMC[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 15.13 RSENTnMDC Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
2 to 0	OMC[2:0]	Operation Mode Control 000_B: RESET 001_B: CONFIGURATION 011_B: OPERATION IDLE 101_B: OPERATION ACTIVE Other than above: Setting prohibited

RSENTnMDC.OMC (Operation Mode Control)

These bits are used to control the operation mode of the RSENT module.

- 000_B: RESET
  In RESET mode, the mode can only be changed to CONFIGURATION mode.
- 001_B: CONFIGURATION
  In CONFIGURATION mode, the mode can only be changed to RESET mode or OPERATION ACTIVE mode.
- 011_B: OPERATION IDLE
  In OPERATION IDLE mode, the mode can be changed to OPERATION ACTIVE mode, CONFIGURATION mode, or RESET mode.
- 101_B: OPERATION ACTIVE
  In OPERATION ACTIVE mode, the mode can be changed to OPERATION IDLE mode, CONFIGURATION mode, or RESET mode. However, it is recommended to process to the OPERATION IDLE mode first.
  For the recommended methods to select the operation mode, see **Section 15.4.3.1, Changing Operation Modes**.
- Other than above: Setting prohibited
  The CPU should not write any other value than listed above into this register.
  The CPU must follow the mode selection procedure shown in **Section 15.4.3.1, Changing Operation Modes**.

15.3.8 RSENTnSPCT — RSENT SPC Transmission Register

Access: This register can be read/written in 32-bit units.

Address: <RSENTn_base> + 0020_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TLL[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.14 RSENTnSPCT Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
6 to 0	TLL[6:0]	Length of the Trigger Low Phase in Ticks 0000000_B: 1 tick 0000001_B: 2 ticks 0000010_B: 3 ticks : 1111110_B: 127 ticks 1111111_B: 128 ticks

RSENTnSPCT.TLL (Trigger Low Length)

These bits define the length of the SPC trigger pulse.

When the CPU writes to these bits, an SPC trigger pulse with the configured length is sent immediately independently of the current status of the RSENT module.

For details about SPC communication, see **Section 15.4.4, SPC Function**.

The CPU can only write to these bits if the RSENT module is in OPERATION ACTIVE mode (the RSENTnMST.OMS bits are 101_B) and SPC communication is enabled (RSENTnCC.SPCE is 1).

It is important to note that two consecutive write access might not cause a no response error as the previous request might not have started yet.

After writing to this register, the CPU should wait for at least one SPC trigger tick before writing again to this register.

15.3.9 RSENTnMST — RSENT Mode Status Register

Access: This register is read-only in 32-bit units.

Address: <RSENTn_base> + 0024_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	OMS[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 15.15 RSENTnMST Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
2 to 0	OMS[2:0]	Operation Mode Control 000_B: RESET 001_B: CONFIGURATION 011_B: OPERATION IDLE 101_B: OPERATION ACTIVE Other than above: Reserved

RSENTnMST.OMS (Operation Mode Status)

These bits indicate the current operation mode.

These bits are read only.

These bits are updated after a mode change request is made in the RSENTnMDC.OMC register.

- 000_B: RESET mode
  When in RESET mode, all registers are set to their reset values and write access to all registers except the RSENTnMDC register is disabled.
  When in RESET mode, RSENT communication is disabled.
- 001_B: CONFIGURATION mode
  When in CONFIGURATION mode, write access to the timestamp registers (RSENTnTSPC and RSENTnTSC register), configuration registers (RSENTnCC and RSENTnBRP register), RSENTnIDE register, and mode control register (RSENTnMDC.OMC) is enabled.
  When in CONFIGURATION mode, RSENT communication is disabled.
  When entering CONFIGURATION mode, all status registers and receive buffer registers are set to their reset values.
- 011_B: OPERATION IDLE mode
  In OPERATION IDLE mode, no reception or SPC trigger transmission is possible.
  When entering OPERATION IDLE mode, frames in the receive buffer can be analyzed as in OPERATION ACTIVE mode, but no new frames are received.
- 101_B: OPERATION ACTIVE mode
  In OPERATION ACTIVE mode, reception and SPC trigger transmission are possible.
- Other than above: Reserved

15.3.10 RSENTnCS — RSENT Communication Status Register

Access: This register is read-only in 32-bit units.

Address: <RSENTn_base> + 0028_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SES	SMS	SCS	NRS	CVS	CLS	FNS	FES	FMS	FCS	FRS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 15.16 RSENTnCS Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 11	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
10	SES	Slow channel Encoding Error Interrupt Detection 0: Not detected 1: Detected
9	SMS	Slow Channel Message Lost Interrupt Detection 0: Not detected 1: Detected
8	SCS	Slow Channel CRC Error Interrupt Detection 0: Not detected 1: Detected
7	NRS	No Response Error Interrupt Detection 0: Not detected 1: Detected
6	CVS	Calibration Pulse Length Variation Error Interrupt Detection 0: Not detected 1: Detected
5	CLS	Calibration Pulse Length Error Interrupt Detection 0: Not detected 1: Detected
4	FNS	Fast Channel Nibble Count Error Interrupt Detection 0: Not detected 1: Detected
3	FES	Fast Channel Nibble Encoding Error Interrupt Detection 0: Not detected 1: Detected
2	FMS	Fast Channel Message Lost Interrupt Detection 0: Not detected 1: Detected

Table 15.16 RSENTnCS Register Contents (2/2)

Bit Position	Bit Name	Function
1	FCS	Fast Channel CRC Error Interrupt Detection 0: Not detected 1: Detected
0	FRS	Fast Channel Receive Interrupt Detection 0: Not detected 1: Detected

RSENTnCS.SES (Slow Channel Encoding Error Status)

This bit represents the slow channel encoding error status.

This bit is read only.

In the short serial message format (RSENTnCC.SMF = 01_B), this bit is set when the sequence on serial start bit (status & communication nibble bit #3) is different from "1000 0000 0000 0000_B" (a single 1 and 15 0s).

In the enhanced serial message format (RSENTnCC.SMF = 10_B), this bit is set unless bit 13 or bit 18 is received as 0 after a serial message start frame (sequence of status & communication nibble bit 3 is 0111 1110_B) is received.

When this bit is set in the short serial message format the received communication and status nibble is used to assemble a serial message.

When this bit is set in the enhanced serial message format the RSENT module checks the presence of a new start sequence at the same time and uses the received communication and status nibble to assemble a serial message.

This bit is cleared by writing 1 to RSENTnCSC.SEC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.SMS (Slow Channel Message Lost Status)

This bit represents the slow channel message lost status.

This bit is read only.

This bit is set when there is an attempt to update the slow channel message reception buffer, but the previous message has not been read yet.

This bit is cleared when writing 1 to RSENTnCSC.SEC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION). If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.SCS (Slow Channel CRC Error Status)

This bit represents the slow channel CRC error status.

This bit is read only.

This bit is set when a CRC error is detected on the slow channel and the slow channel CRC detection is enabled (RSENTnCC.SCCD is set to 0).

This bit is cleared when writing 1 to RSENTnCSC.SCC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION). If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.NRS (No Response Error Status)

This bit represents the no response error status.

This bit is read only.

This bit is set when

- the CPU writes to the RSENTnSPCT.TLL bits and
- SPC mode enabled (RSENTnCC.SPCE set to 1) and
- no complete response was received from the sensor for the previous SPC trigger.

This bit is cleared when writing 1 to RSENTnCSC.NRC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.CVS (Calibration Pulse Length Variation Error Status)

This bit represents the calibration pulse length variation error status.

This bit is read only.

When RSENTnCC.PPTC is 0, then this bit is set when two successive calibration pulses differ by more than 1.5625%.

When RSENTnCC.PPTC is 1, this bit is never set. In this mode (pause pulse with fixed message length), the CPU needs to check the variation of the ratio of calibration pulse to message length by reading the RSENTnCPL and RSENTnML registers.

This bit is cleared when writing 1 to RSENTnCSC.CVC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION). If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.CLS (Calibration Pulse Length Error Status)

This bit represents the calibration pulse length error status.

This bit is read only.

This bit is set when the measured calibration pulse length is less than 42 clock ticks or more than 70 clock ticks (deviation of 25% from specification length (56 clock ticks)).

This bit is cleared when writing 1 to RSENTnCSC.CLC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.FNS (Fast Channel Nibble Count Error Status)

This bit represents the fast channel nibble count error status.

This bit is read only.

This bit is set when there is an unexpected number of falling edges between two calibration pulses.

This bit is cleared when writing 1 to RSENTnCSC.FNC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION). If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.FES (Fast Channel Nibble Encoding Error Status)

This bit represents the fast channel nibble encoding error status.

This bit is read only.

This bit is set when on the fast channel a measured nibble period is less than 12 clock ticks or more than 27 clock ticks.

This bit is cleared when writing 1 to RSENTnCSC.FEC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION). If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.FMS (Fast Channel Message Lost Status)

This bit represents the fast channel message lost status.

This bit is read only.

This bit is set when the fast channel message receive buffer is updated, but the previous messages have not been read yet.

This bit is cleared when writing 1 to RSENTnCSC.FMC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION). If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.FCS (Fast Channel CRC Error Status)

This bit represents the fast channel CRC error status.

This bit is read only.

This bit is set when a CRC error is detected on the fast channel and the fast channel CRC detection is enabled (RSENTnCC.FCCD is set to 0).

This bit is cleared when writing 1 to RSENTnCSC.FCC.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION). If the set condition occurs simultaneously with the clear condition, the bit is set.

RSENTnCS.FRS (Fast Channel Receive Status)

This bit represents the fast channel receive status.

This bit is read only.

This bit is set when the fast channel message receive buffer was updated.

This bit is cleared when the CPU reads the RSENTnFRXD.FND bit.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION). If the set condition occurs simultaneously with the clear condition, the bit is set.

15.3.11 RSENTnCSC — RSENT Communication Status Clear Register

Access: This register can be read/written in 32-bit units.

Address: <RSENTn_base> + $002C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SEC	SMC	SCC	NRC	CVC	CLC	FNC	FEC	FMC	FCC	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 15.17 RSENTnCSC Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 11	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
10	SEC	Slow Channel Encoding Error Interrupt Clear 0: No effect 1: Clear
9	SMC	Slow Channel Message Lost Interrupt Clear 0: No effect 1: Clear
8	SCC	Slow Channel CRC Error Interrupt Clear 0: No effect 1: Clear
7	NRC	No Response Error Interrupt Clear 0: No effect 1: Clear
6	CVC	Calibration Pulse Length Variation Error Interrupt Clear 0: No effect 1: Clear
5	CLC	Calibration Pulse Length Error Interrupt Clear 0: No effect 1: Clear
4	FNC	Fast Channel Nibble Count Error Interrupt Clear 0: No effect 1: Clear
3	FEC	Fast Channel Nibble Encoding Error Interrupt Clear 0: No effect 1: Clear

Table 15.17 RSENTnCSC Register Contents (2/2)

Bit Position	Bit Name	Function
2	FMC	Fast Channel Message Lost Interrupt Clear 0: No effect 1: Clear
1	FCC	Fast Channel CRC Error Interrupt Clear 0: No effect 1: Clear
0	Reserved	When read, these bits are read as 0. When writing, write the value after reset.

RSENTnCSC.SEC (Slow Channel Encoding Error Clear)

Writing 1 clears RSENTnCS.SES to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.SMC (Slow Channel Message Lost Clear)

Writing 1 clears RSENTnCS.SMS to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.SCC (Slow Channel CRC Error Clear)

Writing 1 clears RSENTnCS.SCS to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.NRC (No Response Error Clear)

Writing 1 clears RSENTnCS.NRS to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.CVC (Calibration Pulse Length Variation Error Clear)

Writing 1 clears RSENTnCS.CVS to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.CLC (Calibration Pulse Length Error Clear)

Writing 1 clears RSENTnCS.CLS to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.FNC (Fast Channel Nibble Count Error Clear)

Writing 1 clears RSENTnCS.FNS to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.FEC (Fast Channel Nibble Encoding Error Clear)

Writing 1 clears RSENTnCS.FES to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.FMC (Fast Channel Message Lost Clear)

Writing 1 clears RSENTnCS.FMS to 0.

Writing 0 has no effect.

This bit is always read as 0.

RSENTnCSC.FCC (Fast Channel CRC Error Clear)

Writing 1 clears RSENTnCS.FCS to 0.

Writing 0 has no effect.

This bit is always read as 0.

15.3.12 RSENTnSRTS — RSENT Slow Channel Receive Timestamp Register

Access: This register is read-only in 32-bit units.

Address: <RSENTn_base> + 0030_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	STS[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 15.18 RSENTnSRTS Register Contents

Bit Position	Bit Name	Function
31 to 0	STS	Slow Channel Receive Timestamp

RSENTnSRTS.STS (Slow Channel Timestamp)

These bits are read only.

These bits are updated when the slow channel message reception buffer is updated with the timestamp counter value of the last frame provided to the slow channel message.

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

15.3.13 RSENTnSRXD — RSENT Slow Channel Receive Data Register

Access: This register is read-only in 32-bit units.

Address: <RSENTn_base> + 0034_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SND	—	SCRC[5:0]						—	—	—	SMGC	IDD[19:16]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IDD[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 15.19 RSENTnSRXD Register Contents

Bit Position	Bit Name	Function
31	SND	Slow Channel New Data 0: Slow channel frame data is not updated since last read. 1: Slow channel frame data is updated since last read.
30	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
29 to 24	SCRC[5:0]	Slow Channel CRC Data
23 to 21	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
20	SMGC	Slow Channel Configuration Bit Data
19 to 0	IDD[19:0]	Slow Channel Data / ID Information

RSENTnSRXD.SND (Slow Channel New Data)

This bit indicates that the slow channel message reception buffer is holding data that has not been read.

This bit is read only.

This bit is set when the slow channel message reception buffer is updated.

This bit is cleared automatically whenever it is read.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

RSENTnSRXD.SCRC (Slow Channel CRC)

These bits indicate the slow channel CRC data.

These bits are read only.

These bits are updated when the slow channel message reception buffer is updated.

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

RSENTnSRXD.SMGC (Slow Channel Configuration Bit)

This bit represents the slow channel configuration bit data.

This bit is read only.

This bit is updated when the slow channel message reception buffer is updated.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

RSENTnSRXD.IDD (ID/Data)

These bits are representing the slow channel data and ID information.

The alignment within this register depends on the message format. For details, see **Section 15.4.3.7,** Slow Channel Message Reception.

These bits are read only.

These bits are updated when the slow channel message reception buffer is updated.

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

15.3.14 RSENTnCPL — RSENT Calibration Pulse Length Register

Access: This register is read-only in 32-bit units.

Address: <RSENTn_base> + 0038_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CPLV [16]
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CPLV[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 15.20 RSENTnCPL Register Contents

Bit Position	Bit Name	Function
31 to 17	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
16 to 0	CPLV[16:0]	Calibration Pulse Length Value of Received Message

RSENTnCPL.CPLV (Calibration Pulse Length Value)

When suspension of pulses is enabled in fixed message length mode, these bits and the RSENTnML.MLV bits can be used by software to calculate the ratios of the respective message lengths in consecutive frames and of the calibration pulses and compare the variations to produce diagnostic messages.

These bits are read only.

Updating of the fast channel message reception buffer is storage of the value counted over the calibration pulse length (one tick time × the number of ticks between calibration pulses) based on the sample clock (fSAMPLE = 16 MHz).

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

15.3.15 RSENTnML — RSENT Message Length Register

Access: This register is read-only in 32-bit units.

Address: <RSENTn_base> + $003C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	MLV[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MLV[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 15.21 RSENTnML Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	When read, these bits are read as 0. When writing, write the value after reset.
20 to 0	MLV[20:0]	Message length of received message

RSENTnML.MLV (Message Length Value)

When suspension of pulses is enabled in fixed message length mode, these bits and the RSENTnCPL.CPLV bits can be used by software to calculate the ratios of the respective message lengths in consecutive frames and of the calibration pulses and compare the variations to produce diagnostic messages.

These bits are read only.

These bits are updated with the measured message length in sample clock ticks when the fast channel message receive buffer is updated.

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

15.3.16 RSENTnFRTS — RSENT Fast Channel Receive Timestamp Register

Access: This register is read-only in 32-bit units.

Address: <RSENTn_base> + 0040_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FTS[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 15.22 RSENTnFRTS Register Contents

Bit Position	Bit Name	Function
31 to 0	FTS[31:0]	Fast Channel Receive Timestamp

RSENTnFRTS.FTS (Fast Channel Timestamp)

These bits are read only.

These bits are updated when the fast channel message receive buffer is updated.

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

15.3.17 RSENTnFRXD — RSENT Fast Channel Receive Data Register

Access: This register is read-only in 32-bit units.

Address: <RSENTn_base> + 0044_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SNDM	FND	FCCN[1:0]		FCRC[3:0]				ND[23:16]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ND[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 15.23 RSENTnFRXD Register Contents

Bit Position	Bit Name	Function
31	SNDM	Slow Channel New Data Mirror 0: Slow channel frame data is not updated since last read. 1: Slow channel frame data is updated since last read.
30	FND	Fast Channel New Data 0: Fast channel frame data is not updated since last read. 1: Fast channel frame data is updated since last read.
29, 28	FCCN[1:0]	Fast Channel Status and Communication Nibble[1:0]
27 to 24	FCRC[3:0]	Fast Channel CRC Data
23 to 0	ND[23:0]	Fast Channel Nibble Data

RSENTnFRXD.SNDM (Slow Channel New Data Mirror)

This bit indicates that the slow channel message reception buffer is holding data that has not been read.

This bit is read only.

This bit is set when the slow channel message reception buffer is updated.

This bit is cleared automatically whenever the slow channel new data bit (RSENTnSRXD.SND) is read.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

RSENTnFRXD.FND (Fast Channel New Data)

This bit indicates that the fast channel message receive buffer is holding data that has not been read. This bit is read only.

This bit is set when the fast channel message receive buffer is updated.

This bit is cleared automatically whenever it is read.

This bit is cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

RSENTnFRXD.FCCN (Fast Channel Status and Communication Nibble)

These bits are representing the fast channel status and communication nibble bits [1:0].

These bits are read only.

These bits are updated when the fast channel message receive buffer is updated.

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

RSENTnFRXD.FCRC (Fast Channel CRC)

These bits are representing the fast channel CRC data.

These bits are read only.

These bits are updated when the fast channel message receive buffer is updated.

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

RSENTnFRXD.ND (Fast Channel Nibble Data)

These bits are representing the fast channel nibble data.

The alignment of the nibble data depends on nibble data count (RSENTnCC.NDN). For details, see **Section 15.4.3.5, Fast Channel Message Reception**.

These bits are read only.

These bits are updated when the fast channel message receive buffer is updated.

These bits are cleared when the RSENTnMST.OMS bits are changed to 001_B (CONFIGURATION).

15.4 Functions

15.4.1 Modes of Operation

RSENTn operates in any of the following modes.

- RESET mode
- CONFIGURATION mode
- OPERATION IDLE mode
- OPERATION ACTIVE mode

CPU should follow the mode change flow as shown **Section 15.4.3.1, Changing Operation Modes**.

The current operation mode status can be seen in the RSENTnMST.OMS bits.

Figure 15.2 shows the possible transitions between the channel modes:

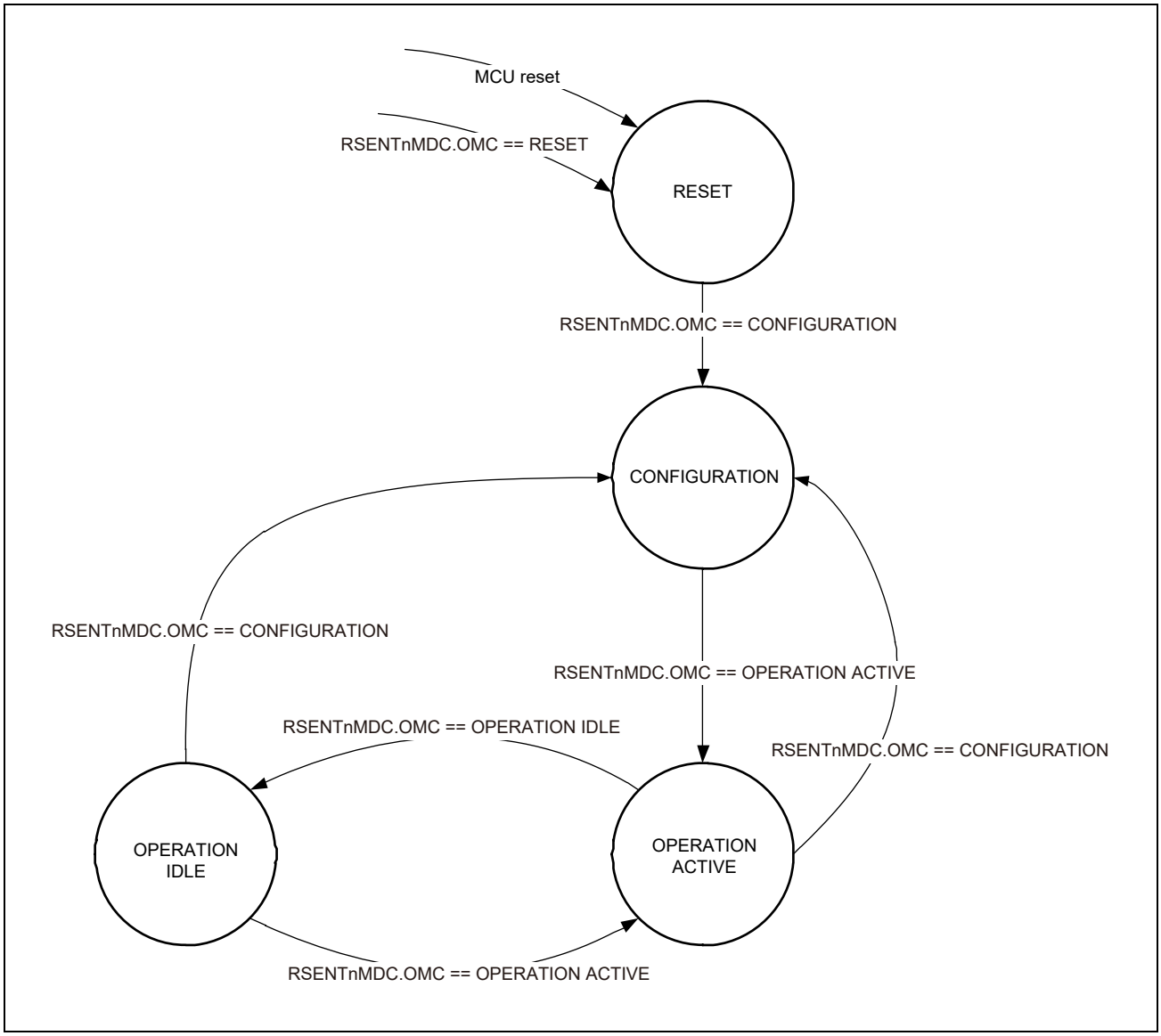


Figure 15.2 Transition between Operation Modes

15.4.1.1 RESET Mode

This mode is the initial mode that the RSENT module automatically enters after release from the reset state. This initializes the registers in the RSENT module.

The RESET mode is also entered after the RSENTnMDC.OMC bits have been set to 000_B. In this state, all, configuration, control (except RSENTnMDC.OMC bits), and status registers are set to the state after release from the reset state. Any on-going transmission or reception process is stopped immediately and the interface pins of the RSENT module are set to the value after their reset.

Read access to all registers is possible in RESET mode. Write access is limited to the RSENTnMDC register.

15.4.1.2 CONFIGURATION Mode

CONFIGURATION mode is entered after the RSENTnMDC.OMC bits are set to 001_B.

The interface pins of the RSENT module are set to their default values.

Regarding the output polarity setting of RSENTnSPCO pin and the timing that becomes effective, please refer to the explanation of "RSENTnCC.SOPC (SPC Output Polarity Control)" in **Section 15.3.4, RSENTnCC — RSENT Communication Configuration Register**.

In CONFIGURATION mode, all status registers (RSENTnCS) and the receive buffer registers

(RSENTnSRTS, RSENTnSRXD, RSENTnCPL, RSENTnML, RSENTnFRTS, and RSENTnFRXD) are set to the value after their reset.

Read access to all registers is possible in this state.

Write access is limited to both timestamp registers (RSENTnTSPC and RSENTnTSC) and configuration registers (RSENTnCC, RSENTnBRP, RSENTnIDE, RSENTnMDC, and RSENTnCSC).

15.4.1.3 OPERATION IDLE Mode

This mode is entered after the RSENTnMDC.OMC bits have been set to 011_B. In OPERATION IDLE mode, no reception and transmission are done.

When entering OPERATION IDLE mode, frames in the receive buffer can be analyzed as in OPERATION ACTIVE mode, but no new frames are received.

Read access to all registers is possible in OPERATION IDLE mode.

Write access is limited to RSENTnTSC, RSENTnIDE, RSENTnMDC, and RSENTnCSC.

15.4.1.4 OPERATION ACTIVE Mode

This mode is entered after the RSENTnMDC.OMC bits have been set to 101_B. In OPERATION ACTIVE mode, transmission and reception can take place.

Frame reception and status flagging starts after a valid calibration pulse (including the falling edge at the beginning) was detected.

Read access to all registers is possible in this state.

Write access is limited to RSENTnTSC, RSENTnIDE, RSENTnMDC, RSENTnSPCT, and RSENTnCSC.

15.4.1.5 Register Behavior in Operation Modes

Table 15.24 shows the register behavior when the RSENT module transitions to the indicated operation modes. The table also gives an overview about the access restriction in each operation mode.

Table 15.24 Register Behavior in Operation Modes

Register Name	Symbol	Reset	RESET mode		CONFIGURATION mode		OPERATION IDLE mode		OPERATION ACTIVE mode	
		Change	Change	R/W	Change	R/W	Change	R/W	Change	R/W
Timestamp prescaler configuration register	RSENTnTSPC	0000 0000$_H$	0000 0000$_H$	R	Unchanged	R/W	Unchanged	R	Unchanged	R
Timestamp counter register	RSENTnTSC	0000 0000$_H$	0000 0000$_H$	R	Unchanged	R/W	Unchanged	R/W *1	Unchanged	R/W *1
Communication configuration register	RSENTnCC	0000 0000$_H$	0000 0000$_H$	R	Unchanged	R/W	Unchanged	R	Unchanged	R
Baud rate prescaler register	RSENTnBRP	0000 0000$_H$	0000 0000$_H$	R	Unchanged	R/W	Unchanged	R	Unchanged	R
Interrupt/DMA enable register	RSENTnIDE	0000 0000$_H$	0000 0000$_H$	R	Unchanged	R/W	Unchanged	R/W	Unchanged	R/W
Mode control register	RSENTnMDC	0000 0000$_H$	0000 0000$_H$	R/W	Unchanged	R/W	Unchanged	R/W	Unchanged	R/W
SPC transmission register	RSENTnSPCT	0000 0000$_H$	0000 0000$_H$	R	Unchanged	R	Unchanged	R	Unchanged	R/W
Mode status register	RSENTnMST	0000 0000$_H$	0000 0000$_H$	R	0000 0001$_H$	R	0000 0003$_H$	R	0000 0005$_H$	R
Communication status register	RSENTnCS	0000 0000$_H$	0000 0000$_H$	R	0000 0000$_H$	R	Unchanged	R	Unchanged	R
Communication status clear register	RSENTnCSC	0000 0000$_H$	0000 0000$_H$	R	Unchanged	R/W	Unchanged	R/W	Unchanged	R/W
Slow channel receive timestamp register	RSENTnSRTS	0000 0000$_H$	0000 0000$_H$	R	0000 0000$_H$	R	Unchanged	R	Unchanged	R
Slow channel receive data register	RSENTnSRXD	0000 0000$_H$	0000 0000$_H$	R	0000 0000$_H$	R	Unchanged	R	Unchanged	R
Calibration pulse length register	RSENTnCPL	0000 0000$_H$	0000 0000$_H$	R	0000 0000$_H$	R	Unchanged	R	Unchanged	R
Message length register	RSENTnML	0000 0000$_H$	0000 0000$_H$	R	0000 0000$_H$	R	Unchanged	R	Unchanged	R
Fast channel receive timestamp register	RSENTnFRTS	0000 0000$_H$	0000 0000$_H$	R	0000 0000$_H$	R	Unchanged	R	Unchanged	R
Fast channel receive data register	RSENTnFRXD	0000 0000$_H$	0000 0000$_H$	R	0000 0000$_H$	R	Unchanged	R	Unchanged	R

Note 1. Means write restriction exists.

15.4.2 Clock Configuration

15.4.2.1 Timestamp

(1) Timestamp Clock Configuration

The RSENT module includes a timestamp counter.

The user should set the RSENTnTSPC.TTPV bits according to the communications frequency (that of clkc) such that the output of the prescaler, TPV, is a 1-µs clock tick.

The resolution of the timestamps can be decreased by setting the RSENTnTSPC.TTM bits according to the set tick length. The input frequency, which has already been divided by the prescaler, is further divided by the value of the RSENTnTSPC.TTM bits (a multiplier for the period).

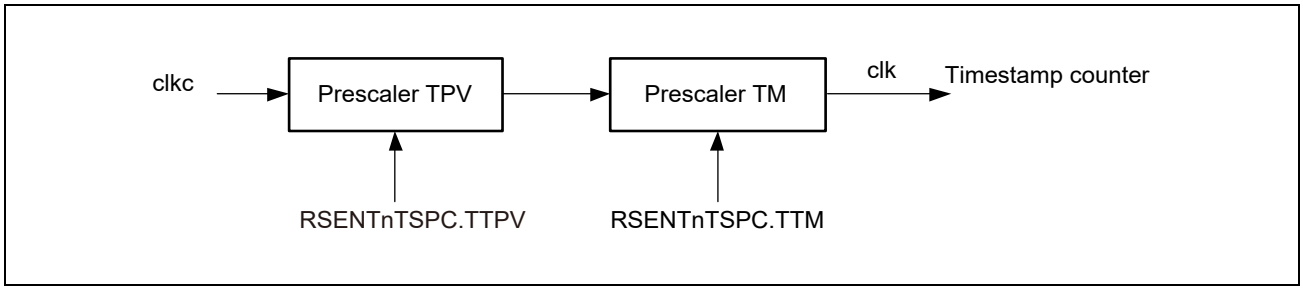


Figure 15.3 Timestamp Counter Clock Generation

(2) Timestamp Counter Operation

The timestamp counter value can be initialized to any value by writing to the RSENTnTSC.TS bits only when the RSENT module is in CONFIGURATION mode.

When timestamp counters are configured to operate in master mode (RSENTnTSPC.TMS = 0), the CPU can reset the timestamp counter by writing 0000 0000_H to the RSENTnTSC.TS bits when the RSENT module is in OPERATION_IDLE or OPERATION_ACTIVE mode.

When timestamp counters are configured to operate in slave mode (RSENTnTSPC.TMS = 1), the RSENTtimestamp counter is cleared when the CPU writes 0000 0000_H to the RSENTnTSC.TS bits of the channel set in the master when the RSENT module is in OPERATION_IDLE or OPERATION ACTIVE mode. Make the same settings for the timestamp counter prescalers of channels operating in master mode and slave mode. When timestamp counter synchronization occurs, the internal timestamp counter prescalers are also synchronized.

The current timestamp counter value can be read from the RSENTnTSC.TS bits.

When the RSENT module is in OPERATION ACTIVE mode, each received message is stored with its related timestamp. Timestamp values are taken for fast channel and slow channel data.

The timestamp value is captured when the calibration pulse is detected.

The timestamp value for the fast channel is stored in the RSENTnFRTS.FTS bits.

The timestamp value for the slow channel is stored in the RSENTnSRTS.STS bits. The timestamp value for the slow channel is identical to the timestamp value of the last fast channel message contributing to the slow channel message reception buffer.

In case timestamp counter synchronization is required, the following flow should be used.

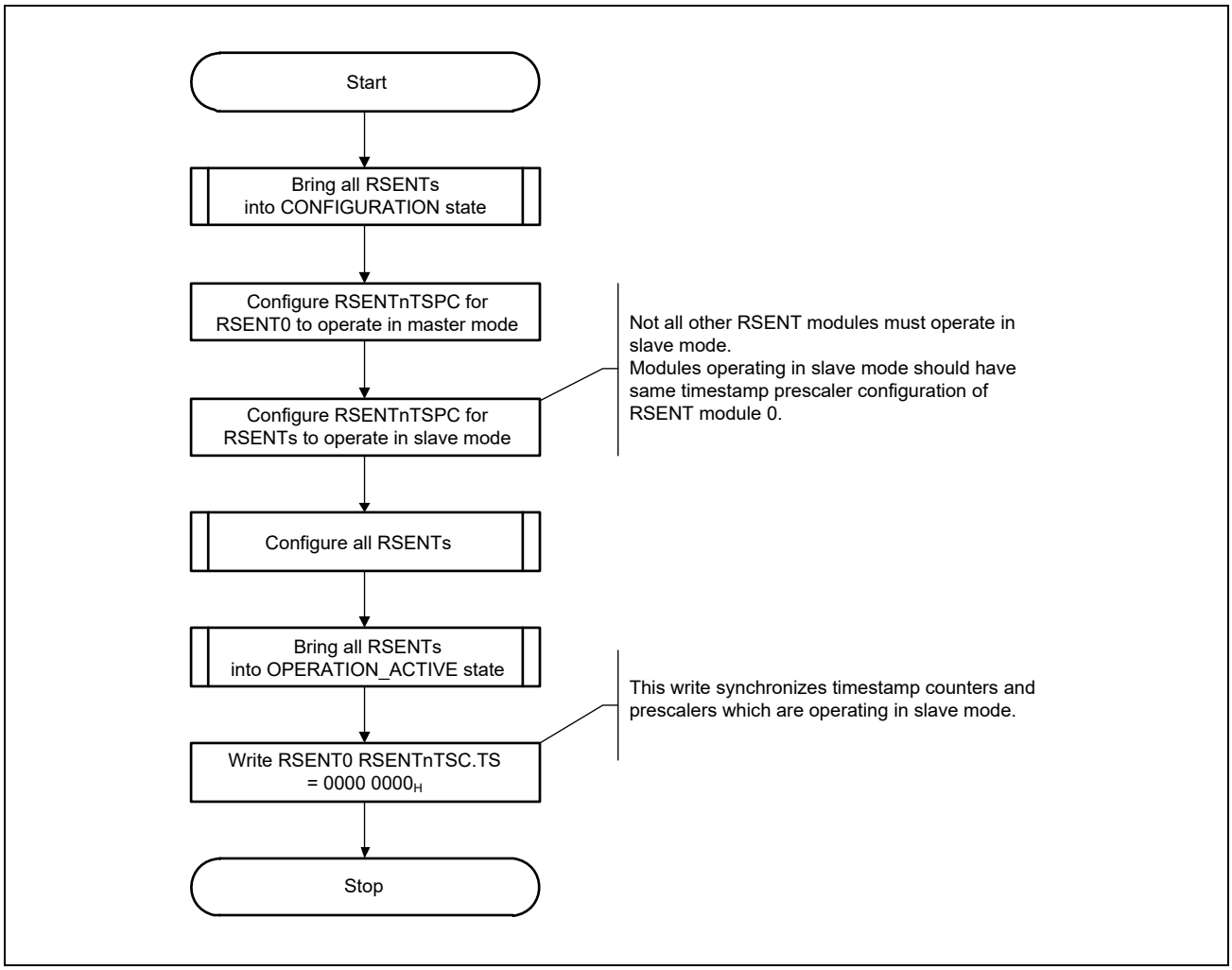


Figure 15.4 Timestamp Counter Synchronization

Synchronization of the timestamp can be done if the state of the master module has changed to OPERARION_ACTIVE or OPERARION_IDLE.

15.4.2.2 Communication Clock Configuration

(1) RX BRP Setting

Configure the communication baud rate by setting the RSENTnBRP.SCMV bit and the RSENTnBRP.SCDV bit according to the following formula.

Select the values of RSENTnBRP.SCMV and RSENTnBRP.SCDV so that frequency-dividing $f_{COMMUNICATION}$ (the clock clkc) sets f_{SAMPLE} to 16MHz.

$$f_{SAMPLE} = 16\,MHz = f_{COMMUNICATION} \times \frac{Sample\ clock\ multiplication\ value\ (BRP.SCMV + 1)}{Sample\ clock\ division\ value\ (BRP.SCDV + 1)}$$

Here, Sample clock multiplication value = 1 (BRP.SCMV = 00000B)

Sample clock division value = 5 (BRP.SCDV = 0000100B)

$f_{COMMUNICATION}$ = 80 MHz

f_{SAMPLE} = 80 × 1/5 = 16 MHz

(2) RX and SPC Tick Settings

The tick length to be used by the receive and SPC functions can be configured by setting the RSENTnBRP.TTI and RSENTnBRP.TTF bits. Tick lengths of 1.0 µs to 90.0 µs can be configured in 0.1 µs resolution.

RSENTnBRP.TTI holds the integer part of the tick length and the RSENTnBRP.TTF bits hold the fractional part of the tick length. The tick length is then calculated by:

$$T_{TICK} = T_{BRP.TTI} + T_{BRP.TTF}$$

Here, VRP.TTI = 0000000B, BRP.TTF = 0011B

T_{Tick} = 1 + 0.3 = 1.3 µs

15.4.3 RSENT Operation

When starting communication with RSENTn installed in a product of this series, be sure to configure the pints that are associated with the RSENTn interface in question before following the configuration steps explained in this section.

Make PORT settings for each of SENT reception system, SPC 1-wire system, and SPC 2-wire system as summarized below.

Table 15.25 PORT Settings when RSENT is used

		Port Setting Registers				
		PMC	PIPC	PM	PIBC	PBDC
SENT reception	RSENTnRX	1	0	1	0	0
SPC single-wire system	RSENTnRX/RSENTnSPCO	1	1	1	0	0
SPC two-wire system	RSENTnRX	1	0	1	0	0
	RSENTnSPCO	1	0	0	0	0

For details of the pin settings, see **Section 2, Pins**.

15.4.3.1 Changing Operation Modes

Once initialization has been completed in CONFIGURATION mode, operation can be enabled by entering OPERATION ACTIVE mode. This is done by setting the RSENTnMDC.OMC bits to OPERATION ACTIVE and waiting for the RSENTnMST.OMS to transition to OPERATION ACTIVE.

Once in OPERATION ACTIVE mode the RSENT module begins to receive messages or SPC communication can be started depending on the configuration.

Figure 15.5 shows the communication enabled flow assuming that the RSENT module is in RESET mode:

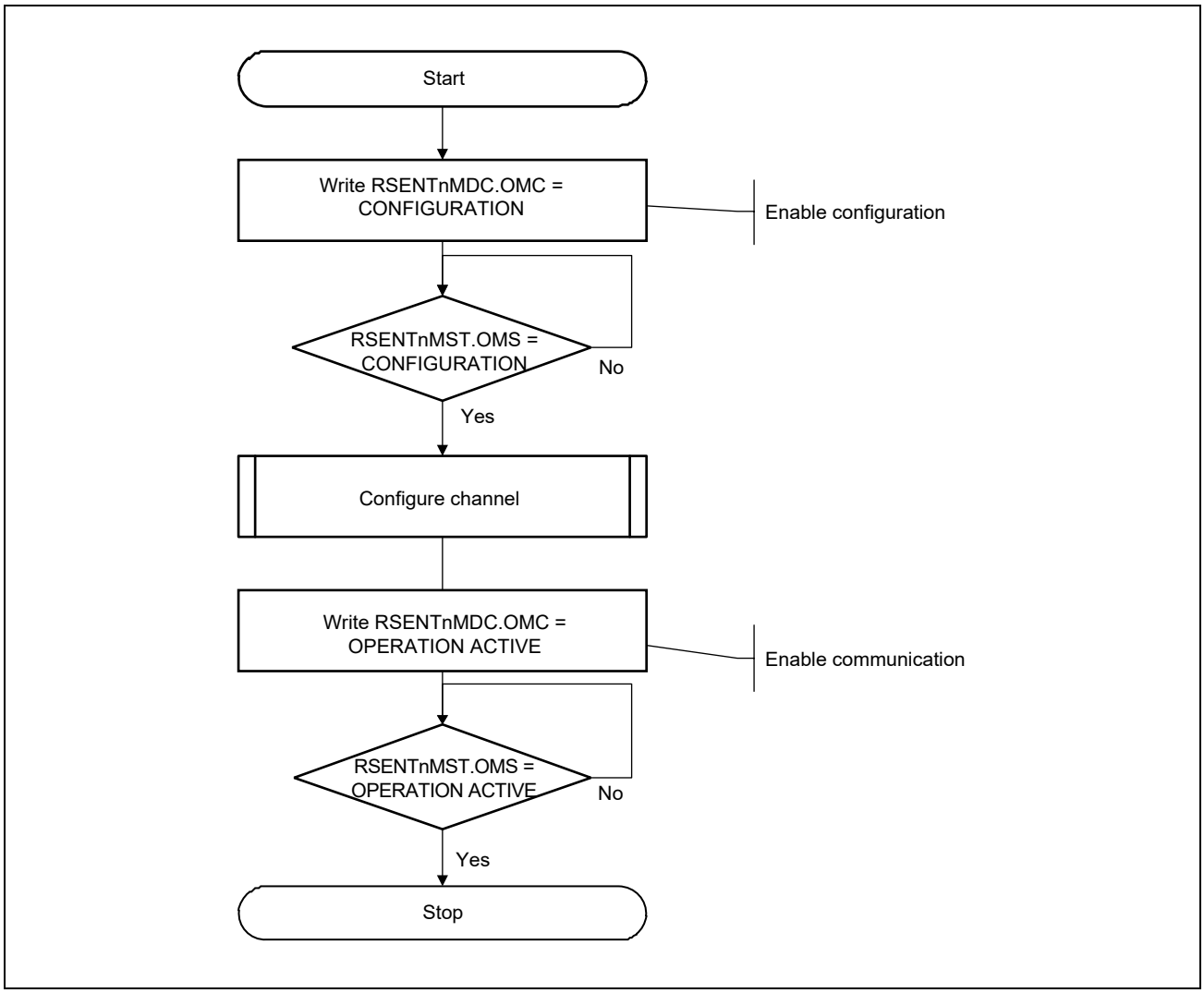


Figure 15.5 Communication Enable Flow

To leave OPERATION ACTIVE mode, communication should be disabled first by transition to OPERATION IDLE mode. This is done by setting the RSENTnMDC.OMC bits to OPERATION IDLE and waiting for the RSENTnMST.OMS bits to transition to OPERATION IDLE.

However, when the SPC mode is enabled (CC.SPCE = 1) and the SPC trigger transmission has not been requested after the previous SPC communication has been completed (e.g. successful reception for the previous SPC trigger transmission), the RSENT module can directly enter the CONFIGURATION mode.

The transition between OPERATION ACTIVE and OPERATION IDLE depends on the setting of the RSENTnCC.SPCE bit.

(1) RSENTnCC.SPCE = 0

In case a reception is currently ongoing, the mode change from OPERATION ACTIVE to OPERATION IDLE takes place when the receive buffer was updated or error was flagged (see **Section 15.4.3.5, Fast Channel Message Reception**).

In case no reception is ongoing, the mode change from OPERATION ACTIVE to OPERATION IDLE takes place immediately.

(2) RSENTnCC.SPCE = 1

In case a reception is ongoing, the mode change from OPERATION ACTIVE to OPERATION IDLE takes place when the falling edge of the end pulse is received.

In case a no response error is flagged, the mode change from OPERATION ACTIVE to OPERATION IDLE takes place at the same time as the error flagging.

The mode change from OPERATION ACTIVE to OPERATION IDLE takes place when the sequence of making a SPC trigger and receiving the response has been completed. This means when a response was already received, the transition takes place immediately. When the response is still pending, the mode change from OPERATION ACTIVE to OPERATION IDLE takes place when the falling edge of the end pulse is received.

CONFIGURATION mode can be entered by writing CONFIGURATION into the RSENTnMST.OMS bits and waiting for RSENTnMST.OMS to switch to the CONFIGURATION state.

Once CONFIGURATION mode is entered, the remaining status and message information stored in the RSENT module is lost since status and message information is cleared in CONFIGURATION mode.

Figure 15.6 shows the communication disable flow assuming that the RSENT module is in OPERATION ACTIVE mode.

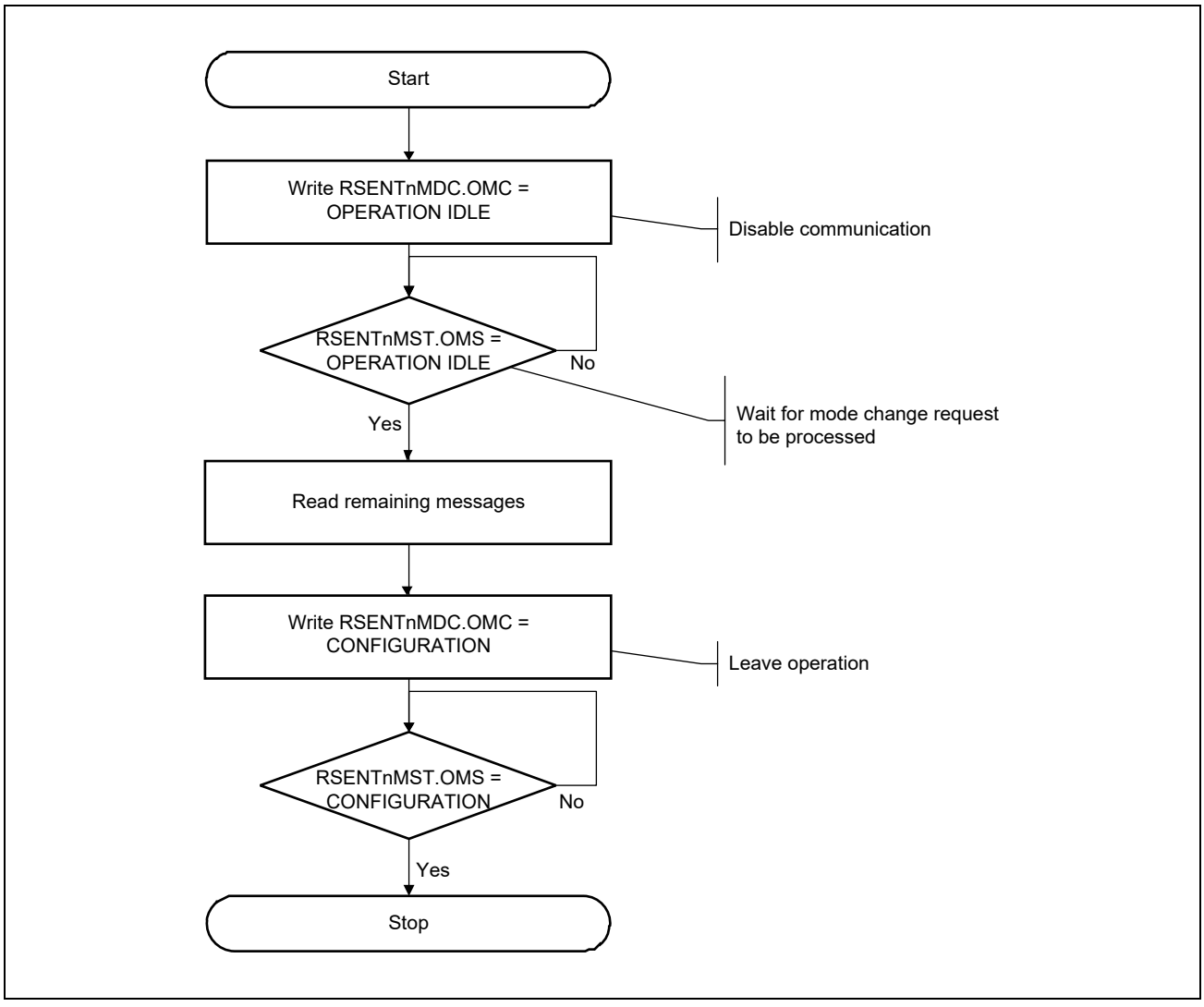


Figure 15.6 Communication Stop Flow

15.4.3.2 Message Reception

RSENT message reception is composed of the calibration pulse reception followed by the data nibble pulse reception.

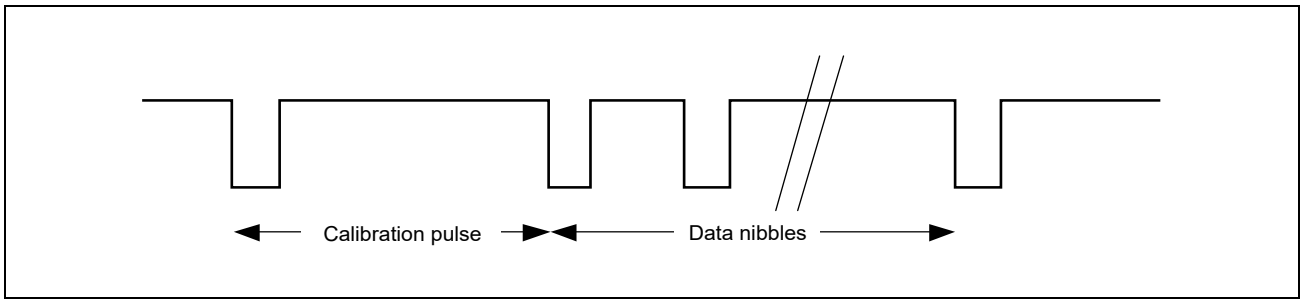


Figure 15.7 RSENT Received Message Structure

15.4.3.3 Calibration Pulse Reception

Within the calibration pulse reception phase the internally generated clock tick is adjusted to the transmit clock speed.

In addition, the calibration pulse is used to end the previous message and perform message diagnostics. The RSENT module supports automatic calibration pulse length diagnostics in variable message length modes (RSENTnCC.PPTC = 0). In case the calibration pulse ratio check fails, the calibration pulse length variation error flag (RSENTnCS.CVS) is set to 1.

15.4.3.4 Data Nibble Reception

The receive function of the RSENT module is a straightforward capture and compare function. The RSENT module receives sensor information encoded by the temporal distance of two consecutive falling edges on the data line. The temporal distance (in # of clock ticks) is captured and compared against a set of values to determine the actual nibble value. The data encoding is illustrated in **Table 15.26** below.

Table 15.26 Data Nibble Encoding

Nibble Period (# Clock Ticks)	Nibble Value (Binary)
12	0000_B
13	0001_B
14	0010_B
15	0011_B
16	0100_B
17	0101_B
18	0110_B
19	0111_B
20	1000_B
21	1001_B
22	1010_B
23	1011_B
24	1100_B
25	1101_B
26	1110_B
27	1111_B

The received data nibbles are composed into an RSENT message which is then stored in the fast channel message receive buffer.

Any other received nibble period during the reception of data nibbles will cause a fast channel nibble encoding error.

15.4.3.5 Fast Channel Message Reception

Messages received on the fast message channel are stored in a receive buffer.

A fast channel message receive buffer is composed of the calibration pulse length register (RSENTnCPL), the message length register (RSENTnML), the fast channel receive timestamp register (RSENTnFRTS), and the fast channel receive data register (RSENTnFRXD).

These registers are arranged on successive addresses for a transfer of the register content into memory using DMA.

The RSENT module is equipped with a double receive buffer structure that allows the storage of two complete RSENT messages including the related timestamp and message length information. Message decoding and assembling are done in a separate register stage.

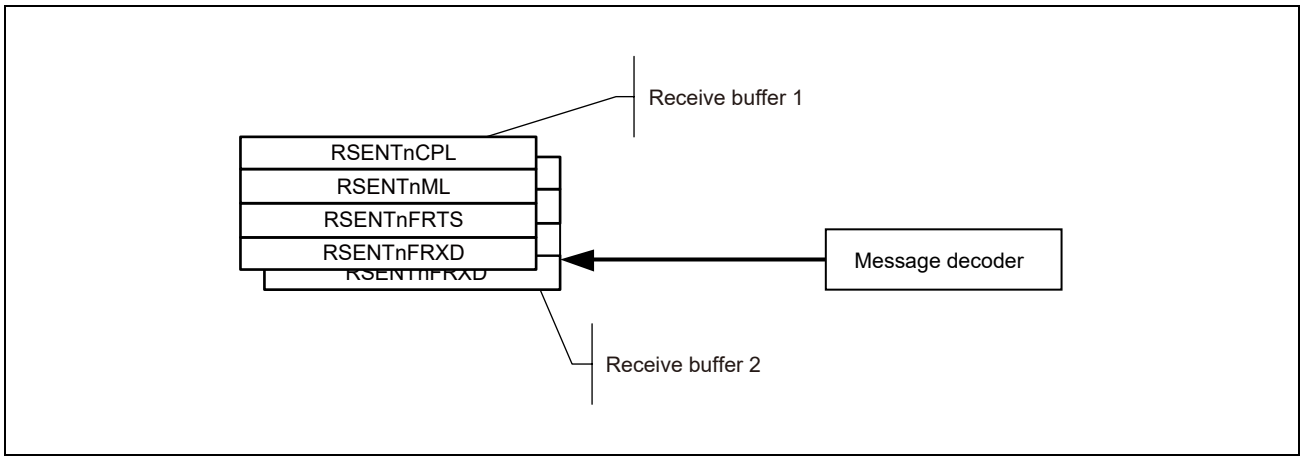


Figure 15.8 Fast Channel Message Receive Buffer

The first received message is placed into the message buffer that can be accessed by the CPU. This buffer (except the RSENTnFRXD.SNDM bit) is not updated any more until the RSENTnFRXD.FND bit was read.

When a new message is placed into a receive buffer, the RSENTnFRXD.FND bit is set. At the same time, the RSENTnCS.FRS bit is set and, if enabled, a receive interrupt request is generated.

When receive buffer 1 is holding an unprocessed message (the RSENTnFRXD.FND bit is 1), any further incoming message is placed in receive buffer 2. Receive buffer 2 is updated with any further incoming messages. In case an unprocessed message in receive buffer 2 is overwritten, the RSENTnCS.FMS bit is set to 1.

When the CPU reads the RSENTnFRXD.FND bit and there is valid data in the buffer 2, the data previously located in receive buffer 2 becomes available in the receive buffer becomes available in the receive buffer and is accessible by the CPU. If enabled, a new interrupt request for fast channel data is generated and RSENTnCS.FRS is set.

When the RSENTnFRXD.FND/ RSENTnCS.FRS bit is not set, the data in the receive buffer is not defined and the CPU should not access the receive buffer.

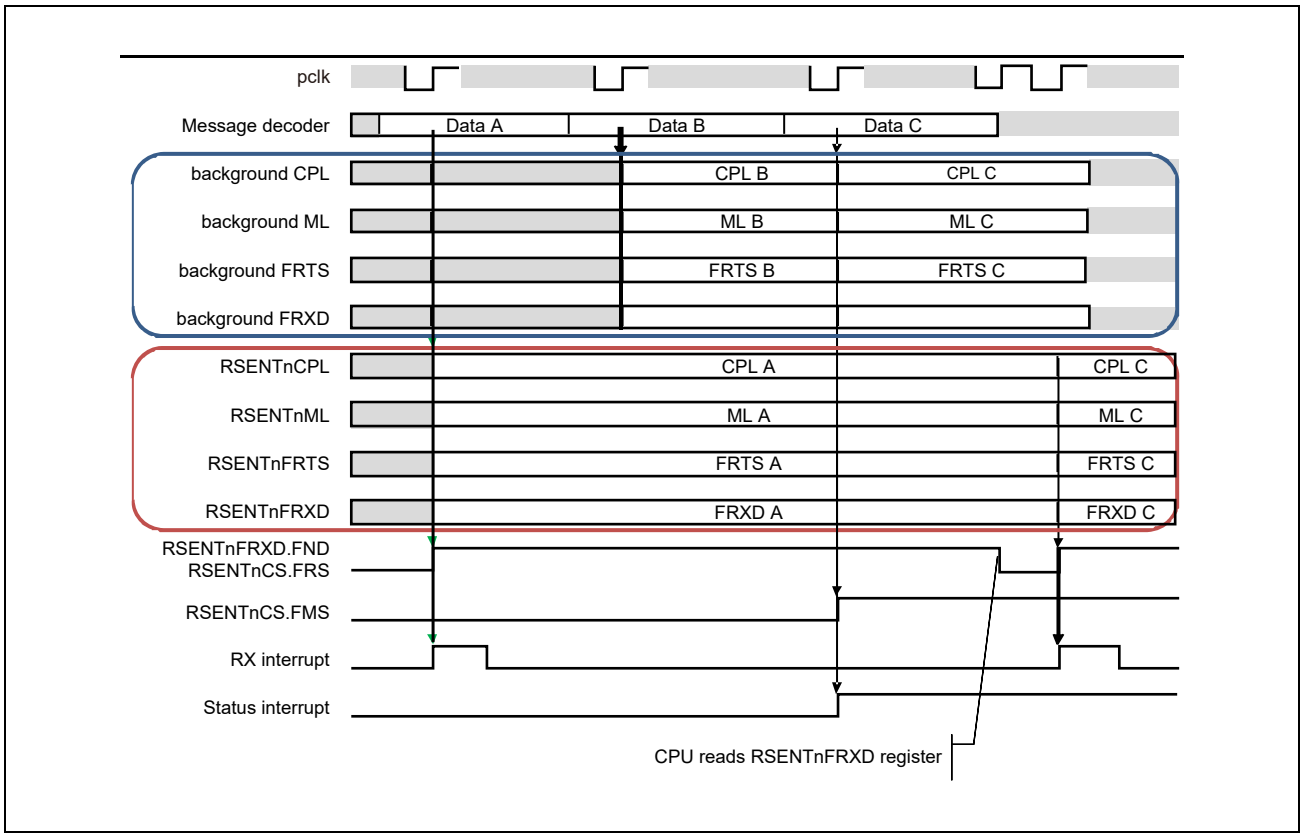


Figure 15.9 Fast Channel Message Receive Buffer Update Timing

The update timing of the receive buffer depends on the applied configuration as depicted in **Figure 15.10** to **Figure 15.13**.

The RSENTnFRTS register is updated with the current timestamp counter register value when the calibration pulse is detected.

The data alignment in the RSENTnFRXD register depends on the nibble data count (RSENTnCC.NDN).

Table 15.27 Data Nibble Alignment in RSENTnFRXD Register

RSENTnCC.NDN	23:20	19:16	15:12	11:8	7:4	3:0
000_B	Undefined	Undefined	Undefined	Undefined	Undefined	Nibble 1
001_B	Undefined	Undefined	Undefined	Undefined	Nibble 1	Nibble 2
010_B	Undefined	Undefined	Undefined	Nibble 1	Nibble 2	Nibble 3
011_B	Undefined	Undefined	Nibble 1	Nibble 2	Nibble 3	Nibble 4
100_B	Undefined	Nibble 1	Nibble 2	Nibble 3	Nibble 4	Nibble 5
101_B	Nibble 1	Nibble 2	Nibble 3	Nibble 4	Nibble 5	Nibble 6

(1) SAE operation with variable message length and preferred check method (RSENTnCC.SPCE = 0, RSENTnCC.PPTC = 0, RSENTnCC.FCM = 0)

In this operation mode, the RSENT module automatically performs the check for successive calibration pulse variation according to the preferred option in the J2716 2010 specification. In this mode, message diagnostics is done after the calibration pulse was received following a message.

If this check is passed, the message receive buffer is updated.

If this check is not passed, the message receive buffer is not updated and RSENTnCS.CVS is set to 1.

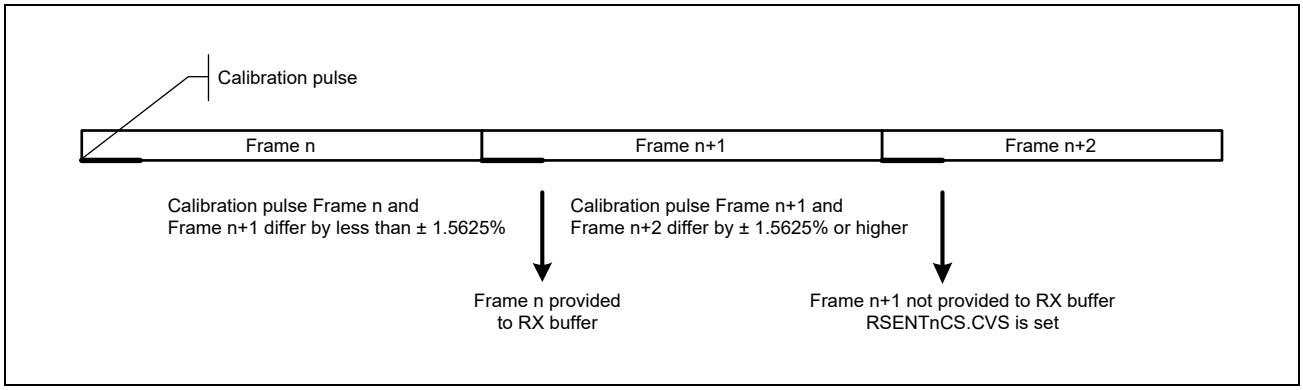


Figure 15.10 Buffer Update in Variable Message Length Mode and Preferred Check Method

(2) SAE operation with variable message length and optional check method (RSENTnCC.SPCE = 0, RSENTnCC.PPTC = 0, RSENTnCC.FCM = 1)

In this operation mode, the RSENT module automatically performs the check for successive calibration pulse variation according to the optional frame check method stipulated in the J2716 2010 specification. In this mode, message diagnostics is done after the calibration pulse was received following a message. In this mode, the calibration pulse in the current frame is compared with the calibration pulse in the preceding valid frame.

If this check is passed, the message receive buffer is updated.

If this check is not passed, the message receive buffer is not updated and RSENTnCS.CVS is set to 1.

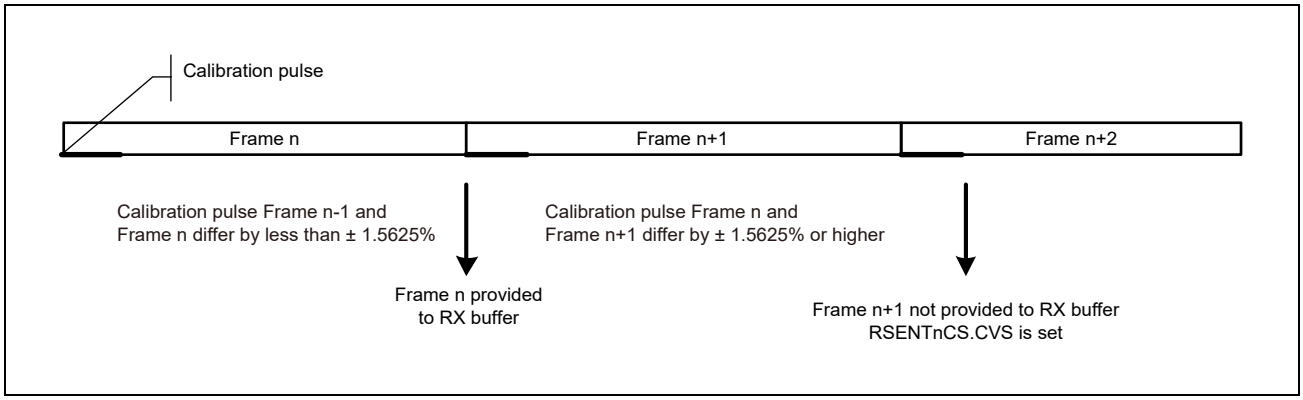


Figure 15.11 Buffer Update in Variable Message Length Mode and Optional Check Method

(3) SAE operation with fixed message length (RSENTnCC.SPCE = 0, RSENTnCC.PPTC = 1, RSENTnCC.PPC = 1)

In this mode, the RSENT module does not perform the check for calibration pulse and message length ratio according to the priority stipulated in the J2716 2010 specification. In this mode, the RSENT module provides the RSENTnCPL register with the calibration pulse length and the RSENTnML register with the message length information. The numbers that are provided are based on the sample.

The message buffer is updated at the beginning of the following calibration pulse irrespective of the values in the RSENTnCPL and RSENTnML registers. The CPU may accept or discard the message by calculating their ratio.

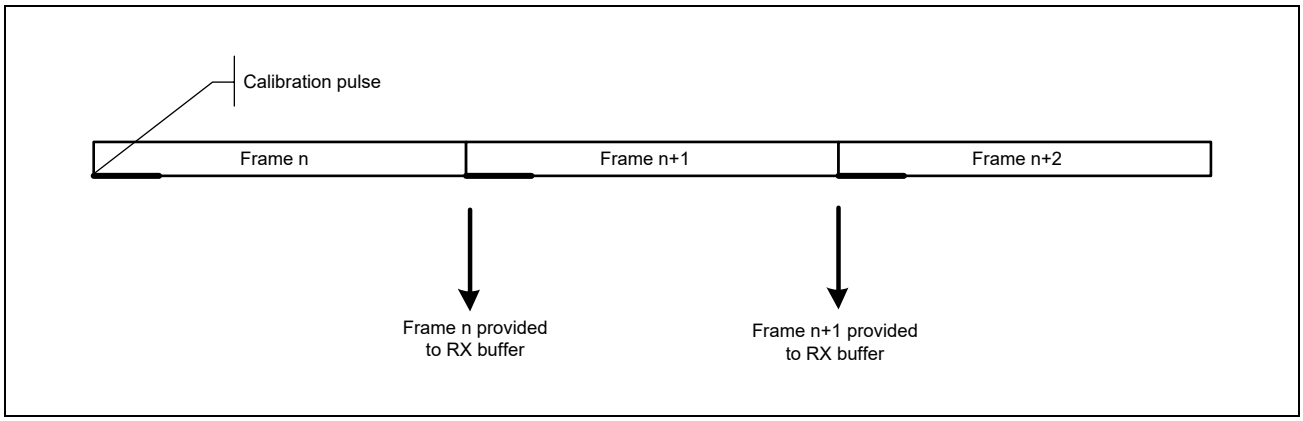


Figure 15.12 Buffer Update in Fixed Message Length Mode

The RSENTnCS.CVS (calibration pulse width variation error status) bit is never set in this mode.

(4) SPC operation (RSENTnCC.SPCE = 1)

In this operation mode, sensor data transmission is done following a SPC master trigger pulse. In SAE SENT communication, the calibration pulse or pause pulse interrupts the perceiving message. In SPC communication, the sensor sends only the data that follows a SPC trigger request. The end pulse sent by the sensor interrupts the message. The message buffer is updated when the end pulse is started.

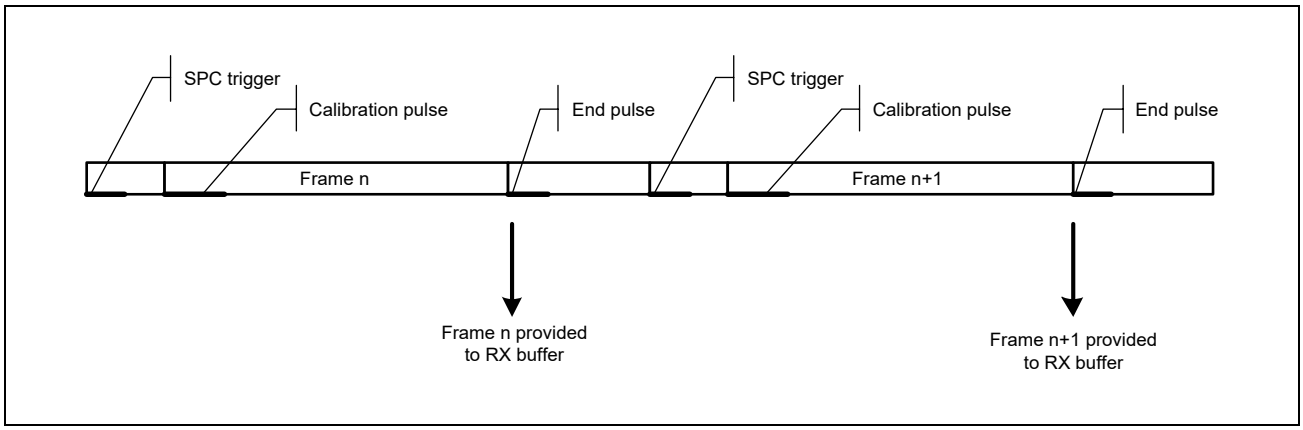


Figure 15.13 Buffer Update in SPC Mode

The RSENTnCS.CVS (calibration pulse width variation error status) bit is never set in this mode.

The RSENT module provides the RSENTnCPL register with the calibration pulse length and the RSENTnML register with the message length information. The numbers that are provided are based on the sample. The CPU may accept or discard the message by calculating their ratio.

In variable message length mode, the RSENT module cannot check the receive timing of the next calibration pulse because it depends on the timing of the next SPC trigger.

15.4.3.6 Fast Channel Reception Flow

Figure 15.14 shows the recommended reception flow for the fast channel message receive buffer. Using a polling or event driven method, the CPU can check for the presence or absence of new fast channel data by reading out only the setting of the RSENTnCS.FRS bit.

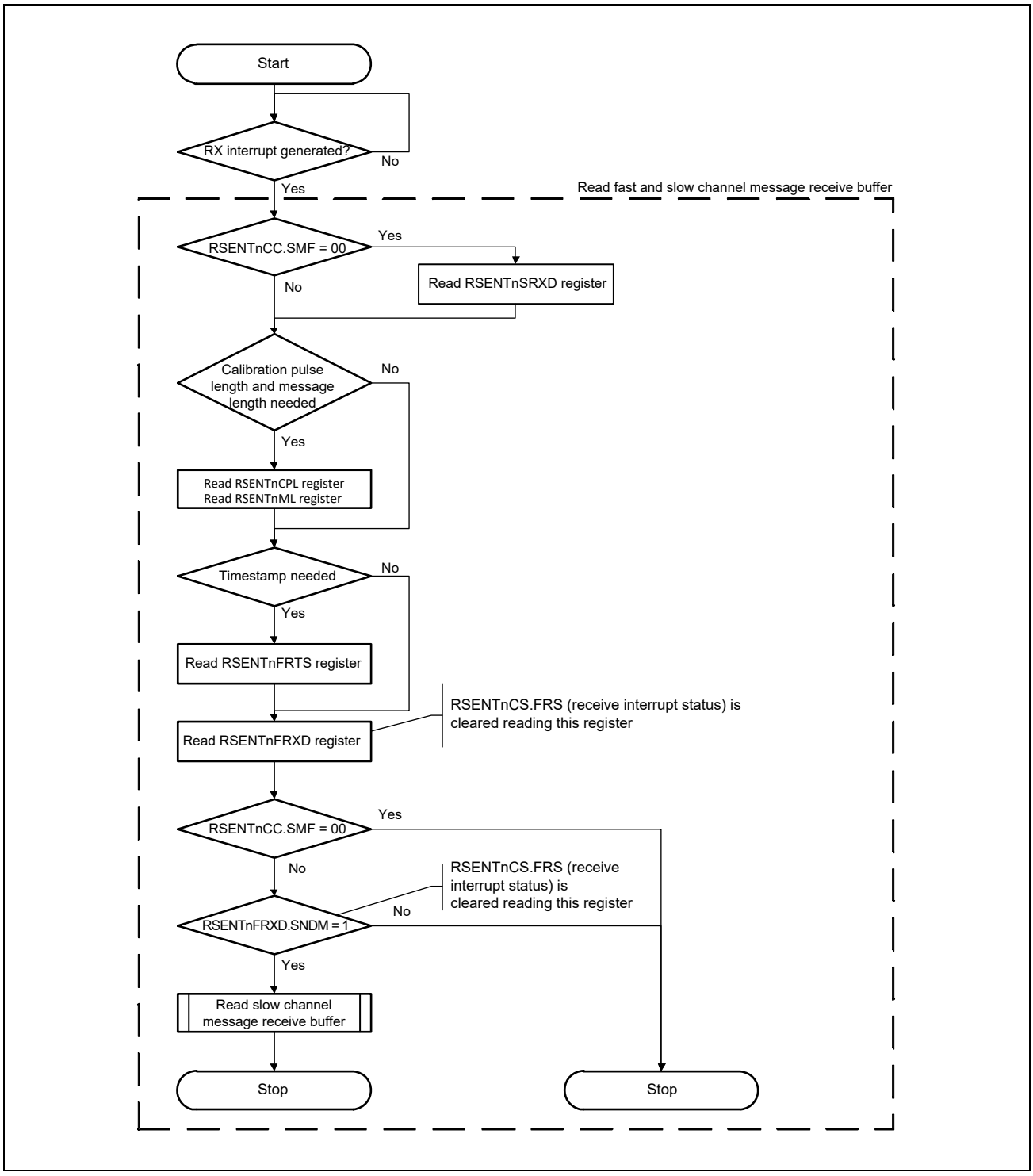


Figure 15.14 Fast Channel Reception Flow

In either case, the CPU needs to observe the receive buffer register read sequence shown in the flow. The RSENTnFRXD register need be the last register to be accessed.

The processing of the slow channel message receive buffer is described in **Section 15.4.3.8, Slow Channel Reception Flow**.

In SAE communication with pause pulse and fixed message length, the flow must be extended up to the message length by checking the ratio of the calibration pulse. This variation check must be performed by the CPU. If the variable check fails, the CPU must discard the received message.

15.4.3.7 Slow Channel Message Reception

The RSENT module supports the function to extract the slow message out of the fast channel messages using the status and communication nibble bits 3 and 2. To enable slow channel extraction, the CPU must set the RSENTnCC.SMF bits to the expected serial message format.

If serial message extraction is not selected (RSENTnCC.SMF = 00_B), the RSENTnSRXD register becomes part of the fast channel message receive buffer structure (including buffer 2) and the RSENTnSRTS register is ignored. The status and communication nibble is placed in the RSENTnSRXD.IDD bits. Furthermore neither slow channel new data nor slow channel message lost flags is generated.

For a slow channel serial message to be received, all the fast channel serial messages contributing to the slow channel serial message must be received successfully and the received slow channel serial message must comply with the selected serial message format.

A message lost on the fast channel will exert no influence on the reception on the slow channel. A slow channel message receive buffer is made up of a slow channel receive timestamp register (RSENTnSRTS) and a slow channel receive data register (RSENTnSRXD).

Unlike the fast channel message receive buffer, the slow channel message receive buffer does not have the double receive buffer structure but the single receive buffer structure. Message decoding and assembling is done in separate register stages.

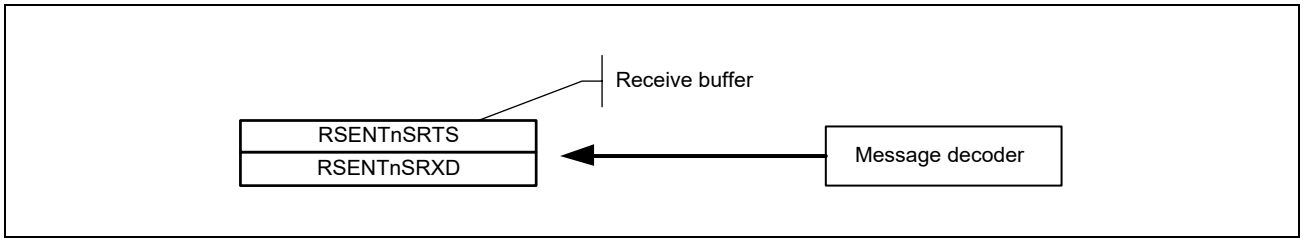


Figure 15.15 Slow Channel Message Reception Buffer

The slow channel message reception buffer is updated at the same time as the fast channel message receive buffer that holds the last status and communication nibble required for the slow channel message. The RSENTnSRXD.SND bit is set to 1 at the same time.

No Further updates to the buffer are carried out until the RSENTnSRXD.SND bit are read.

If the receive buffer is holding an unprocessed message (RSENTnSRXD.SND is 1), any further incoming message is lost (the slow channel message receive buffer is not updated). RSENTnCS.SMS is also set to 1.

When the CPU reads the RSENTnSRXD register, RSENTnSRXD.SND is automatically cleared.

The RSENTnSRTS register is updated with the current timestamp counter register value of the last frame contributing to the slow channel message.

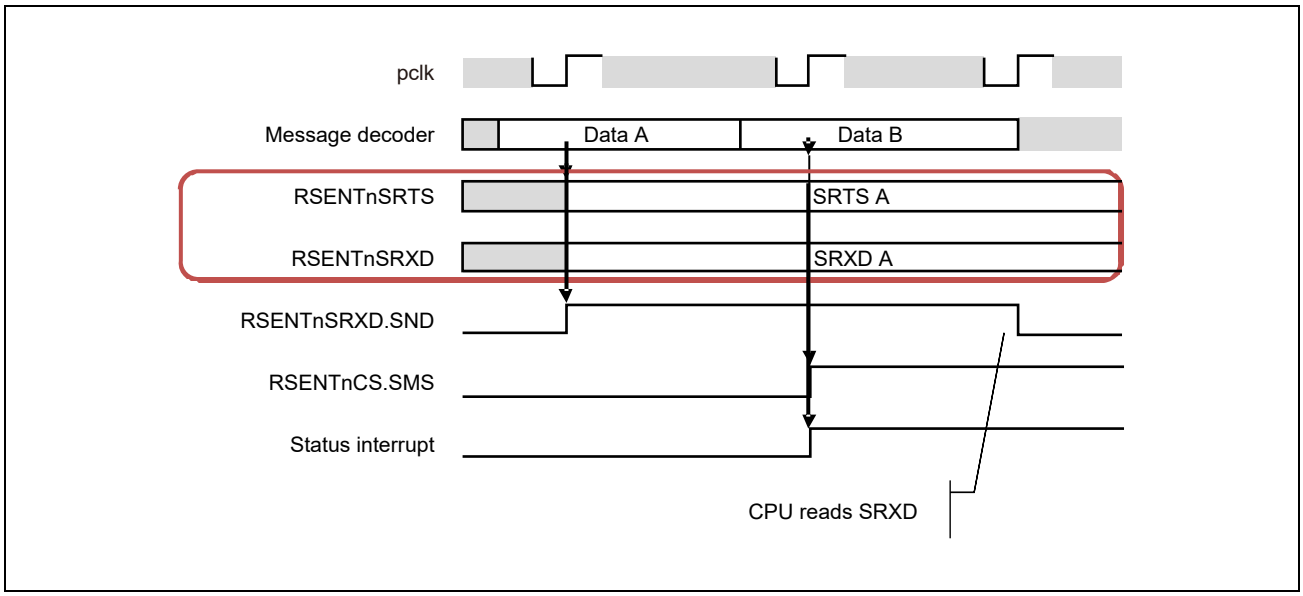


Figure 15.16 Slow Channel Message Receive Buffer Update Timing

The arrangement of the data in the SRDX register depends on the slow channel message format (RSENTnCC.SMF) and the received configuration bit.

Table 15.28 RSENTnSRXD Register Data Arrangement

RSENTn CC.SMF	RSENTnSRX D. SMGC	RSENTnSRXD. IDD[19:16]	RSENTnSRXD. IDD[15:12]	RSENTnSRXD. IDD[11:8]	RSENTnSRXD. IDD[7:4]	RSENTnSRXD. IDD[3:0]
00_B	Undefined	Undefined	Undefined	Undefined	Undefined	C & S nibble
01_B	Undefined	Undefined	Undefined	Message ID[3:0]	DATA[7:4]	DATA[3:0]
10_B	0	Message ID[7:4]	Message ID[3:0]	DATA[11:8]	DATA[7:4]	DATA[3:0]
10_B	1	Message ID[3:0]	DATA[15:12]	DATA[11:8]	DATA[7:4]	DATA[3:0]

15.4.3.8 Slow Channel Reception Flow

In **Figure 15.17**, the recommended reception flow for the slow channel message receive buffer is shown. When the slow channel receive data is required, this process should be executed as part of the fast channel reception flow.

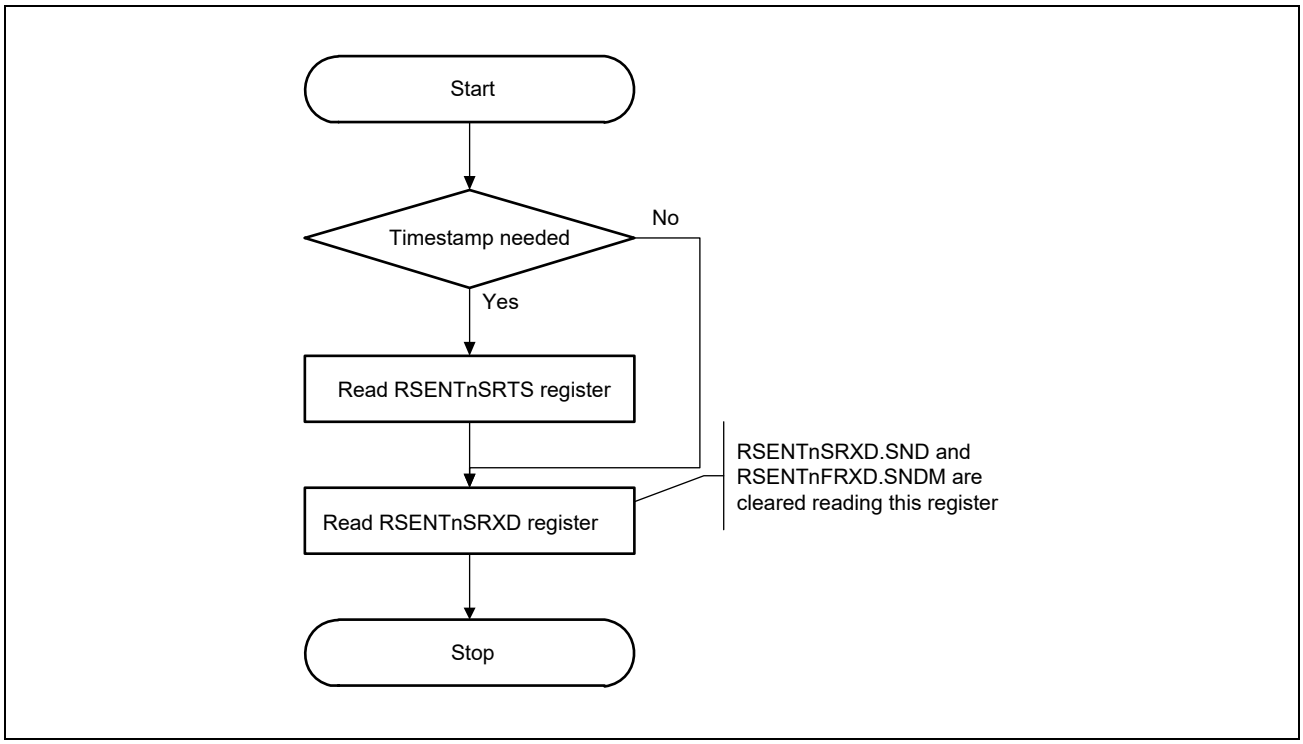


Figure 15.17 Slow Channel Reception Flow

In any case, the CPU should keep the order in reading the slow channel message receive buffer registers as shown in the flow. The RSENTnSRXD.SND bit should be accessed as last.

15.4.3.9 DMA Flow

In case of DMA usage, the start address for the DMA usage and the number of transfers define which part of the receive buffer will be transferred. The RSENTnFRXD register should be the last register to be accessed using a 32 bit access method.

In case of SAE communication with pause pulse and fixed message length, the flow must be extended by checking of the ratio of the calibration pulse to message length. This variation check must be performed by the CPU. In case the variation check fails, the CPU must discard the received message.

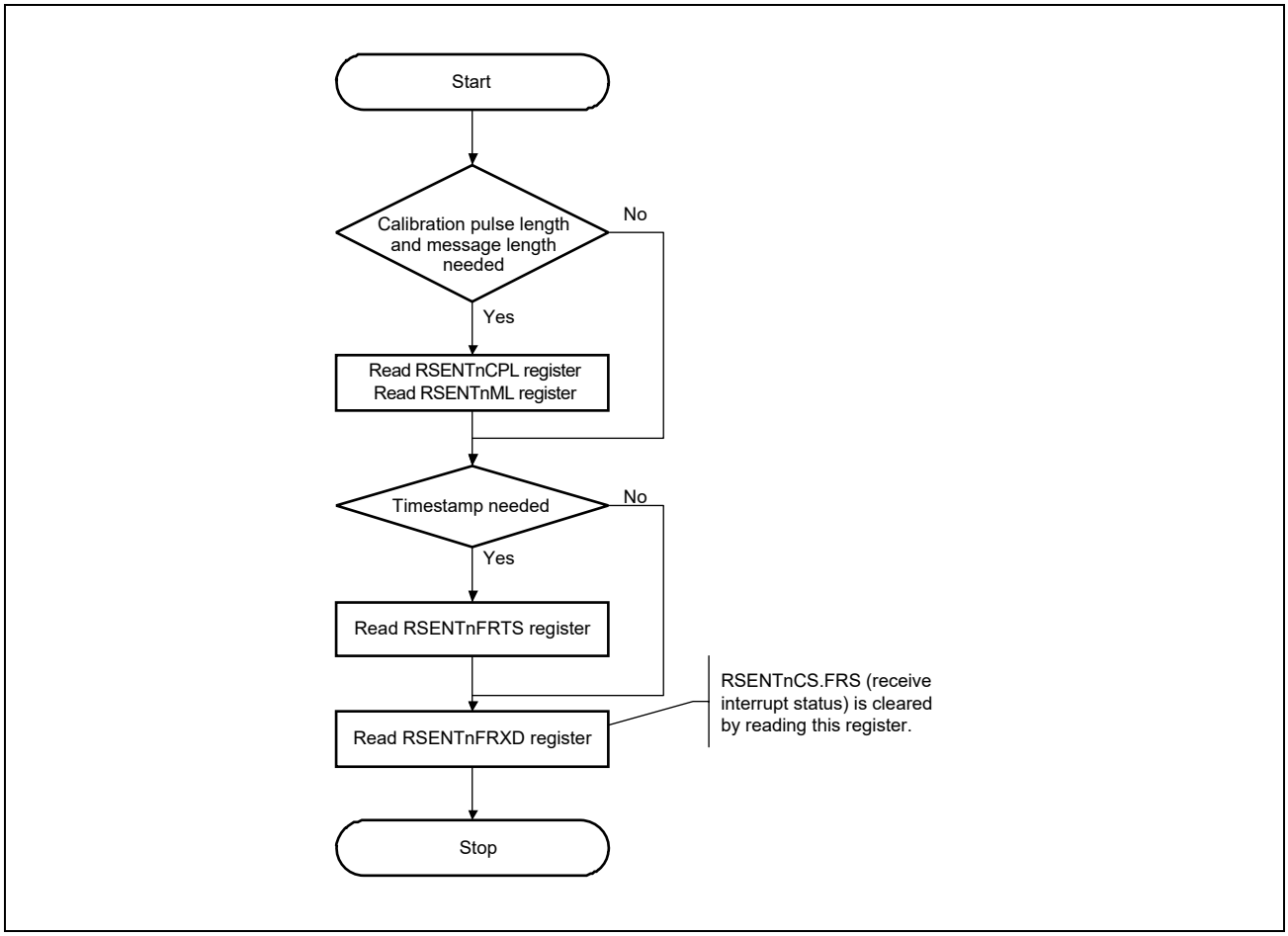


Figure 15.18 DMA Reception Flow

In the software processing, when the transferred data set, the CPU should check the status of the transferred RSENTnFRXD.SNDM bit. If this bit is set to 1, then the user needs to read the slow channel message receive buffer if needed.

15.4.3.10 Error Flagging

The message lost error flag (RSENTnCS.SMS or RSENTnCS.FMS) is set when a new message's diagnostics pass before the previous message is read.

The no response error flag (RSENTnCS.NRS) is set when the CPU has written to RSENTnSPTC.TLL before or during response reception.

The timing at which the fast channel reception error flags (RSENTnCS.CVS, RSENTnCS.CLS,RSENTnCS.FNS, RSENTnCS.FES, and RSENTnCS.FCS) and the slow channel reception error flags (RSENTnCS.SCS, and RSENTnCS.SES) are updated varies with the setting of bits in the communications configuration register (RSENTnCC.SPCE, RSENTnCC.FCM, RSENTnCC.PPC, and RSENTnCC.PPTC).

Table 15.29 and **Table 15.30** list the timings with which the error flags corresponding to each setting are updated.

When a fast channel nibble encoding error or calibration pulse length error is detected, message reception is aborted immediately. No further error flagging for the given message is done. Message decoding starts again after the detection of a valid calibration pulse.

When OPERATION IDLE mode is entered in response to the setting of RSENTnMDC.OMC, error flags of errors for the calibration or the fast channel reception which have been detected in a message being received are not set. A received message is also discarded.

If a fast channel nibble encoding error or calibration pulse length error is detected, entry to OPERATION IDLE mode is immediate.

If a fast channel nibble count error, fast channel CRC error, or calibration pulse length variation error is detected, entry to OPERATION IDLE mode proceeds at the end of the next status and communications nibble.

The fast channel nibble count error flag (RSENTnCS.FNS) is only set when a valid calibration pulse is detected and the subsequent data nibble has a valid length (≥ 12 ticks and ≤ 27 ticks) or no data nibble has been received between two valid calibration pulses.

The fast channel nibble encoding error flag (RSENTnCS.FES) is only set when an encoding error has occurred in a status and communications nibble, CRC nibble, or data nibble.

When the SPC function is enabled (i.e. RSENTnCC.SPCE = 1), the calibration pulse length error flag (RSENTnCS.CLS) is set when the width of a pulse at the expected calibration pulse position is not a valid pulse width for a calibration pulse.

When the SPC function is disabled (i.e. RSENTnCC.SPCE = 0), the calibration pulse length error flag (RSENTnCS.CLS) is set when the width of a pulse at a next expected calibration pulse position after the reception of a valid calibration pulse is not a valid pulse width for a calibration pulse.

An additional error flag being set while seeking a valid calibration pulse does not affect the reception of subsequent frames.

Table 15.29 Timing at which Error Flag is Set while SPC Mode is Disabled

RSENTnCC.SPCE	0							
RSENTnCC.FCM	0				1			
RSENTnCC.PPC	0		1		0		1	
RSENTnCC.PPTC	0	1	0	1	0	1	0	1
RSENTnCS.FCS	EC	×	EC	IM	IM	×	IM	IM
RSENTnCS.FES	EC	×	EC	IM	IM	×	IM	IM
RSENTnCS.FNS	EC	×	EC	—	—	×	—	—
RSENTnCS.SCS	IM	×	IM	IM	IM	×	IM	IM
RSENTnCS.SES	IM	×	IM	IM	IM	×	IM	IM
RSENTnCS.CLS	IM	×	IM	IM	IM	×	IM	IM
RSENTnCS.CVS	EC	×	EC	—	EC	×	EC	—

EC: A calibration pulse is received

IM: Detected

—: Not detected

×: Setting prohibited

Table 15.30 Timing at which Error Flag is Set while SPC Mode is Enabled

RSENTnCC.SPCE	1							
RSENTnCC.FCM	0				1			
RSENTnCC.PPC	0		1		0		1	
RSENTnCC.PPTC	0	1	0	1	0	1	0	1
RSENTnCS.FCS	IM	×	IM	IM	IM	×	IM	IM
RSENTnCS.FES	IM	×	IM	IM	IM	×	IM	IM
RSENTnCS.FNS	—	×	—	—	—	×	—	—
RSENTnCS.SCS	IM	×	IM	IM	IM	×	IM	IM
RSENTnCS.SES	IM	×	IM	IM	IM	×	IM	IM
RSENTnCS.CLS	IM	×	IM	IM	IM	×	IM	IM
RSENTnCS.CVS	—	×	—	—	—	×	—	—

IM: Detected

—: Not detected

×: Setting prohibited

NOTE

When the sensor stops communications, the reception buffer and the state will not be updated after reception of the last message. Confirm the timeout by software.

15.4.4 SPC Function

RSENTn in this product series supports the SPC extension to the SAE J2716 specification.

The RSENTnCC.SPCE bit is used to enable or disable the SPC extension.

When this module is to be used with SPC enabled, the RSENTnSPCO pin must be externally pulled up.

The polarity of RSENTnSPCO can be set by RSENTnCC.SOPC.

When using the SPC extension in the single-wire SENT system, however, be sure to set RSENTnCC.SOPC to 1 (configuring the SPC pulse as active low).

The use of the RSENTnSPCO output allows an RSENTn module to pull down the SPCO signal line via an external transistor.

The signal line is held at the low level for the tick time specified by the RSENTnSPCT.TLL bits.

The tick time is implemented by the RSENTnBRP.TTI and RSENTnBRP.TTF bits whose values form the time equivalent to the transmission tick time. For details, see **Section 15.4.2.2(2), RX and SPC Tick Settings**.

Figure 15.19 shows a sample circuit diagram for the external transistor.

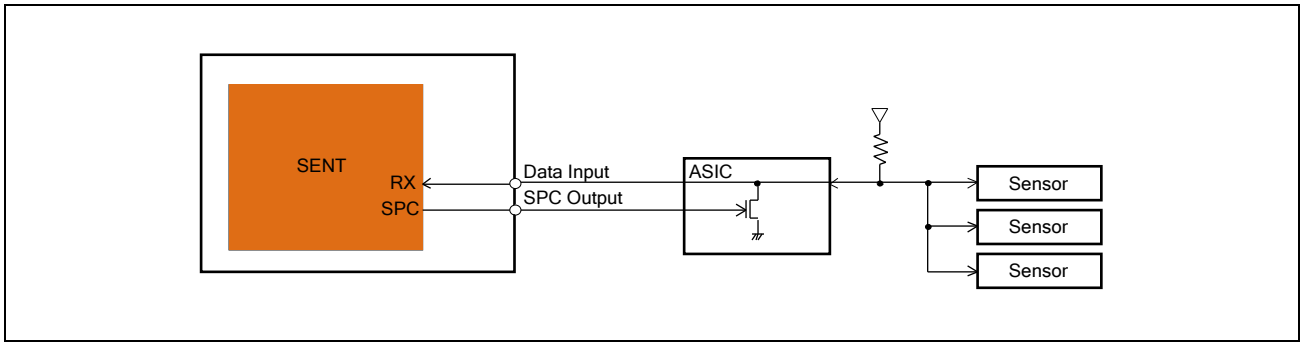


Figure 15.19 Sample Circuit Diagram for the External Transistor

In the SPC extended single-wire system, the RSENTnSPCO pin can be used to trigger data transmission from the sensor. Furthermore, the data can be sent to the sensor by changing the trigger pulse length.

In the SPC extended 2-wire system, the RSENTnSPCO pin can be used to request the dedicated sensor to transmit data.

Once SPC initialization is complete, transmission can be triggered by writing the trigger pulse width to the RSENTnSPCT.TLL register. When the transmission is triggered, a trigger pulse of the specified length is transmitted. Subsequently, a new trigger pulse can be transmitted after the reception of a frame is anticipated and a frame is received.

Writing to the RSENTnSPCT.TLL bits requests the transmission of an SPC trigger.

After writing to RSENTnSPCT.TLL, read the value of RSENTnCS.NRS to confirm whether the response to the previous SPC trigger request was completed.

If RSENTnCS.NRS is set to 1, no SPC trigger is transmitted and any reception that may be in progress at this time is aborted.

If this is the case, it is necessary to clear RSENTnCS.NRS by writing 1'b1 to RSENTnCSC.NRC and to write to RSENTnSPCT.TLL again to request the transmission of an SPC trigger.

For an error case in which RSENTnCS.NRS is not set to 1, it is necessary to count the number of receive timeouts with SW (Soft Ware). If a reception occurs before the timeout counter times out, it is necessary to process the received slow and fast channel data according to the fast channel receive flow (**Figure 15.14**) and slow channel receive flow (**Figure 15.17**). When the timeout counter times out, it indicates that the specified sensor did not transmit a valid response.

In such a case, it is necessary to check the RSENTnCS register to identify the reason for the absence of a successful reception. After confirming that RSENTnCS.NRS is not set, set up a new request. The purpose of the timeout function is to define a timeout window for receiving responses under SW.

Figure 15.20, Transmission Flow shows the flow of transmission using a timeout function implemented by software.

The timeout function is optional and can be omitted if not required.

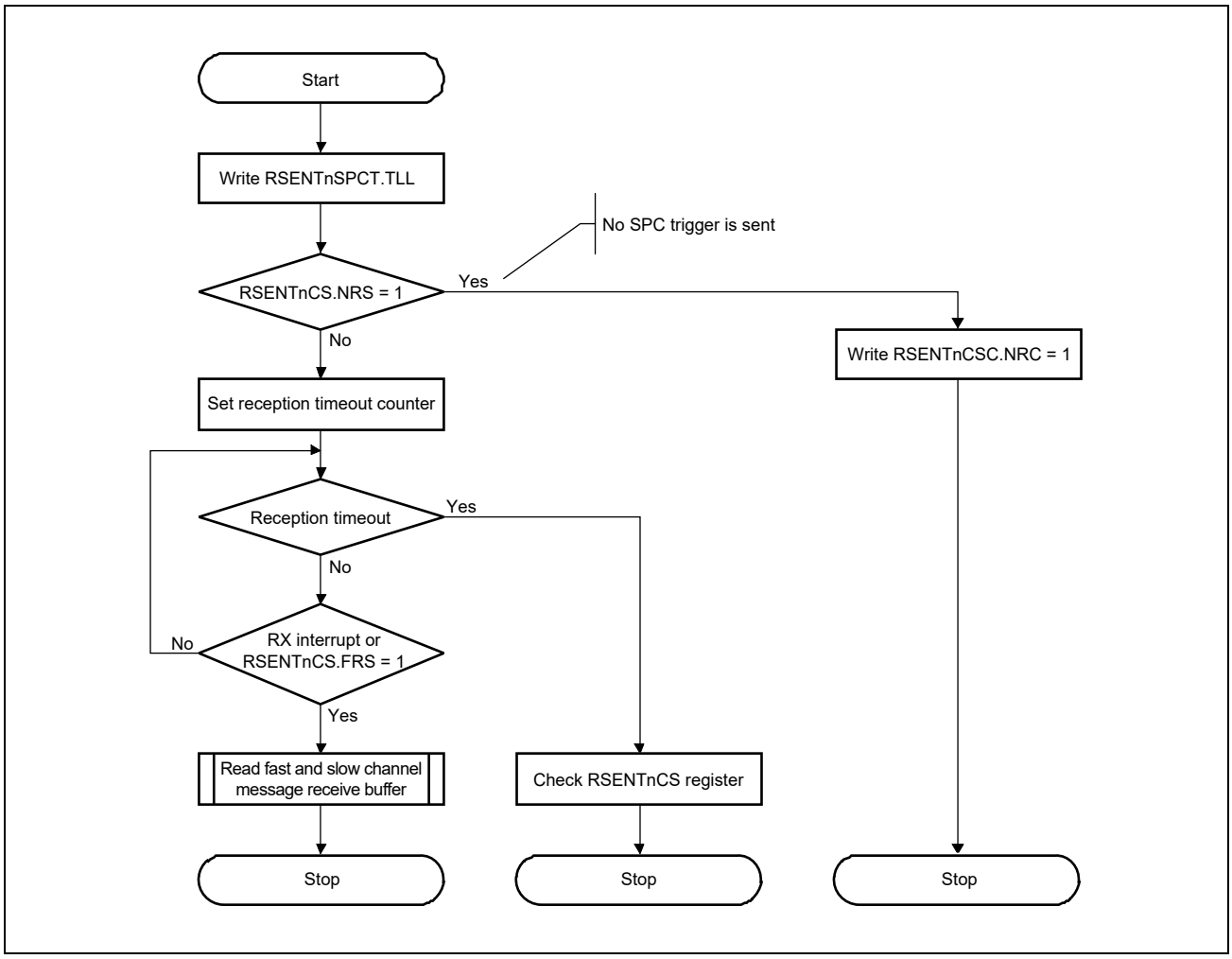


Figure 15.20 Transmission Flow

15.4.5 Interrupts and Checks

The RSENT module provides two interrupt lines. A successful fast channel receive interrupt notifies the CPU that the fast channel message receive buffer was updated and is holding a set of valid received data. Also, the reception status bit is set (RSENTnCS.FRS).

A status interrupt notifies the CPU that at least one of the error flags or message lost flags in the RSENTnCS register is set.

Whether the status flags in the RSENTnCS register are to contribute to the generation of interrupt events can be set individually.

The CRC check can be disabled for the slow channel and fast channel individually. If the check is disabled, the CRC of the received message is not checked and the related error flag is never set.

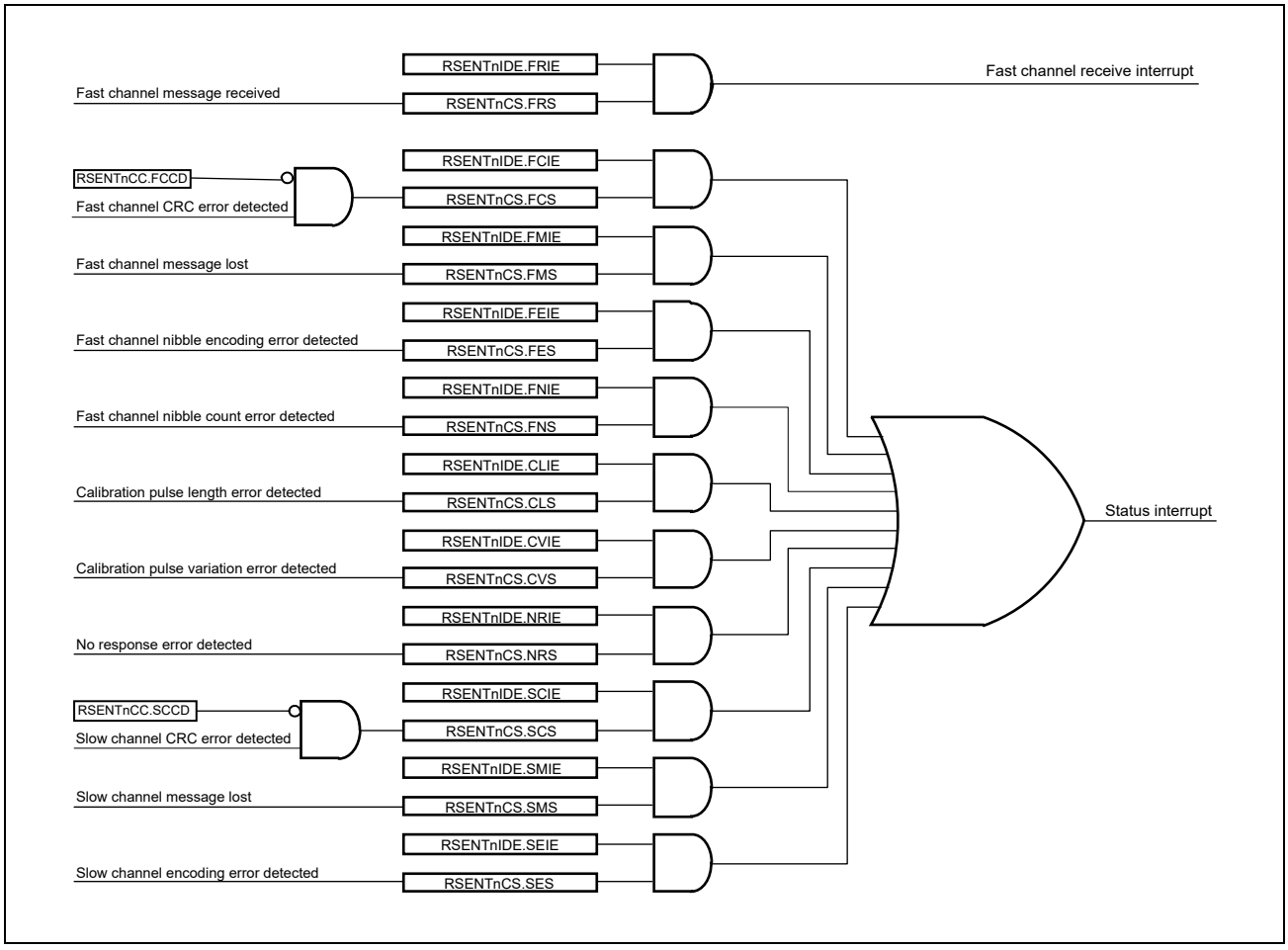


Figure 15.21 Interrupt Structure

Table 15.31 gives an outline of the relationship between set status flags and the buffer update.

Table 15.31 Status Flag Influence on the Receive Buffer Behaviors

RSENTnCS	Fast Channel Message Receive Buffer	Slow Channel Message Receive Buffer
FRS	Updated	Updated if all status and communication nibbles of slow channel messages are received and RSENTnCS.SES = 0 and RSENTnCS.SCS = 0.
FCS	Not updated	Receive processing is suspended. New start condition is searched for.
FMS	Message lost	Not impacted
FES	Not updated	Receive process aborted. Search for new start condition
FNS	Not updated	Receive process aborted. Search for new start condition
CLS	Not updated	Receive process aborted. Search for new start condition
CVS	Not updated	Receive process aborted. Search for new start condition
NRS	Not updated	Receive process aborted. Search for new start condition
SCS	Not impacted	Not updated
SMS	Not impacted	Message lost
SES	Not impacted	Receive process aborted. Search for new start condition

Section 16 Window Watchdog Timer A (WDTA)

This section contains a generic description of the window watchdog timer (WDTA).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of WDTA.

16.1 Features of RH850/C1M-A WDTA

16.1.1 Number of Units

This microcontroller has the following number of WDTA units.

Table 16.1 Number of Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of Units	2	1
Name	WDTAn (n = 0, 1)	WDTAn (n = 0)

Table 16.2 Index

Index	Meaning
n	Throughout this section, the individual WDTA units are identified by the index “n” (n = 0, 1) (n = 0, 1 for RH850/C1M-A2; n = 0 for RH850/C1M-A1); for example, WDTAnWDTE is the WDTAn enable register.

16.1.2 Register Base Address

WDTA base addresses are listed in the following table.

WDTA register addresses are given as offsets from the base addresses in general.

Table 16.3 Register Base Address

Base Address Name	Base Address
<WDTA0_base>	FFED 0000$_H$
<WDTA1_base>	FF8D 1000$_H$

16.1.3 Clock Supply

The WDTA clock supply is shown in the following table.

Table 16.4 WDTAn Clock Supply

WDTAn	Clock Name for the Unit	Supply Clock Name
WDTA0	WDTATCKI	WDTCLKI
	PCLK	CLKC_LSB (non-modulated low-speed peripheral clock)
WDTA1	WDTATCKI	WDTCLKI
	PCLK	CLKC_LSB (non-modulated low-speed peripheral clock)

16.1.4 Interrupts

WDTA can generate the following interrupt requests.

Table 16.5 Interrupt Requests

WDTAn Signals	Function	Interrupt Number
WDTA0		
WDTA0TIT	75% interrupt	54
WDTA1		
WDTA1TIT	75% interrupt	55*1

Note 1. This is not supported by C1M-A1.

16.1.5 Reset Sources

WDTA is initialized by reset sources listed in the following table.

Table 16.6 Reset Sources

Unit Name	Reset Source
WDTA0	All reset sources
WDTA1	All reset sources

16.1.6 WDTA Start-Up Options

The start-up options determine the start-up configuration of WDTA0 after reset release. The start-up options are described in **Table 16.7, WDTA0 Start-Up Options**. They are not supported in WDTA1.

Table 16.7 WDTA0 Start-Up Options

Start-Up Option	Function	Flash Option Assignment
OPWDOVF[2:0]	Setting for overflow interval time	OPBT0[27:25]
OPWDWS[1:0]	Setting for window open period	OPBT0[29:28]
OPWDINT	Setting for 75% interrupt request	OPBT0[30]
OPWDRUN	Setting for start mode	OPBT0[31]

16.2 Overview

16.2.1 Functional Overview

The WDTA has the following functions:

- Selection of the operating mode after release from reset using the option byte Starting/stopping of the counter after the WDTA is reset, enabling or disabling of 75% interrupt requests, the window-open period, and overflow time can be selected.
- WDTA trigger function
  Writing an activation code to the WDTA trigger register starts WDTA and restarts the counter.
- 75% interrupt request signals
  An interrupt request signal can be generated when the WDTA counter reaches 75% of the overflow interval time (this function can be enabled or disabled by the setting of WDTAnMD.WDTAnWIE).
- Window function
  The period during which writing to the WDTA trigger register is valid (window-open period) can be set. Writing to the WDTA trigger register at a time outside the window-open period causes an error.
- WDTA error detection
  When an error is detected, the WDTAnTRES signal indicates the error to the ECM. For details about the error sources, refer to **Section 16.5.2, Error Detection**.

16.2.2 Block Diagram

Figure 16.1 shows the main components of the WDTA.

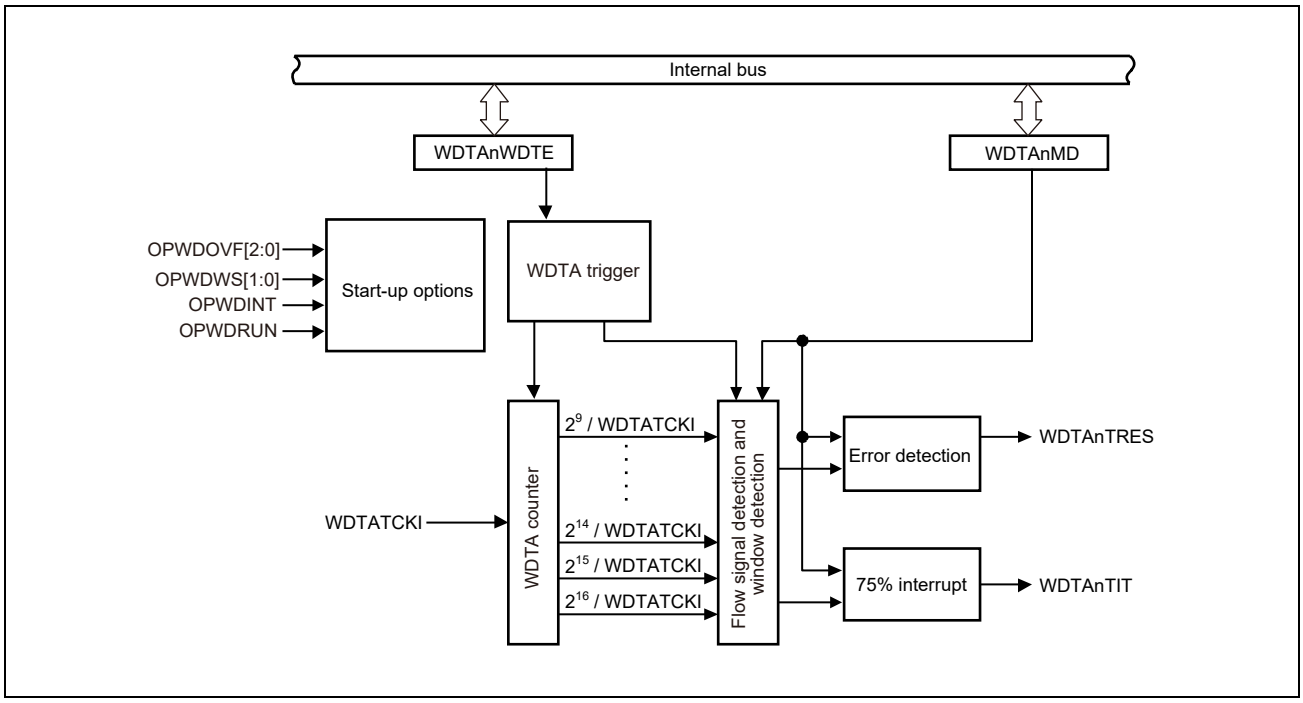


Figure 16.1 Block Diagram of the WDTA

16.3 Register

16.3.1 List of Registers

WDTA registers are listed in the following table.

For details about <WDTAn_base>, see **Section 16.1.2, Register Base Address**.

Table 16.8 WDTA Registers Overview

Module Name	Register Name	Symbol	Address
WDTAn	WDTAn enable register	WDTAnWDTE	<WDTAn_base> + 0_H
WDTAn	WDTAn mode register	WDTAnMD	<WDTAn_base> + C_H

16.3.2 WDTAnWDTE — WDTA Enable Register

This register is the WDTA start control and trigger register.

Writing AC_H to this register generates a WDTA trigger and starts or restarts the WDTA counter.

Refer to **Section 16.5.1.6, WDTA Trigger**, for details.

The writable value of this register is only AC_H.

Access: This register can be read/written in 8-bit units.

Address: <WDTAn_base> + 0000_H

Value after reset: (1) WDTA0

The value after reset of WDTA0 depends on the settings of OPWDRUN.

The initial state of OPWDRUN in the product as shipped is OPWDRUN = 0_B.

Accordingly, the initial register value for WDTA0 at the time of shipment is $2C_H$.

(2) WDTA1*1

The initial register value for WDTA1 is $2C_H$.

For details on the OPWDRUN settings, see **Section 16.5, Function**. Any reset source triggers initialization.

Note 1. Not supported for C1M-A1.

Bit	7	6	5	4	3	2	1	0
	WDTAnRUN[7:0]							
Value after reset	0	0	1	0	1	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.9 WDTAnWDTE Register Contents

Bit Position	Bit Name	Function
7 to 0	WDTAnRUN[7:0]	Writing a fixed activation code (AC_H) generates a WDTA trigger to control WDTAn count start and restart. Writing a value other than AC_H generates an error. After WDTAn starts, it cannot be stopped.

The value of bit WDTAnRUN[7] after reset depends on other start-up options as listed below.

Table 16.10 WDTAnRUN[7] Values after Reset

Start-Up Options	
OPWDRUN	WDTAnRUN[7] Value after Reset
1	1
0	0

16.3.3 WDTAnMD — WDTA Mode Register

This register specifies the overflow interval time, the 75% interrupt enable/disable, and the window-open period.

It can be updated only once after reset release and before the first trigger. The updated value will be effective after the next WDTA trigger.

Updating this register after the first WDTA trigger has been generated leads to error detection, but an error does not occur if the same value has been written to it.

Access: This register can be read/written in 8-bit units.

Address: <WDTAn_base> + $000C_H$

Value after reset: (1) WDTA0

The value after reset of WDTA0 depends on the settings of OPWDOVF2 to OPWDOVF0, OPWDINT, and OPWDWS1 to OPWDWS0. Since the settings of the option bytes are for the initial state of WDTA0 in the product as shipped to be the same as that of WDTA1, OPWDOVF2 to OPWDOVF0, OPWDINT, and OPWDWS1 and OPWDWS0 are all set to 1_B. Accordingly, the initial register value for WDTA0 at the time.

(2) WDTA1*1

The initial register value for WDTA1 is $7F_H$.

This register is initialized by a reset of any type.

Note 1. Not supported for C1M-A1.

Bit	7	6	5	4	3	2	1	0
	—	WDTAnOVF[2:0]			WDTAnWIE	—	WDTAnWS[1:0]	
Value after reset	0	1	1	1	1	1	1	1
R/W	R	R/W	R/W	R/W	R/W	R	R/W	R/W

Table 16.11 WDTAnMD Register Contents

Bit Position	Bit Name	Function
7	Reserved	When written, write the value after reset.
6 to 4	WDTAnOVF [2:0]	Select the overflow interval time:

WDTAnOVF2	WDTAnOVF1	WDTAnOVF0	Overflow Interval Time
0	0	0	2^9 / WDTATCKI
0	0	1	2^{10} / WDTATCKI
0	1	0	2^{11} / WDTATCKI
0	1	1	2^{12} / WDTATCKI
1	0	0	2^{13} / WDTATCKI
1	0	1	2^{14} / WDTATCKI
1	1	0	2^{15} / WDTATCKI
1	1	1	2^{16} / WDTATCKI

3	WDTAnWIE	Enables/disables the 75% interrupt request WDTAnTIT. 0: WDTAnTIT disabled 1: WDTAnTIT enabled
2	Reserved	When written, write the value after reset.
1, 0	WDTAnWS [1:0]	Select the window-open period:

WDTAnWS1	WDTAnWS0	Window-Open Period
0	0	25%
0	1	50%
1	0	75%
1	1	100%

16.4 Interrupt Sources

The WDTA detects the state of the WDTA counter value or illegal accesses to the WDTA-related registers, and generates an interrupt request. A WDTA interrupt request is described below.

(1) WDTAnTIT (WDTA timer count 75% interrupt request)
An interrupt request signal is generated at 75% of the counter overflow time of the WDTA timer. An interrupt request signal can be enabled or disabled by using the WDTA mode register (WDTAnMD).

16.5 Function

16.5.1 WDTA after Reset Release

16.5.1.1 Start modes

The WDTA provides two modes for the counter start after release from the reset state:

- Software trigger start mode (for WDTA0 and WDTA1)
  The counter value remains 0000_H after reset release. The counter is started with the first WDTA trigger.
- Default start mode (only for WDTA0)
  The counter starts automatically after release from the reset state. However, default start mode is disabled in serial programming mode even if OPWDRUN is set to 1_B for the star-up options.

WDTA1 is fixed to software trigger start mode, and cannot be set by the option byte.

16.5.1.2 Start mode selection (only for WDTA0)

The start mode can be selected by the start-up options. The start mode selection is listed in **Table 16.12**.

Table 16.12 Start Mode Selection

Start-Up Options		
OPWDRUN	Reset Type	Start Mode
0	Ignored	Software trigger
1		Default

16.5.1.3 WDTA settings after reset release

The WDTA settings are as follows between reset release and the first trigger:

Function	Setting after WDTA0 is Reset	Setting after WDTA1 is Reset
Start mode	Specified by start-up options	Software trigger mode
Overflow interval time	Specified by start-up options	2^{16}/WDTATCKI
75% interrupt mode	Specified by start-up options	75% interrupt enabled
Window-open period	Specified by start-up options	100%

Change WDTA settings

The setting of the WDTA mode register (WDTAnMD) is effective when the first WDTA trigger is generated (writing an activation code to WDTAnWDTE). Change the setting of the WDTAnMD register before a WDTA trigger is generated.

Setting of the WDTA by using WDTAnMD is possible only once. If the value set for WDTAnMD is changed after a WDTA trigger is generated, an error occurs. However, an error does not occur if the same value has been set.

16.5.1.4 Default start mode timing (only for WDTA0)

The default start mode timing and the changes to the WDTA settings are illustrated in **Figure 16.2**.

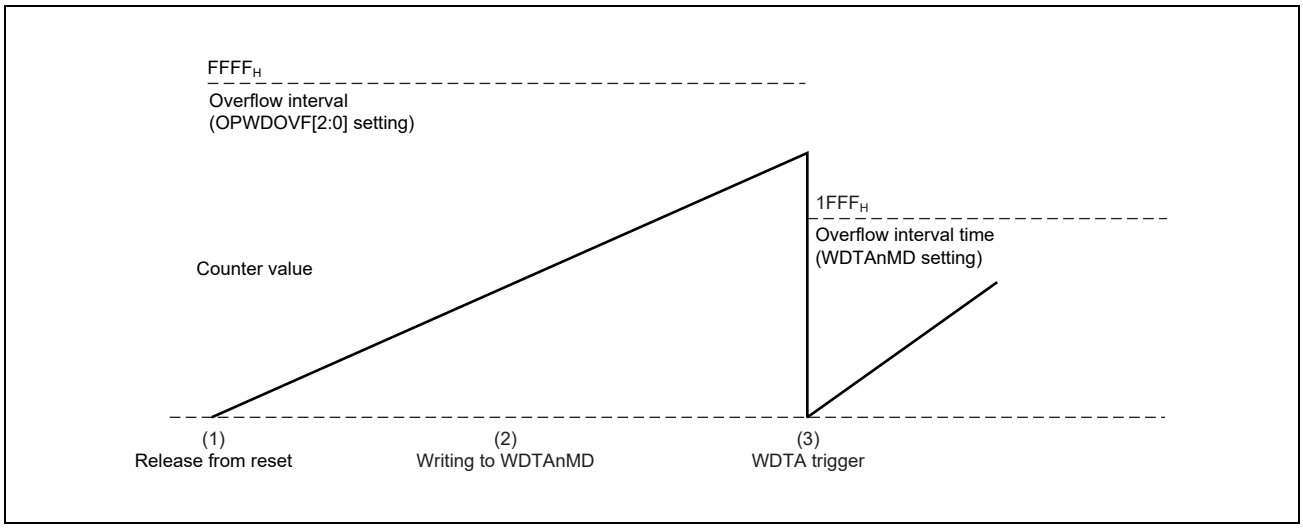


Figure 16.2 Timing Diagram of WDTA Start in Default Start Mode

The timing diagram in **Figure 16.2** shows the following behaviors:

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time after release from the reset state is set by start-up options.

Example: Overflow interval time after release from the reset state
$= 2^{16}$/WDTATCKI (OPWDOVF[2:0] = 111_B)

(2) WDTAnMD is set before a WDTA trigger is generated. Note, however, that the setting is not applied immediately.

(3) Write to the WDTA trigger register before the WDTA counter overflows.
The WDTAnMD setting is applied due to the WDTA trigger.

Example: Overflow interval time after a WDTA trigger is generated
$= 2^{13}$/WDTATCKI

16.5.1.5 Software trigger start mode timing (common to WDTA0 and WDTA1)

The software trigger start mode timing and the changes to the WDTA settings are illustrated in **Figure 16.3**.

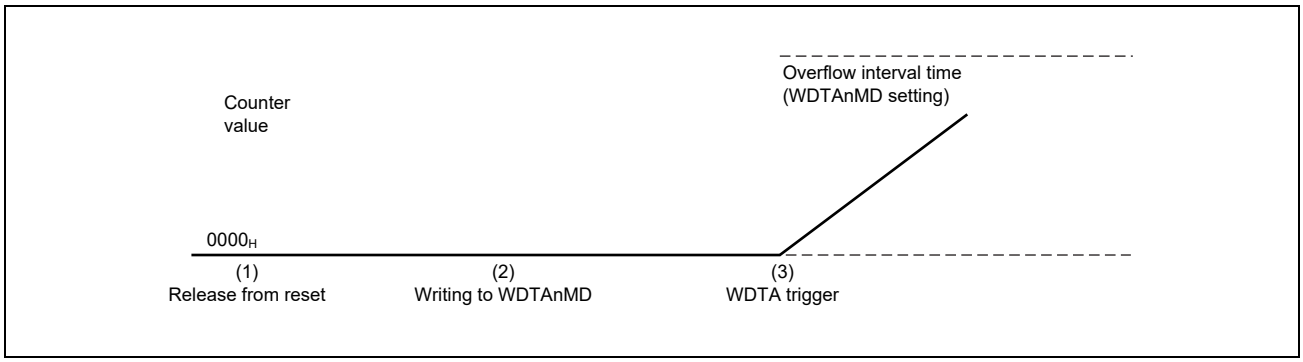


Figure 16.3 Timing Diagram of WDTA Start in Software Trigger Start Mode

The timing diagram in **Figure 16.3** shows the following behaviors:

(1) After release from the reset state, the counter remains 0000_H until the first WDTA trigger.
The overflow interval time is set by using the start-up options, but it does not have any effect.

(2) WDTAnMD is set before the first WDTA trigger. However, the settings are not applied immediately.

(3) The WDTA counter starts at the first WDTA trigger.
The overflow interval time and other settings specified in WDTAnMD are applied.

16.5.1.6 WDTA Trigger

Writing a special value called an activation code to the WDTA enable register (WDTAnWDTE) leads to generation of a WDTA trigger.

The WDTA trigger has the following functions:

- Starting the WDTA counter in software trigger start mode
- Restarting the WDTA counter
- WDTA mode setting by the WDTAnMD register
  (only for the first WDTA trigger after release from the reset state)

The WDTA can be triggered by writing a fixed activation code to the trigger register.

Table 16.13 Trigger Register and Activation Code

Type of Activation Code	Trigger Register	Activation Code
Fixed	WDTAnWDTE	AC_H

16.5.2 Error Detection

The WDTA detects an overflow of the WDTA counter and illegal operations as an error. The conditions for error detection are:

- WDTA counter overflow.
- Wrong activation code is written to the WDTA trigger register.

- Writing to the trigger register at the time outside the window-open period.
- When an attempt is made to change the setting value of the WDTA mode register WDTAnMD after the first WDTA trigger has been generated.
- When updating the setting value of the WDTA mode register WDTAnMD twice before the first WDTA trigger is generated.

16.5.3 WDTA Error Mode

When an error is detected, the WDTAnTRES signal indicates the error to the ECM.

Figure 16.4 shows generation of a reset when the counter overflows and default start mode is selected.

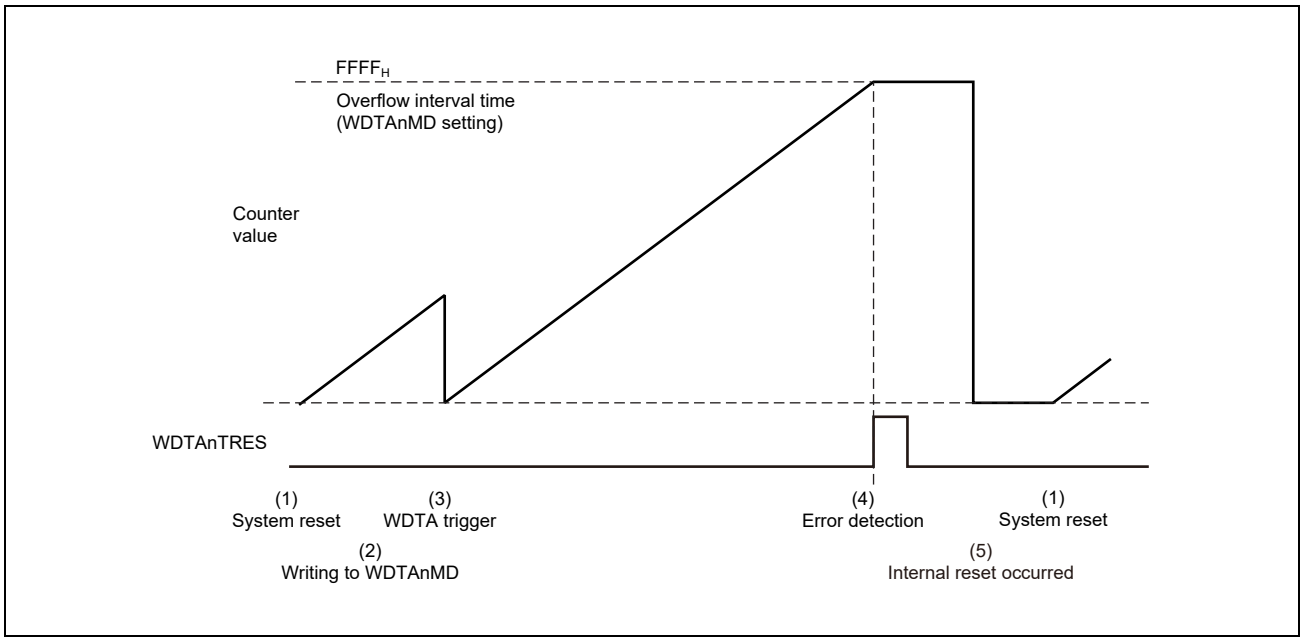


Figure 16.4 Timing Diagram of WDTA Internal Reset Generation

The timing diagram in **Figure 16.4** shows the following behaviors:

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time after release from the reset state is set by using start-up options.

(2) WDTAnMD is set before a WDTA trigger is generated.
In this case, 2^{16}/WDTATCKI is set for the overflow interval time.

(3) The WDTAnMD setting is applied due to the WDTA trigger.

(4) If the counter overflows, an error is detected and the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

(5) If an internal reset occurs due to the ECM and other sources, the counter is cleared and stopped until release from the reset state.

16.5.4 75% Interrupt Request Signals

When the WDTA counter reaches 75% of the time set for the overflow interval, an interrupt request WDTAnTIT is generated.

By use of WDTAnMD.WDTAnWIE this function can be enabled or disabled afterwards.

Figure 16.5 shows the 75% interrupt request generation under following conditions:

- Default start mode is selected.
- 75% interrupt request is enabled after the first WDTA trigger is generated.
- WDTA overflow interval time: 2^{16}/WDTATCKI.

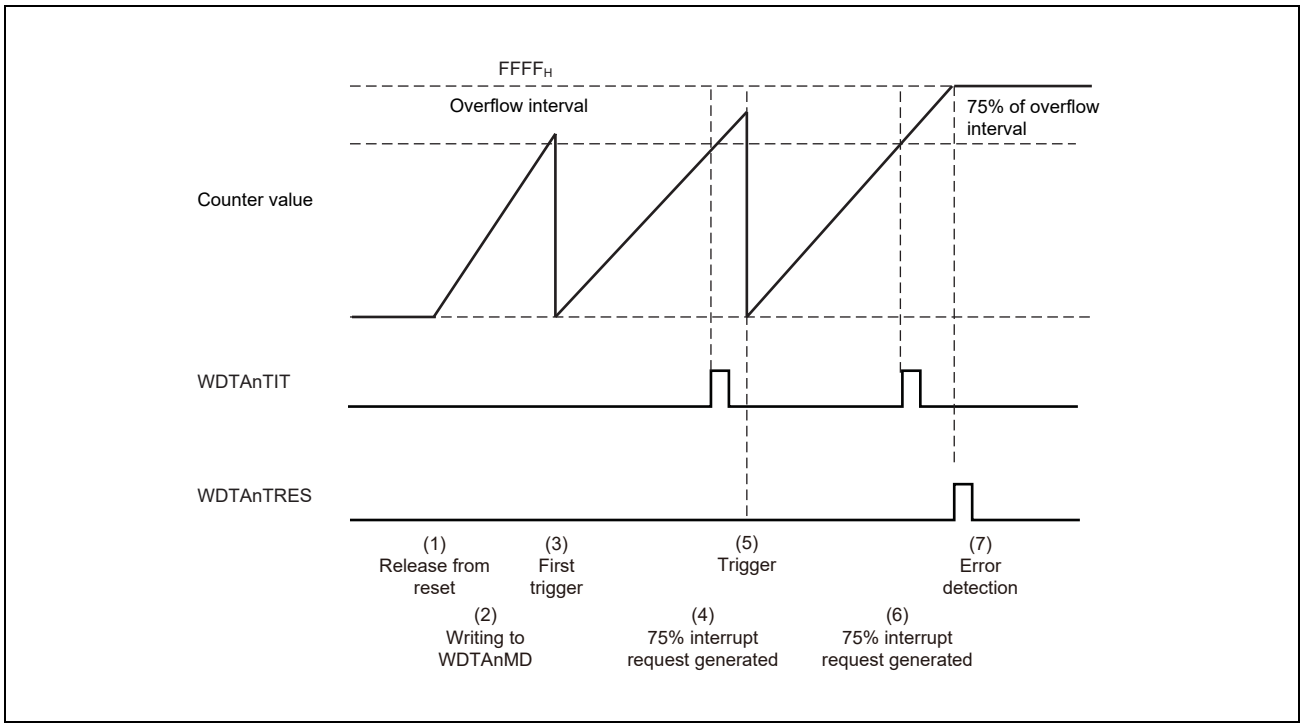


Figure 16.5 Timing Diagram of WDTA 75% Interrupt Request Signals

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time period after release from the reset state is set by using the start-up options.

(2) WDTAnMD is set before the WDTA trigger is generated. In this figure, 2^{16}/WDTATCKI is set as the overflow interval time period.

(3) The WDTAnMD setting is applied at the WDTA trigger.

(4) When the WDTA counter reaches 75% of the set overflow interval time, an interrupt request WDTAnTIT is generated.

(5) The counter restarts at the WDTA trigger.

(6) Then the WDTA counter reaches 75% of the set overflow interval time, an interrupt request WDTAnTIT is generated.

(7) When the counter overflows, an error is detected and the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

16.5.5 Window Function

The period when a WDTA trigger is valid (window-open period) can be set.

If the window-open period is set to the value less than 100%, an error occurs by the WDTA trigger generated at the time outside the window-open period. The window-open period after release from the reset state is 100%. The period is set to the value set in the WDTAnMD.WDTAnWS[1:0] after the first WDTA trigger is generated.

Figure 16.6 shows window function operations under the following conditions.

- Default start mode is selected.
- The 25% window open period is valid (WDTAnWS[1:0] = 00_B) after the first WDTA trigger.
- WDTA overflow interval time period: 2^{16}/WDTATCKI

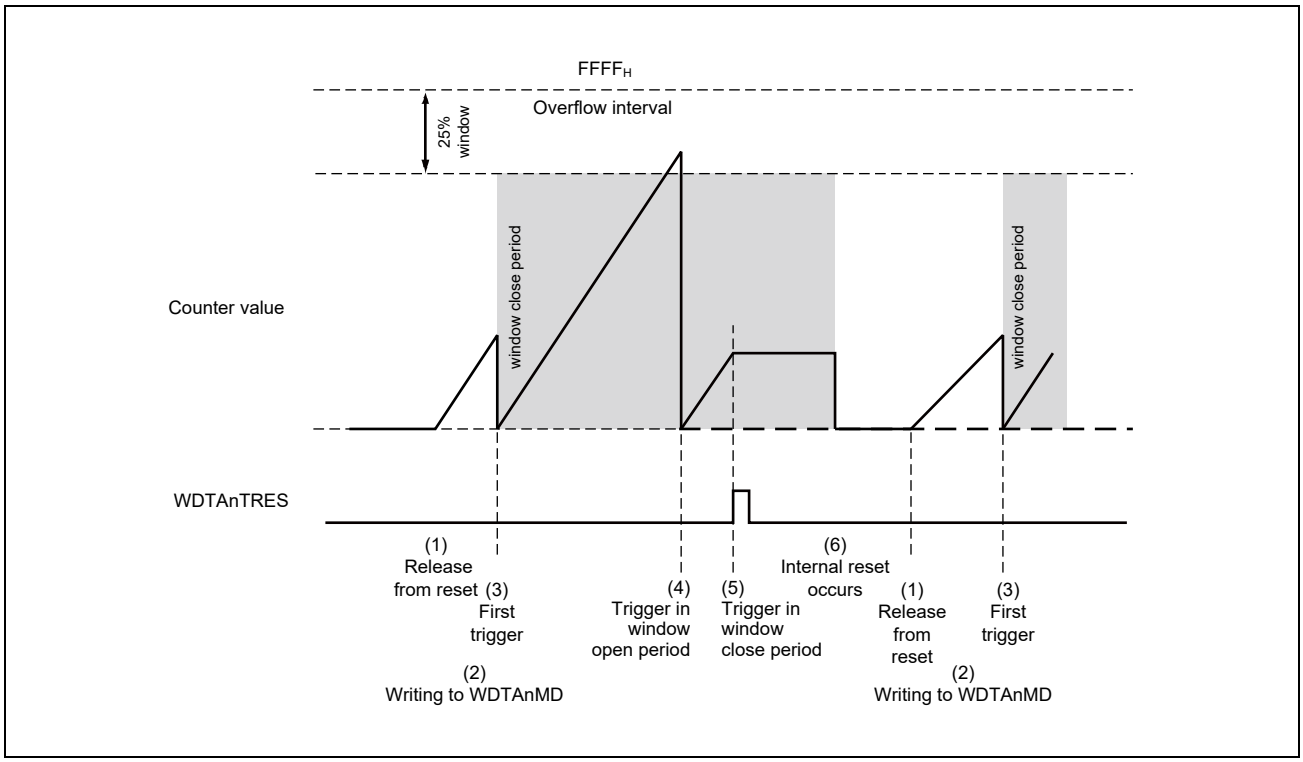


Figure 16.6 Timing Diagram of WDTA Window Function

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time period after release from the reset state is set by using the start-up options.

(2) WDTAnMD is set before the WDTA trigger is generated. In this figure, 2^{16}/WDTATCKI is set as the overflow interval time period.

(3) The WDTAnMD setting is applied at the WDTA trigger.

(4) The WDTA counter restarts at the WDTA trigger during the window open period.

(5) An error is detected at the WDTA trigger during the window close period, and then the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

(6) If an internal reset occurs due to the ECM and other sources, the counter is cleared and stopped until release from the reset state.

Section 17 OS Timer (OSTM)

This section contains a generic description of the OS timer (OSTM).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of the OSTM.

17.1 Features of RH850/C1M-A OSTM

17.1.1 Number of Units

This microcontroller has the following number of OSTM units.

Table 17.1 Number of Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of Units	4	3
Name	OSTMn (n = 0 to 3)	OSTMn (n = 0 to 2)

Table 17.2 Index

Index	Meaning
n	Throughout this section, the individual OSTM units are identified by the index “n” (n = 0 to 3) (n = 0 to 3 for RH850/C1M-A2; n = 0 to 2 for RH850/C1M-A1); for example, OSTMnCNT is the OSTM counter register.

17.1.2 Register Base Address

OSTM base addresses are listed in the following table.

OSTM register addresses are given as offsets from the base addresses in general.

Table 17.3 Register Base Address

Base Address Name	Base Address
<OSTM0_base>	FFEC 0000$_H$
<OSTM1_base>	FF8C 1000$_H$
<OSTM2_base>	FFEC 2000$_H$
<OSTM3_base>	FF8C 3000$_H$

Note: RH850/C1M-A1 does not have OSTM3.

17.1.3 Clock Supply

The OSTM clock supply is shown in the following table.

Table 17.4 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
OSTMn	PCLK	CLKC_LSB (unmodulated low-speed peripheral clock)

Note: RH850/C1M-A1 does not have OSTM3.

17.1.4 Interrupt Requests

OSTM interrupt requests are listed in the following table.

Table 17.5 Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt Number
OSTM0TINT	OSTM0 interrupt	25
OSTM1TINT	OSTM1 interrupt	26
OSTM2TINT	OSTM2 interrupt	27
OSTM3TINT	OSTM3 interrupt	28

Note: RH850/C1M-A1 does not have OSTM3.

17.1.5 Reset Sources

OSTM reset sources are listed in the following table. The OSTM is initialized by these reset sources.

Table 17.6 Reset Source

Unit Name	Reset Source
OSTMn	All reset sources

17.2 Overview

The OSTM is a 32-bit timer/counter.

It can be used in interval timer mode or in free-run compare mode. The settings for operating mode specify the direction of counting (up or down) to control the generation of interrupt requests.

A counter-start signal (OSTMnTSST) is input to the OSTM, allowing synchronization of these timers with other peripheral functions.

17.2.1 Functional Overview

The OSTM has the following features.

- Two operating modes
  - Interval timer mode
  - Free-run compare mode
- Simultaneous start trigger function between the units
- OSTMnTINT interrupt

17.2.2 Block Diagram

The following block diagram shows the main components of the OSTM.

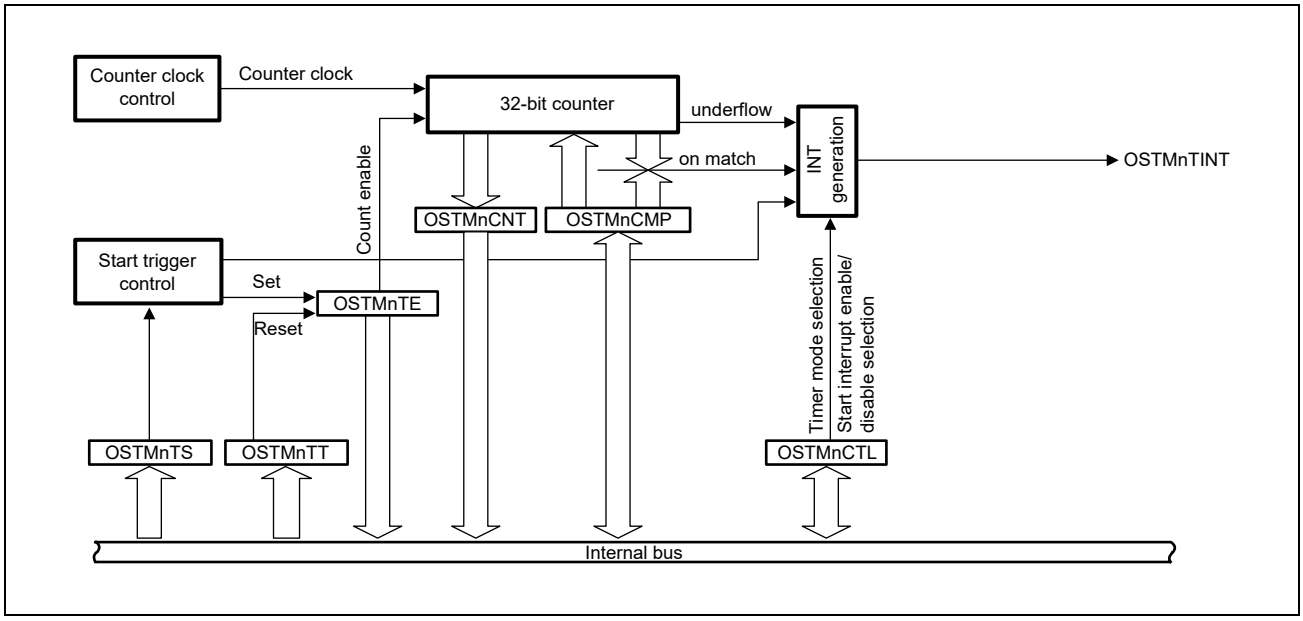


Figure 17.1 Block Diagram of the OSTM

17.2.3 Counter Clock

The OSTM uses PCLK as the counter clock.

17.2.4 Interrupt Request (OSTMnTINT)

By default, an OSTMnTINT interrupt request is generated on counter underflow (interval timer mode) or when the counter matches the compare value (free-run compare mode).

An interrupt request can also be generated on starting and restarting of the counter. This is controlled by the OSTMnCTL.OSTMnMD0 bit.

Since OSTMnTINT triggers toggling of the OSTMnTTOUT output in timer-output toggling mode (OSTMnTOE.OSTMnTOE is 1), the setting of the OSTMnCTL.OSTMnMD0 bit also affects the output (OSTMnTTOUT).

This is illustrated in the following figure.

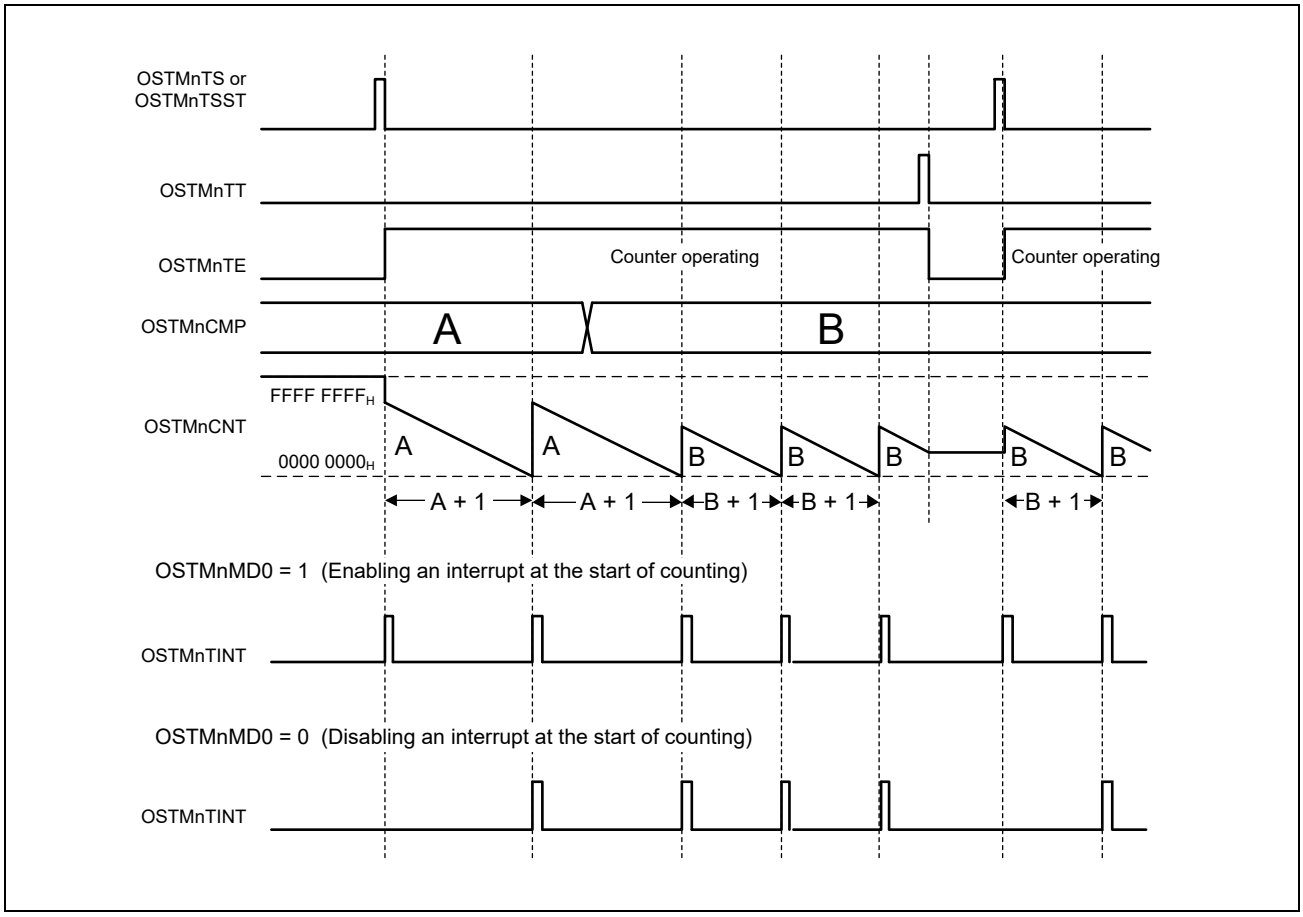


Figure 17.2 Generating an Interrupt when Counting Starts (in Interval Timer Mode)

17.3 Registers

17.3.1 List of Registers

OSTM registers are listed in the following table.

For details about <OSTMn_base>, see **Section 17.1.2, Register Base Address**.

Table 17.7 Registers

Module	Register	Symbol	Address
OSTMn	OSTMn compare register	OSTMnCMP	<OSTMn_base> + 00_H
OSTMn	OSTMn counter register	OSTMnCNT	<OSTMn_base> + 04_H
OSTMn	OSTMn output register	OSTMnTO	<OSTMn_base> + 08_H
OSTMn	OSTMn output enable register	OSTMnTOE	<OSTMn_base> + $0C_H$
OSTMn	OSTMn count enable status register	OSTMnTE	<OSTMn_base> + 10_H
OSTMn	OSTMn count start trigger register	OSTMnTS	<OSTMn_base> + 14_H
OSTMn	OSTMn count stop trigger register	OSTMnTT	<OSTMn_base> + 18_H
OSTMn	OSTMn control register	OSTMnCTL	<OSTMn_base> + 20_H

17.3.2 OSTMnCMP — OSTMn Compare Register

This register stores the start value of the counter or the value with which the counter is compared, depending on the operation mode.

Access: This register can be read/written in 32-bit units.

Address: <OSTMn_base> + 00_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCMP[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCMP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.8 OSTMnCMP Register Contents

Bit Position	Bit Name	Function
31 to 0	OSTMnCMP [31:0]	• In interval timer mode: start value of the counter • In free-run compare mode: compare value

17.3.3 OSTMnCNT — OSTMn Counter Register

This register indicates the counter value of the timer.

Access: This register can be read in 32-bit units.

Address: <OSTMn_base> + 04_H

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCNT[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCNT[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.9 OSTMnCNT Register Contents

Bit Position	Bit Name	Function
31 to 0	OSTMnCNT [31:0]	Timer counter value

Table 17.10 shows the correspondence between OSTMn's operating mode, counting direction, and initial value. The initial value is the value read from the counter after a change to the operating mode.

Table 17.10 Correspondence Between Operating Mode, Counting Direction and Initial Value

Timer Operating Mode	OSTMnCTL.OSTMnMD1	Counting Direction	Initial Value
Interval timer mode	0*1	Down	FFFF $FFFF_H$
Free-run compare mode	1	Up	0000 0000_H

Note 1. Value after reset.

17.3.4 OSTMnTO — OSTMn Output Register

This register is used to specify and read the level of an OSTMnTTOUT output signal. The setting of this register is only valid in OSTMn (n = 0).

Access: This register can be read/written in 8-bit units. Writing can only proceed when the software control mode is selected (OSTMnTOE.OSTMnTOE = 0).

Address: <OSTMn_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTO
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 17.11 OSTMnTO Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is returned. When writing, write the value after reset.
0	OSTMnTO	This bit is used to specify and read the level of the OSTMnTTOUT output signal. 0: Low level 1: High level

17.3.5 OSTMnTOE — OSTMn Output Enable Register

This register specifies OSTMnTTOUT output mode. The setting of this register is only valid in OSTMn (n = 0).

Access: This register can be read/written in 8-bit units.

Address: <OSTMn_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTOE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 17.12 OSTMnTOE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is returned. When writing, write the value after reset.
0	OSTMnTOE	This bit specifies the OSTMnTTOUT output mode. 0: Software control mode: The level corresponding to the setting of OSTMnTO.OSTMnTO is output to OSTMnTTOUT. 1: Timer-output toggling mode: OSTMnTTOUT output is toggled whenever an OSTMnTINT interrupt request is generated.

17.3.6 OSTMnTE — OSTMn Count Enable Status Register

This register indicates whether the counter is enabled or disabled.

Access: This register can be read in 8-bit units.

Address: <OSTMn_base> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 17.13 OSTMnTE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read.
0	OSTMnTE	Indicates whether the counter is enabled or disabled: 0: Counter disabled 1: Counter enabled This bit is set to 1 in response to OSTMnTS.OSTMnTS being set to 1, or to OSTMnTSST being 1. Setting OSTMnTT.OSTMnTT to 1 resets this bit to 0.

NOTE

If the counter is disabled, the counter value OSTMnCNT retains its value. If the counter is restarted, it

- restarts counting down from the value in the OSTMnCMP register if it is in interval timer mode or,
- restarts counting up from the counter value 0000 0000_H if it is in free-run compare mode.

17.3.7 OSTMnTS — OSTMn Count Start Trigger Register

This register starts the counter.

Access: This register can be written in 8-bit units. It is always read as 00_H.

Address: <OSTMn_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 17.14 OSTMnTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	OSTMnTS	Starts the counter: 0: This setting disables the counter. 1: Starts the counter and sets OSTMnTE.OSTMnTE = 1. • In interval timer mode, a forced restart is executed if this bit is set while OSTMnTE.OSTMnTE = 1. • In free-run compare mode, setting this bit is ignored as long as OSTMnTE.OSTMnTE = 1.

17.3.8 OSTMnTT — OSTMn Count Stop Trigger Register

This register stops the counter.

Access: This register can be written in 8-bit units. It is always read as 00_H.

Address: <OSTMn_base> + 18_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 17.15 OSTMnTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	OSTMnTT	Stops the counter: 0: This setting disables the counter. 1: Stops the counter and clears the OSTMnTE.OSTMnTE bit.

17.3.9 OSTMnCTL — OSTMn Control Register

This register specifies the operating mode for the counter and controls the generation of OSTMnTINT interrupt requests when counting starts.

Although this register is readable and writable, writing to it is only possible when OSTMnTE.OSTMnTE = 0; that is, the register becomes read-only when OSTMnTE.OSTMnTE = 1.

Access: This register can be read/written in 8-bit units.

Address: <OSTMn_base> + 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OSTMnMD1	OSTMnMD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.16 OSTMnCTL Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, value after reset is read. When writing, write the value after reset.
1	OSTMnMD1	Specifies the operating mode for the counter: 0: Interval timer mode 1: Free-run compare mode
0	OSTMnMD0	Controls the generation of OSTMnTINT interrupt requests at the start of counting: 0: Interrupts when counting starts are disabled. 1: Interrupts when counting starts are enabled.

17.4 Functions

17.4.1 Starting and Stopping the Timer

The OSTM is started and stopped as follows:

Starting the timer

The timer is started in either of the following ways:

- Setting the OSTMnTS.OSTMnTS bit to 1
- Setting the OSTMnTSST signal to the high level (when the Simultaneous start trigger function is in use)

The OSTMnTE.OSTMnTE status bit is set to 1.

The counter starts to count up or down in accord with the settings for operating mode. For details, refer to **Section 17.4.2, Interval Timer Mode** and **Section 17.4.3, Free-Run Compare Mode**.

If the OSTMnTS.OSTMnTS bit is to be used to start the timer, the OSTMnTSST input must correspond to logical zero.

Stopping the timer

Setting the OSTMnTT.OSTMnTT bit to 1 stops the timer.

This also clears the OSTMnTE.OSTMnTE status flag.

When the counter is stopped, values in the OSTMnTO and OSTMnCNT registers and the level of the OSTMnTTOUT output are retained until further counting operations start.

Simultaneous start trigger function

The OSTMnTSST signal output from the PIC1B module can be used to start multiple timers at the same time. Refer to **Section 24, Peripheral Interconnection (PIC)**.

17.4.2 Interval Timer Mode

In interval timer mode, the OSTM can be used as a reference timer generating interrupt requests at fixed intervals.

17.4.2.1 Basic Operation in Interval Timer Mode

In interval timer mode, the timer counts down from the value specified in the OSTMnCMP register. An OSTMnTINT interrupt request is generated when the counter underflows (reaches 0000 0000_H).

To select interval timer mode, set OSTMnCTL.OSTMnMD1 = 0.

New values can be written to the OSTMnCMP register at any time. If it is rewritten during count operation, the counter loads the new OSTMnCMP value when the next 0000 0000_H is reached. Then the counter continues with the new value.

OSTMnTINT period

The periods of OSTMnTINT is:

- OSTMnTINT generation period = counter clock period × (OSTMnCMP + 1)

The following figure shows the basic operation of the OSTM in interval timer mode when interrupts at the start of counting are enabled.

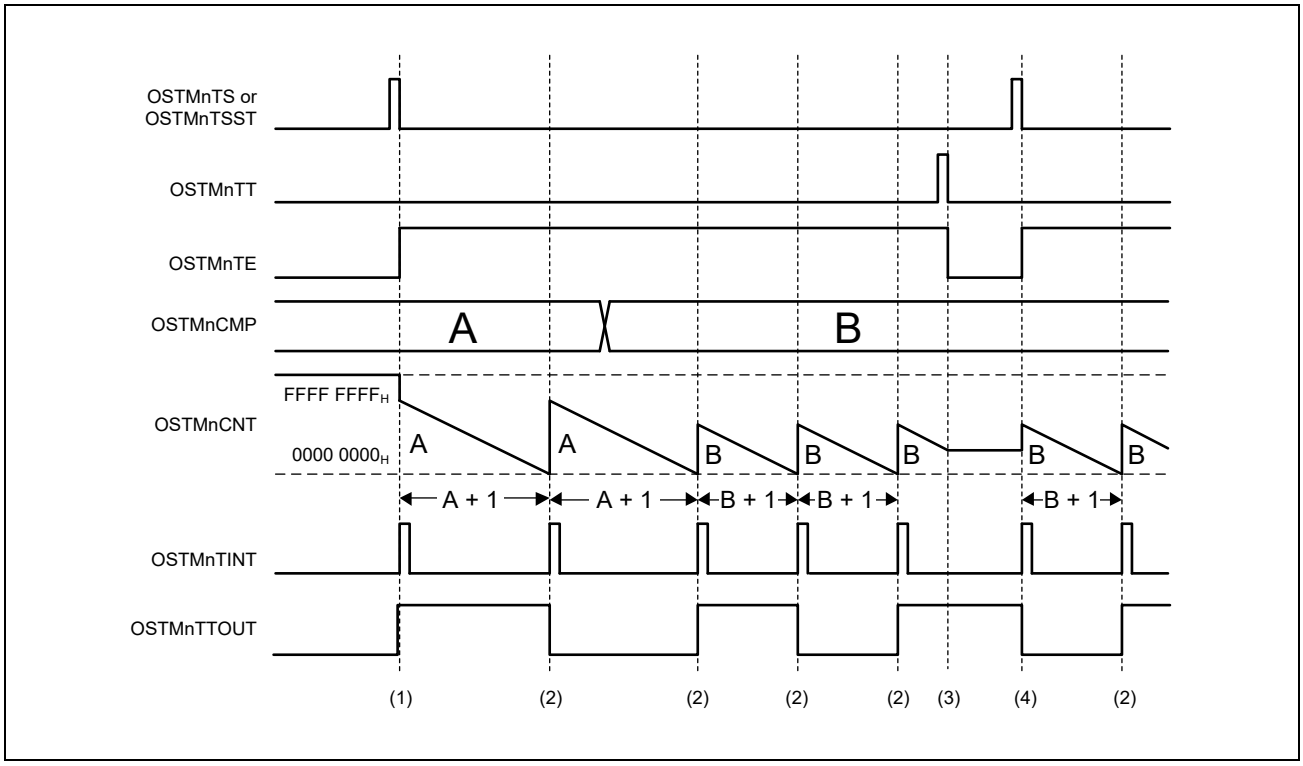


Figure 17.3 Timing Diagram of OSTM in Interval Timer Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1 or OSTMnTSST = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter.
The counter starts counting-down from the value of OSTMnCMP.
If OSTMnCTL.OSTMnMD0 is 1, OSTMnTINT interrupt requests are generated at the start of counting. The OSTMnCNT register indicates the current value of the counter.

(2) When the counter reaches 0000 0000_H, an OSTMnTINT interrupt request is generated. The counter loads the new start value from OSTMnCMP and continues counting down.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter. The counter retains its current value until it is restarted.

(4) When counting is restarted (OSTMnTS.OSTMnTS = 1, or OSTMnTSST = 1), the counter loads the new start value from OSTMnCMP and starts counting down.

Forced restart

The counter is forcibly restarted by setting OSTMnTS.OSTMnTS = 1 or by a transition of the OSTMnTSST signal from the high level to the low level during counting.

The counter loads the start value from the OSTMnCMP register and continues to counting down.

The following figure shows the forced restart of the OSTM in interval timer mode, with interrupts at the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1).

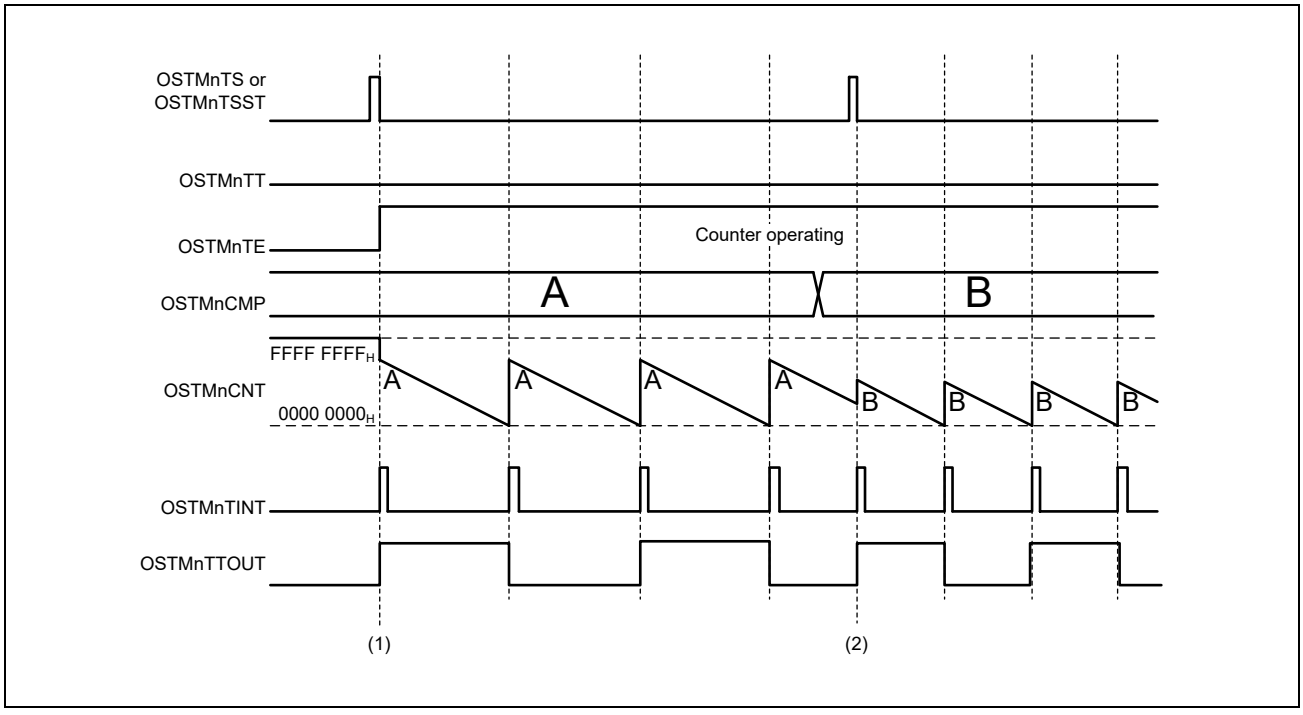


Figure 17.4 Timing Diagram of Forced Restart in Interval Timer Mode

Operations shown in the above timing diagram are as follows.

(1) The counter is started and stopped as described under **Figure 17.3, Timing Diagram of OSTM in Interval Timer Mode**.

(2) Setting OSTMnTS.OSTMnTS = 1 or OSTMnTSST = 1 restarts the counter while counting is in progress (i.e. while OSTMnTE.OSTMnTE = 1).
The counter immediately restarts counting down, starting with the current value of OSTMnCMP. When OSTMnCTL.OSTMnMD0 = 1, an OSTMnTINT interrupt request is generated when counting starts.

17.4.2.2 Operation when OSTMnCMP = 0000 0000$_H$

When OSTMnCMP = 0000 0000$_H$ the OSTM behaves as follows.

- When the counter is enabled, the OSTMnTINT interrupt request is always set to 1.
- When the OSTMnTTOUT signal is in timer-output toggling mode, the output is toggled on every cycle of PCLK.

The following figure shows operations of the OSTM when OSTMnCMP = 0000 0000$_H$, and interrupts at the start of counting are enabled.

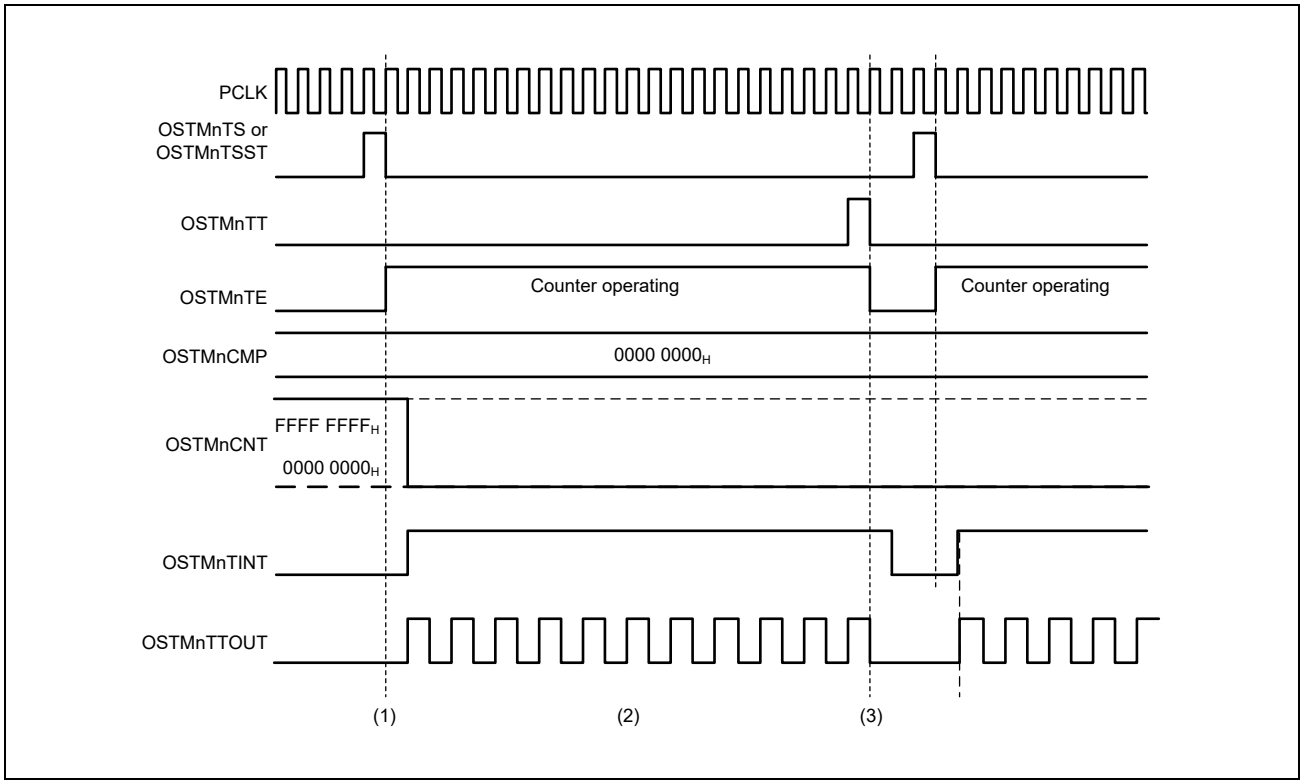


Figure 17.5 Timing Diagram when OSTMnCMP = 0000 0000$_H$ in Interval Timer Mode

The timing diagram above shows the following operations:

(1) The counter is reloaded with the value in OSTMnCMP, so the value 0000 0000_H is retained in OSTMnCMP.

(2) The OSTMnTINT interrupt request is continuously asserted.

(3) After the counter stops, the OSTMnTINT interrupt request signal is deasserted.

When interrupts at the start of counting are disabled, interrupts are not generated at the start timing of the counter.

17.4.2.3 Setting Procedure for Interval Timer Mode

The setting procedure in interval timer mode after reset release is described below:

Setting procedure

(1) Set the start value of the counter in the OSTMnCMP register.

(2) Select interval timer mode by clearing the OSTMnCTL.OSTMnMD1 bit.

(3) Select enabling or disabling of interrupts when counting starts (OSTMnCTL.OSTMnMD0).

17.4.3 Free-Run Compare Mode

17.4.3.1 Basic Operation in Free-Run Compare Mode

In free-run compare mode, the counter counts up from 0000 0000_H to FFFF $FFFF_H$. When the value of the OSTMnCMP register matches the current counter value, an OSTMnTINT interrupt request is output.

When free-run compare mode is used, set OSTMnCTL.OSTMnMD1 to 1.

New values can be written to the OSTMnCMP register at any time.

The following figure shows the basic operation of the OSTM in free-run compare mode with the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1).

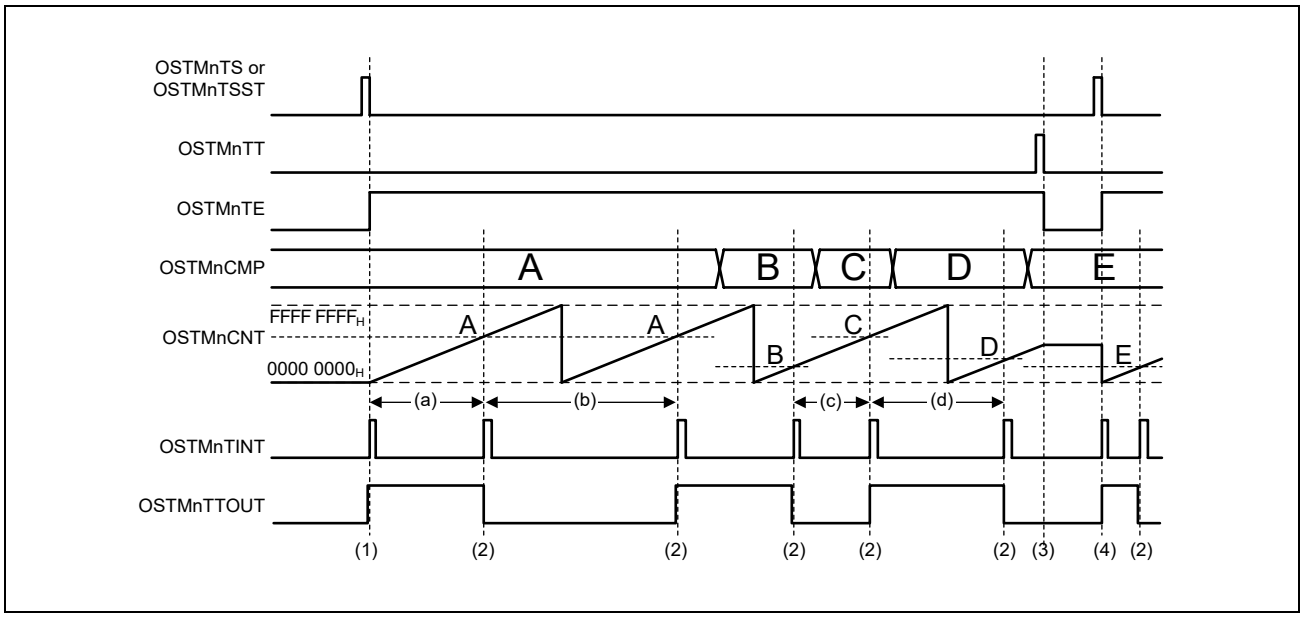


Figure 17.6 Timing Diagram of OSTM in Free-Run Compare Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1, or when OSTMnTSST = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter. The counter counts up from 0000 0000_H to FFFF $FFFF_H$. The OSTMnCNT register indicates the current value of the counter. When OSTMnCTL.OSTMnMD0 = 1, an interrupt request OSTMnTINT is generated when the counter starts.

(2) When the current counter value matches the value in the OSTMnCMP register, an OSTMnTINT interrupt request is generated.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter.
The counter retains its current value until it is restarted.

(4) Counting by the counter restarts from 0000 0000_H when OSTMnTS.OSTMnTS = 1, or when OSTMnTSST = 1.

OSTMnTINT period

The OSTMnTINT generation period is different at the start of counting and depends on the old and new compare value if OSTMnCMP is rewritten during operation.

Table 17.17 OSTMnTINT Generation Timing

Old Value for Comparison	New Value for Comparison	Counter Value at Time of Rewriting	Period of OSTMnTINT Generation	Label in Timing Diagram
Counter starts			(A + 1) × counter clock period	(a)
A	A	No rewriting	(FFFF $FFFF_H$ + 1) × counter clock period	(b)
B	C > B	B < counter value < C	(C – B) × counter clock period	(c)
C	D < C	Counter value > D,C	(FFFF $FFFF_H$ – C + D + 1) × counter clock period	(d)

Forced restart

Forced restarting does not proceed during counting even if the OSTMnTS.OSTMnTS bit is set, or if OSTMnTSST = 1. The counter ignores the attempted setting and continues counting.

17.4.3.2 Operation when OSTMnCMP = 0000 0000$_H$

The following figure shows the operation of the OSTM when OSTMnCMP = 0000 0000$_H$, interrupts at the start of counting are enabled (OSTMnCTL.OSTMnMD0 = 1).

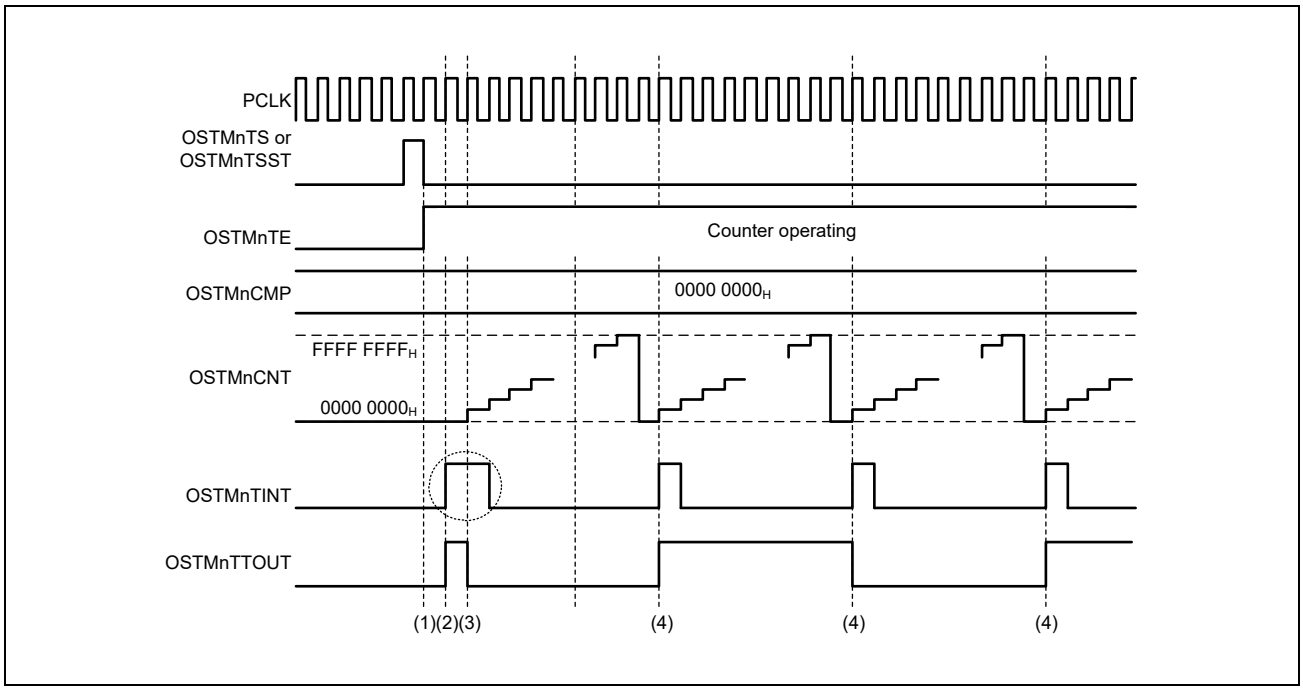


Figure 17.7 Timing Diagram when OSTMnCMP = 0000 0000$_H$ in Free-Run Compare Mode

The timing diagram above shows the following operations.

(1) Once the counter starts, it counts up from 0000 0000$_H$ to FFFF FFFF$_H$.

(2) An OSTMnTINT interrupt request is generated when counting starts.

(3) If the current counter value matches OSTMnCMP, an interrupt request OSTMnTINT is generated. If OSTMnCMP = 0000 0000$_H$ in the above case, OSTMnTINT is generated over two clock cycles.

(4) Every (FFFF FFFF$_H$ + 1) clock cycle the OSTMnTINT interrupt request is asserted.
When interrupts at the start of counting are disabled, interrupts are not generated at the start timing of the counter.

When interrupts at the start of counting are disabled, interrupts are not generated at the start timing of the counter.

17.4.3.3 Setting Procedure for Free-Run Compare Mode

The setting procedure in free-run compare mode after reset release is described below:

Setting Procedure

(1) Set the compare value in the OSTMnCMP register.

(2) Select free-run compare mode by setting the OSTMnCTL.OSTMnMD1 bit.

(3) Enable or disable interrupts when counting is started by the OSTMnCTL.OSTMnMD0 bit.

Section 18 Timer Array Unit D (TAUD)

18.1 Overview of RH850/C1M-A TAUD

18.1.1 Units

This LSI has the following number of TAUD units.

Table 18.1 Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	4	2
Name	TAUDn (n = 0 to 3)	TAUDn (n = 0, 1)

Table 18.2 Indices

Index	Meaning
n	Throughout this section, the individual TAUD units are identified by the index “n” (n = 0 to 3) (n = 0 to 3 for RH850/C1M-A2; n = 0, 1 for RH850/C1M-A1); for example, TAUD0CNT0 is the TAUDn control register 0.
m	The TAUD has up to 16 channels. Throughout this section, the individual channels are identified by the index “m” (m = 0 to 15).

The number of channels included in the individual TAUD units is listed in the following table.

Table 18.3 Channels

Unit Name	Number of Channels
TAUDn	16

18.1.2 Register Base Addresses

TAUD base addresses are listed in the following table.

TAUD register addresses are given as offsets from the base addresses in general.

Table 18.4 Register Base Addresses

Base Address Name	Base Address
<TAUD0_base>	FFE2 0000_H
<TAUD1_base>	FF82 1000_H
<TAUD2_base>*1	FFE2 2000_H
<TAUD3_base>*1	FFE2 3000_H

Note 1. RH850/C1M-A1 does not have TAUD2 and TAUD3.

18.1.3 Clock Supply

TAUD clock is listed in following table.

Table 18.5 TAUDn Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
TAUDn	PCLK	CLKC_HSB (Non-modulated high-speed peripheral clock)

18.1.4 Interrupts Requests

TAUD interrupt requests are listed in the following table.

Table 18.6 Interrupt Requests (1/2)

Interrupt Name	Outline	Interrupt Number	DMAC Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
TAUD0						
INTTAUD0I0	Channel 0 interrupt	98	19	—	19	—
INTTAUD0I1	Channel 1 interrupt	99	20	—	20	—
INTTAUD0I2	Channel 2 interrupt	100	21	—	21	—
INTTAUD0I3	Channel 3 interrupt	101	22	—	22	—
INTTAUD0I4	Channel 4 interrupt	102	23	—	23	—
INTTAUD0I5	Channel 5 interrupt	103	24	—	24	—
INTTAUD0I6	Channel 6 interrupt	104	25	—	25	—
INTTAUD0I7	Channel 7 interrupt	105	26	—	26	—
INTTAUD0I8	Channel 8 interrupt	106	27	—	27	—
INTTAUD0I9	Channel 9 interrupt	107	28	—	28	—
INTTAUD0I10	Channel 10 interrupt	108	29	—	29	—
INTTAUD0I11	Channel 11 interrupt	109	30	—	30	—
INTTAUD0I12	Channel 12 interrupt	110	31	—	31	—
INTTAUD0I13	Channel 13 interrupt	111	32	—	32	—
INTTAUD0I14	Channel 14 interrupt	112	33	—	33	—
INTTAUD0I15	Channel 15 interrupt	113	34	—	34	—
TAUD1						
INTTAUD1I0	Channel 0 interrupt	114	35	—	35	—
INTTAUD1I1	Channel 1 interrupt	115	36	—	36	—
INTTAUD1I2	Channel 2 interrupt	116	37	—	37	—
INTTAUD1I3	Channel 3 interrupt	117	38	—	38	—
INTTAUD1I4	Channel 4 interrupt	118	39	—	39	—
INTTAUD1I5	Channel 5 interrupt	119	40	—	40	—
INTTAUD1I6	Channel 6 interrupt	120	41	—	41	—
INTTAUD1I7	Channel 7 interrupt	121	42	—	42	—
INTTAUD1I8	Channel 8 interrupt	122	43	—	43	—
INTTAUD1I9	Channel 9 interrupt	123	44	—	44	—
INTTAUD1I10	Channel 10 interrupt	124	45	—	45	—
INTTAUD1I11	Channel 11 interrupt	125	46	—	46	—
INTTAUD1I12	Channel 12 interrupt	126	47	—	47	—
INTTAUD1I13	Channel 13 interrupt	127	48	—	48	—
INTTAUD1I14	Channel 14 interrupt	128	49	—	49	—
INTTAUD1I15	Channel 15 interrupt	129	50	—	50	—

Table 18.6 Interrupt Requests (2/2)

Interrupt Name	Outline	Interrupt Number	DMAC Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
TAUD2*2						
INTTAUD2I0	Channel 0 interrupt	130	61	15	61	15
INTTAUD2I1	Channel 1 interrupt	131	62	16	62	16
INTTAUD2I2	Channel 2 interrupt	132	63	17	63	17
INTTAUD2I3	Channel 3 interrupt	133	64	18	64	18
INTTAUD2I4	Channel 4 interrupt	134	65	51	65	51
INTTAUD2I5	Channel 5 interrupt	135	66	52	66	52
INTTAUD2I6	Channel 6 interrupt	136	—	25, 73	—	25, 73
INTTAUD2I7	Channel 7 interrupt	137	—	26, 111	—	26, 111
INTTAUD2I8	Channel 8 interrupt	138	—	27, 112	—	27, 112
INTTAUD2I9	Channel 9 interrupt	139	—	28, 113	—	28, 113
INTTAUD2I10	Channel 10 interrupt	140	—	29, 114	—	29, 114
INTTAUD2I11	Channel 11 interrupt	141	—	30, 115	—	30, 115
INTTAUD2I12	Channel 12 interrupt	142	—	31, 116	—	31, 116
INTTAUD2I13	Channel 13 interrupt	143	—	32, 117	—	32, 117
INTTAUD2I14	Channel 14 interrupt	144	—	33, 118	—	33, 118
INTTAUD2I15	Channel 15 interrupt	145	—	34, 119	—	34, 119
TAUD3*2						
INTTAUD3I0	Channel 0 interrupt	146	—	—	—	—
INTTAUD3I1	Channel 1 interrupt	147	—	—	—	—
INTTAUD3I2	Channel 2 interrupt	148	—	—	—	—
INTTAUD3I3	Channel 3 interrupt	149	—	—	—	—
INTTAUD3I4	Channel 4 interrupt	150	—	—	—	—
INTTAUD3I5	Channel 5 interrupt	151	—	—	—	—
INTTAUD3I6	Channel 6 interrupt	152	—	—	—	—
INTTAUD3I7	Channel 7 interrupt	153	—	—	—	—
INTTAUD3I8	Channel 8 interrupt	154	—	—	—	—
INTTAUD3I9	Channel 9 interrupt	155	—	—	—	—
INTTAUD3I10	Channel 10 interrupt	156	—	—	—	—
INTTAUD3I11	Channel 11 interrupt	157	—	—	—	—
INTTAUD3I12	Channel 12 interrupt	158	—	—	—	—
INTTAUD3I13	Channel 13 interrupt	159	—	—	—	—
INTTAUD3I14	Channel 14 interrupt	160	—	—	—	—
INTTAUD3I15	Channel 15 interrupt	161	—	—	—	—

—: No number is assigned.

Note 1. 1st: Primary channel, 2nd: Secondary channel

Note 2. RH850/C1M-A1 does not have TAUD2 and TAUD3.

18.1.5 Reset Sources

TAUD reset sources are listed in the following table. TAUD is initialized by these reset sources.

Table 18.7 Reset Sources

Unit Name	Reset Source
TAUDn	Reset by any reset source

18.1.6 External Input/Output Signals

External input/output signals of TAUD are listed in the following table.

Table 18.8 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal Name
TAUD0		
TAUDTTINm	Channel m input	TAUD0Im
TAUDTTOUTm	Channel m output	TAUD0Om
TAUD1		
TAUDTTINm	Channel m input	TAUD1Im
TAUDTTOUTm	Channel m output	TAUD1Om
TAUD2*1		
TAUDTTINm	Channel m input	TAUD2Im
TAUDTTOUTm	Channel m output	TAUD2Om
TAUD3*1		
TAUDTTINm	Channel m input	TAUD3Im
TAUDTTOUTm	Channel m output	TAUD3Om

Note 1. RH850/C1M-A1 does not have TAUD2 and TAUD3.

CAUTION

When channel input pins are to be used, noise filters must be set for the corresponding port pin functions.

18.2 Overview

18.2.1 Functional Overview

The TAUD has the following functions:

- 16 channels
- 16-bit counter and 16-bit data register per channel
- Independent channel operation
- Synchronous channel operation (master and slave operation)
- Generation of different types of output signal
- Real-time output
- Counter can be triggered by external signal
- Interrupt generation

The Timer Array Unit D is used to perform various count or timer operations and to output a signal which depends on the result of the operation. It contains one prescaler block for count clock generation and 16 channels, each equipped with a 16 bit counter TAUDnCNTm and a 16-bit data register TAUDnCDRm to hold the start or compare value of the counter.

It also contains several control and status registers.

Independent and synchronous operation

Every channel can operate in different operation modes, either independently or in combination with other channels (synchronously), i.e. multiple channels depend on each other with one master and one or more slave channels.

When a channel is operated independently, its operation mode and functions are not affected by those of other channels. When a channel is operated synchronously it is either a master or a slave. A master channel can have multiple slaves, and the state of one channel affects that of the other channels. For example, this means that one channel can control when another starts to count, is reset, etc.

CAUTION

The timing chart described in this section shows an operating timing image. A delay time is added to a timer input. For details, see **Section 18.4.8, TAUDTTINm Edge Detection**.

18.2.2 Terms

In this section, the following terms are used:

Independent / synchronous channel operation

Independent or synchronous channel operation describes the dependency of channels on each other:

- If a channel operates independent of all other channels, this is called independent channel operation.
- If a channel operates depending on other channels, this is called synchronous channel operation.

Channel group

In synchronous channel operation, all channels that depend on each other are referred to as a “channel group”.

A channel group has one master channel and one or more slave channels.

Operation mode

An operation mode can be selected for every channel m. The operation mode defines the basic operation and features of a channel.

In synchronous channel operation, every channel in the channel group can operate in a different operation mode.

Examples are “Capture Mode”, “Event Count Mode”, and “Interval Timer Mode”.

Channel output mode

The channel output mode defines the operation of TAUDTTOUTm

- of a single channel (independent output operation) or
- of all channels in a channel group (synchronous output operation).

Examples are “Independent Channel Output Mode 1” and “Synchronous Channel Output Mode 2 with Dead Time Output”.

Channel operation function

The channel operation function defines the complete function and all features

- of a single channel (independent channel operation) or
- of all channels in a channel group (synchronous channel operation).

Upper/lower channel

Depending on the channel number m, a channel with a smaller number or with a larger number is referred to as “upper” or “lower” channel, respectively.

- Upper channel: Channel with a smaller channel number
- Lower channel: Channel with a larger channel number

Example:

For channel 5, channel 3 is an upper channel and channel 9 is a lower channel.

18.2.3 Functional List of Timer Operations

This timer provides the following functions by operating individual channels independently or a combination of channels.

Table 18.9 Functional List of TAUD Operations

Operation function	Example
Independent Channel Operation Functions	**Section 18.4.9**
Interval Timer Function	**Section 18.4.9.1**
TAUDTTINm Input Interval Timer Function	**Section 18.4.9.2**
Clock Divide Function	**Section 18.4.9.3**
External Event Count Function	**Section 18.4.9.4**
Delay Count Function	**Section 18.4.9.5**
One-Pulse Output Function	**Section 18.4.9.6**
TAUDTTINm Input Pulse Interval Measurement Function	**Section 18.4.9.7**
TAUDTTINm Input Signal Width Measurement Function	**Section 18.4.9.8**
TAUDTTINm Input Position Detection Function	**Section 18.4.9.9**
TAUDTTINm Input Period Count Detection Function	**Section 18.4.9.10**
TAUDTTINm Input Pulse Interval Judgment Function	**Section 18.4.9.11**
TAUDTTINm Input Signal Width Judgment Function	**Section 18.4.9.12**
Independent Channel Real-Time Functions	**Section 18.4.10**
Real-Time Output Function Type 1	**Section 18.4.10.1**
Real-Time Output Function Type 2	**Section 18.4.10.2**
Independent Channel Simultaneous Rewrite Functions	**Section 18.4.11**
Simultaneous Rewrite Trigger Generation Function Type 1	**Section 18.4.11.1**
Synchronous Channel Operation Functions	**Section 18.4.12**
PWM Output Function	**Section 18.4.12.1**
One-Shot Pulse Output Function	**Section 18.4.12.2**
Trigger Start PWM Output Function	**Section 18.4.12.3**
Delay Pulse Output Function	**Section 18.4.12.4**
Offset Trigger Output Function	**Section 18.4.12.5**
A/D Conversion Trigger Output Function Type 1	**Section 18.4.12.6**
Triangle PWM Output Function	**Section 18.4.12.7**
Triangle PWM Output Function with Dead Time	**Section 18.4.12.8**
A/D Conversion Trigger Output Function Type 2	**Section 18.4.12.9**
Interrupt Request Signals Culling Function	**Section 18.4.12.10**
One-phase PWM output function	**Section 18.4.12.11**
Synchronous Non-Complementary and Complementary Modulation Output Functions	**Section 18.4.13**
Non-Complementary Modulation Output Function Type 1	**Section 18.4.13.1**
Non-Complementary Modulation Output Function Type 2	**Section 18.4.13.2**
Complementary Modulation Output Function	**Section 18.4.13.3**

18.2.4 TAUD I/O and Interrupt Request Signals

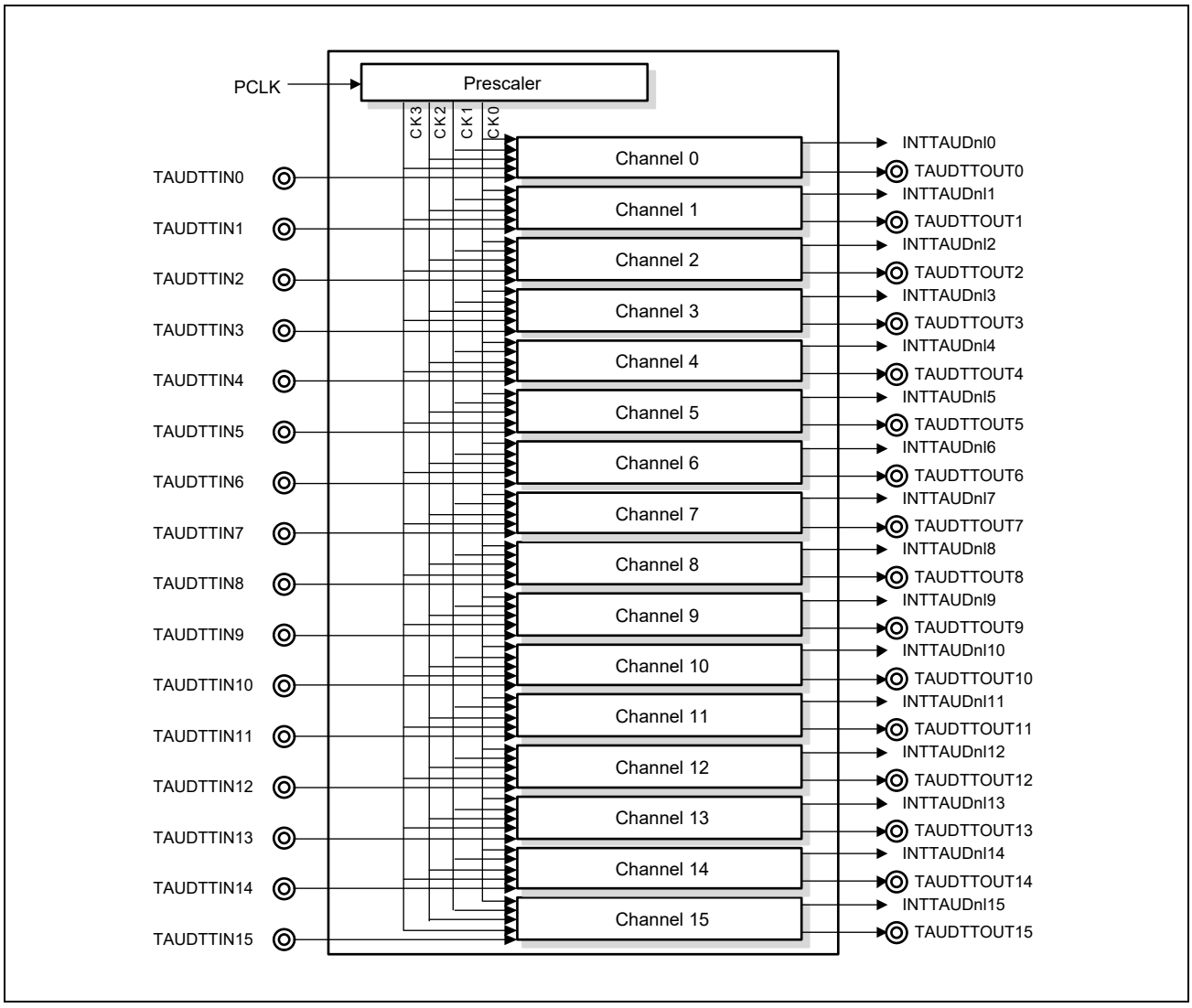


Figure 18.1 TAUD I/O and Interrupt Request Signals

18.2.5 Block Diagram

Figure 18.2 shows the main components of the TAUD:

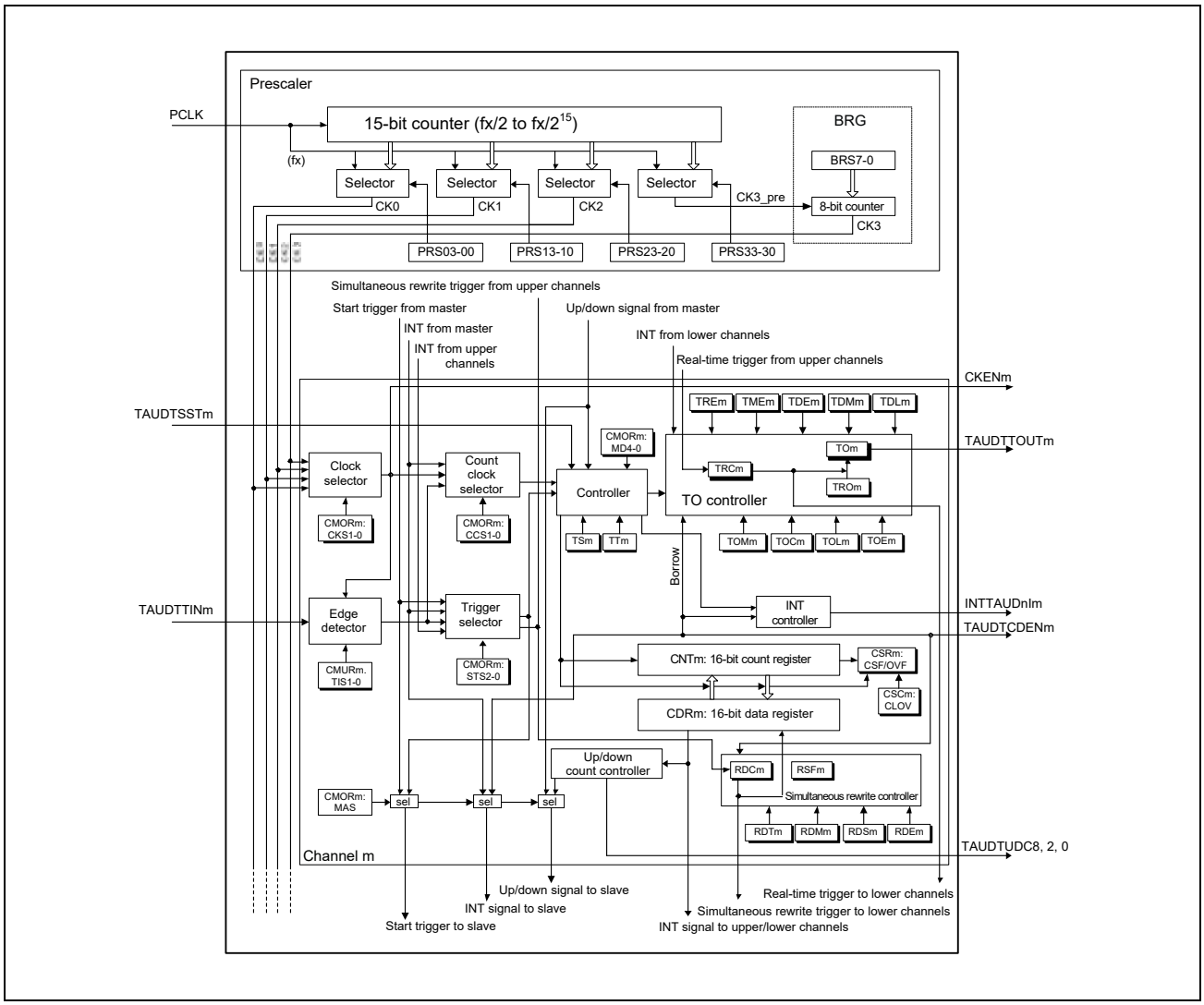


Figure 18.2 Block Diagram of the TAUD

The prefix “TAUDn” has been omitted from the register names for the sake of clarity in the above figure.

- TAUDTSSTm: Simultaneous start trigger (input from PIC1B)

18.2.6 Description of Blocks

The following describes the functional blocks:

Prescaler block

The prescaler block provides up to 4 clock signals (CK0 to CK3) that can be used as count clocks for all channels.

Count clocks CK0 to CK2 are derived from PCLK by a configurable prescaler division factor of 2^0 to 2^{15}. The fourth count clock CK3 can be adjusted more precisely by an additional division factor that is not a power of 2.

Clock and count clock selection

For every channel, the count clock selector selects which of the following is used as the clock source:

- One of the clocks CK0 to CK3 (selected by the clock selector)
- INTTAUDnIm from master channel
- TAUDTTINm input signal effective edge

Controller

The controller controls the main operations of the counter:

- Operation mode (selected by bits TAUDnCMORm.TAUDnMD[4:0])
- Counter start enable (TAUDnTS.TAUDnTSm) and counter stop (TAUDnTT.TAUDnTTm) When counter start is enabled, status flag TAUDnTE.TAUDnTEm is set.
- Count direction (up/down) (can be controlled by master channel)

Trigger selector

Depending on the selected operation mode, the counter starts automatically when it is enabled

(TAUDnTE.TAUDnTEm = 1), or it waits for an external start trigger signal. Any of the following signals can be used as the start trigger:

- Synchronous channel start trigger input TAUDTSSTm

  For details on how to make a simultaneous start between the units, see **Section 24.2.3.1, Simultaneous Start Trigger Function**.
- TAUDTTINm input effective edge
- INTTAUDnIm from the master or any upper channel
- Up/down output trigger signal of the master channel
- Dead-time output signal of the TAUDTTOUTm generation unit.

Simultaneous rewrite controller

Simultaneous rewrite control is a special function that can be used in synchronous operation modes. The data registers (TAUDnCDRm) of all channels in a channel group can be rewritten at any time. The simultaneous rewrite controller ensures that new data register values of all channels become effective at the same time.

TAUDnTO controller

The output control of every channel enables the generation of various output signal forms such as PWM signals or triangular waves.

18.3 Registers

18.3.1 List of Registers

TAUD registers are listed in the following table.

See **Section 18.1.2** for <TAUDn_base>.

Table 18.10 TAUDn Registers Overview

Module Name	Register Name	Symbol	Address
TAUDn prescaler registers			
TAUDn	TAUD prescaler clock select register	TAUDnTPS	<TAUDn_base> + 240_H
TAUDn	TAUDn prescaler baud rate setting register	TAUDnBRS	<TAUDn_base> + 244_H
TAUDn control registers			
TAUDn	TAUDn channel data register m	TAUDnCDRm	<TAUDn_base> + m × 4_H
TAUDn	TAUDn channel counter register m	TAUDnCNTm	<TAUDn_base> + 80_H + m × 4_H
TAUDn	TAUDn channel mode OS register m	TAUDnCMORm	<TAUDn_base> + 200_H + m × 4_H
TAUDn	TAUDn channel mode user register m	TAUDnCMURm	<TAUDn_base> + $C0_H$ + m × 4_H
TAUDn	TAUDn channel status register m	TAUDnCSRm	<TAUDn_base> + 140_H + m × 4_H
TAUDn	TAUDn channel status clear register m	TAUDnCSCm	<TAUDn_base> + 180_H + m × 4_H
TAUDn	TAUDn channel start trigger register	TAUDnTS	<TAUDn_base> + $1C4_H$
TAUDn	TAUDn channel enable status register	TAUDnTE	<TAUDn_base> + $1C0_H$
TAUDn	TAUDn channel stop trigger register	TAUDnTT	<TAUDn_base> + $1C8_H$
TAUDn output registers			
TAUDn	TAUDn channel output enable register	TAUDnTOE	<TAUDn_base> + $5C_H$
TAUDn	TAUDn channel output register	TAUDnTO	<TAUDn_base> + 58_H
TAUDn	TAUDn channel output mode register	TAUDnTOM	<TAUDn_base> + 248_H
TAUDn	TAUDn channel output configuration register	TAUDnTOC	<TAUDn_base> + $24C_H$
TAUDn	TAUDn channel output active level register	TAUDnTOL	<TAUDn_base> + 40_H
TAUDn	TAUDn channel dead time output enable register	TAUDnTDE	<TAUDn_base> + 250_H
TAUDn	TAUDn channel dead time output mode register	TAUDnTDM	<TAUDn_base> + 254_H
TAUDn	TAUDn channel dead time output level register	TAUDnTDL	<TAUDn_base> + 54_H
TAUDn	TAUDn channel real-time output register	TAUDnTRO	<TAUDn_base> + $4C_H$
TAUDn	TAUDn channel real-time output enable register	TAUDnTRE	<TAUDn_base> + 258_H
TAUDn	TAUDn channel real-time output control register	TAUDnTRC	<TAUDn_base> + $25C_H$
TAUDn	TAUDn channel modulation output enable register	TAUDnTME	<TAUDn_base> + 50_H
TAUDn reload registers			
TAUDn	TAUDn channel reload data enable register	TAUDnRDE	<TAUDn_base> + 260_H
TAUDn	TAUDn channel reload data mode register	TAUDnRDM	<TAUDn_base> + 264_H
TAUDn	TAUDn channel reload data control CH select register	TAUDnRDS	<TAUDn_base> + 268_H
TAUDn	TAUDn channel reload data control register	TAUDnRDC	<TAUDn_base> + $26C_H$
TAUDn	TAUDn channel reload data trigger register	TAUDnRDT	<TAUDn_base> + 44_H
TAUDn	TAUDn channel reload status register	TAUDnRSF	<TAUDn_base> + 48_H

18.3.2 TAUDnTPS — TAUDn Prescaler Clock Select Register

This register specifies clocks CK0, CK1, CK2, and CK3_PRE for all channels of the PCLK prescaler. CK3 is generated by dividing CK3_PRE by the factor specified in TAUDnBRS.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 240_H

Value after reset: $FFFF_H$ This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnPRS3[3:0]				TAUDnPRS2[3:0]				TAUDnPRS1[3:0]				TAUDnPRS0[3:0]			
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.11 TAUDnTPS Register Contents (1/3)

Bit Position	Bit Name	Function
15 to 12	TAUDnPRS3[3:0]	These bits specify CK3_PRE clock. CK3_PRE clock is an input clock to BRG unit which supplies the CK3 operation clock to all channels.

TAUDnPRS3[3:0]	CK3_PRE Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK3 are stopped (TAUDnTE.TAUDnTEm = 0).

Table 18.11 TAUDnTPS Register Contents (2/3)

Bit Position	Bit Name	Function
11 to 8	TAUDnPRS2[3:0]	These bits specify the CK2 clock.

TAUDnPRS2[3:0]	CK2 Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK2 are stopped (TAUDnTE.TAUDnTEm = 0).

7 to 4	TAUDnPRS1[3:0]	These bits specify the CK1 clock.

TAUDnPRS1[3:0]	CK1 Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK1 are stopped (TAUDnTE.TAUDnTEm = 0).

Table 18.11 TAUDnTPS Register Contents (3/3)

Bit Position	Bit Name	Function
3 to 0	TAUDnPRS0[3:0]	These bits specify the CK0 clock.

TAUDnPRS0[3:0]	CK0 Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK0 are stopped (TAUDnTE.TAUDnTEm = 0).

NOTE

The TAUDn clock input PCLK is specified in the first part of this section, **Section 18.1.3, Clock Supply**.

18.3.3 TAUDnBRS — TAUDn Prescaler Baud Rate Setting Register

This register specifies the division factor of prescaler clock CK3.

CK3 is generated by dividing CK3_PRE by the factor specified in this register plus one. The PCLK prescaler for CK3_PRE is specified in TAUDnTPS.TAUDnPRS3[3:0].

Access: Readable/writable in 8-bit units.

Address: <TAUDn_base> + 244_H

Value after reset: 00_H This register is initialized by any reset source.

Bit	7	6	5	4	3	2	1	0
	TAUDnBRS[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.12 TAUDnBRS Register Contents

Bit Position	Bit Name	Function
7 to 0	TAUDnBRS[7:0]	These bits specify a CK3_PRE clock division factor for generating CK3.

TAUDnBRS[7:0]	CK3 Clock
$0000\ 0000_B$	CK3_PRE/1
$0000\ 0001_B$	CK3_PRE/2
$0000\ 0010_B$	CK3_PRE/3
$0000\ 0011_B$	CK3_PRE/4
:	:
$1111\ 1110_B$	CK3_PRE/255
$1111\ 1111_B$	CK3_PRE/256

18.3.4 TAUDnCDRm — TAUDn Channel Data Register

This register functions either as a compare register or as a capture register, depending on the operating mode specified in TAUDnCMORm.TAUDnMD[4:1].

Access: Readable/writable in 16-bit units.
Readable in capture mode. Any write operation is ignored.
Readable/writable in compare mode.

Address: <TAUDn_base> + 0_H + m × 4_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCDR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.13 TAUDnCDRm Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnCDR[15:0]	Data register for capture/compare values

18.3.5 TAUDnCNTm — TAUDn Channel Counter Register

This is a channel m counter register.

Access: Readable in 16-bit units.

Address: <TAUDn_base> + 80_H + m × 4_H

Value after reset: $FFFF_H$ The value after reset depends on an operating mode. See **Table 18.15, TAUDnCNTm Read Values after Re-Enabling Counter**. This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCNT[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.14 TAUDnCNTm Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnCNT[15:0]	16-bit counter value

A read value depends on a counter value, a changed operating mode, TAUDnTS.TAUDnTSm or TAUDnTT.TAUDnTTm bit value.

The initial read value of the counter depends on an operating mode and how the counter is stopped.

- Stop by a reset
- Stop by a counter stop trigger (TAUDnTT.TAUDnTTm = 1)

Table 18.15 lists the initial counter read values after the counter is stopped (TAUDnTE.TAUDnTEm = 0) and re-enabled (TAUDnTS.TAUDnTSm = 1).

The table also contains the counter read value one count after the counter is enabled (TAUDnTS.TAUDnTSm = 1) with the counter waiting for a start trigger.

Table 18.15 TAUDnCNTm Read Values after Re-Enabling Counter

Mode Name	Count Method (Up/Down)	TAUDnCNTm Value		
		After Reset	After Stop Trigger	After One Count
Interval timer mode	Count down	$FFFF_H$	Stop value	—
Judge mode	Count down	$FFFF_H$	Stop value	—
Capture mode	Count up	0000_H	Stop value	—
Event count mode	Count down	$FFFF_H$	Stop value	—
One-count mode	Count down	$FFFF_H$	Stop value	Stop value
Capture and one-count mode	Count up	0000_H	Stop value	Capture value + 1 (TAUDnCDRm)
Judge and one-count mode	Count down	$FFFF_H$	Stop value	TAUDnCNTm value – 1
Up/down count mode	Count down/up	$FFFF_H$	Stop value	—
Pulse one count mode	Count down	$FFFF_H$	Stop value	0000_H
Count capture mode	Count up	0000_H	Stop value	—
Capture and gate count mode	Count up	0000_H	Stop value	Stop value

NOTE

If the operating mode is changed while the counter is stopped, the initial counter value after a counter restart becomes undefined. The operating mode is changed by the TAUDnCMORm.TAUDnMD[4:1] register.

18.3.6 TAUDnCMORm — TAUDn Channel Mode OS Register

This register controls channel m operation.

Access: Readable/writable in 16-bit units. Writable when the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + 200_H + m × 4_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.16 TAUDnCMORm Register Contents (1/3)

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select an operation clock. An operation clock is used for the TAUDTTINm input edge detection circuit. TAUDnCMORm.TAUDnCCS[1:0] bit setting enables use as a counter clock.

TAUDn CKS1	TAUDn CKS0	Selection of Operation Clock
0	0	CK0
0	1	CK1
1	0	CK2
1	1	CK3

Bit Position	Bit Name	Function
13, 12	TAUDnCCS[1:0]	These bits select a count clock for TAUDnCNTm counter.

TAUDn CCS1	TAUDn CCS0	Selection of Operation Clock
0	0	Operation clock specified by TAUDnCMORm.TAUDnCKS[1:0]
0	1	Effective edge of TAUDTTINm input signal
1	0	Setting prohibited
1	1	INTTAUDnIm signal of master channel

Bit Position	Bit Name	Function
11	TAUDnMAS	This bit specifies whether the channel is a master channel or slave channel during synchronous channel operation. 0: Slave 1: Master This bit setting is valid only for even-numbered channels (CHm_even). Odd-numbered channels (CHm_odd) are fixed to 0.

Table 18.16 TAUDnCMORm Register Contents (2/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>10 to 8</td><td>TAUDnSTS[2:0]</td><td>These bits select an external start trigger.
<table>
<tr><th>TAUDn STS2</th><th>TAUDn STS1</th><th>TAUDn STS0</th><th>Functional description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Software trigger</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Effective edge of TAUDTTINm input signal, which is specified by TAUDnCMURm.TAUDnTIS[1:0]</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>The effective edge of TAUDTTINm input signal is used as a start trigger and the opposite edge as a stop trigger</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Triggers simultaneous rewriting</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>INTTAUDnIm of the master channel is the starting trigger</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>INTTAUDnIm of the upper channel (m–1) is the start trigger regardless of the master setting</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>Dead time output signal of TAUDTTOUTm generating unit</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Up/down output trigger signal of the master channel</td></tr>
</table></td></tr>
<tr><td>7, 6</td><td>TAUDnCOS[1:0]</td><td>These bits specify the timing for updating capture register TAUDnCDRm and overflow flag TAUDnCSRm.TAUDnOVF of channel m.
These bits are valid only when channel m is in capture mode.
<table>
<tr><th>TAUDn COS1</th><th>TAUDn COS0</th><th>TAUDnCDRm</th><th>TAUDnCSRm.TAUDnOVF</th></tr>
<tr><td>0</td><td>0</td><td rowspan="2">Updated upon detection of effective edge of TAUDTTINm input.</td><td>Updated (cleared or set) by detecting effective edge of TAUDTTINm input:
• If a counter overflow has occurred since the last detection of effective edge, set TAUDnCSRm.TAUDnOVF.
• If no counter overflow has occurred since the last detection of effective edge, clear TAUDnCSRm.TAUDnOVF.</td></tr>
<tr><td>0</td><td>1</td><td>Set when a counter overflow occurs, and cleared when TAUDnCSCm.TAUDnCLOV is set to 1.</td></tr>
<tr><td>1</td><td>0</td><td rowspan="2">Updated upon detection of effective edge of TAUDTTINm input and at the occurrence of counter overflow:
• Detection of an effective TAUDTTINm input edge: Counter value is written into TAUDnCDRm.
• Occurrence of overflow: $FFFF_H$ is loaded into TAUDnCDRm. The next detection of an effective TAUDTTINm input edge is ignored.</td><td>Not set</td></tr>
<tr><td>1</td><td>1</td><td>Set when a counter overflow occurs, and cleared when TAUDnCSCm.TAUDnCLOV is set to 1.</td></tr>
</table></td></tr>
</table>

Table 18.16 TAUDnCMORm Register Contents (3/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>5</td><td>Reserved</td><td>When read, the value after reset is read. When writing, write the value after reset.</td></tr>
<tr><td>4 to 0</td><td>TAUDnMD[4:0]</td><td>These bits specify an operating mode.
<table>
<tr><th>TAUDn MD4</th><th>TAUDn MD3</th><th>TAUDn MD2</th><th>TAUDn MD1</th><th>TAUDn MD0</th><th>Functional Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1/0</td><td>Interval timer mode</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1/0</td><td>Judge mode</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>1/0</td><td>Capture mode</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>Event count mode</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1/0</td><td>One-count mode</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Capture and one-count mode</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>1/0</td><td>Judge and one-count mode</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>Up/down count mode</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>1/0</td><td>Pulse one-count mode</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1/0</td><td>Count capture mode</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Capture and gate count mode</td></tr>
</table>
Settings other than the above are prohibited.
<table>
<tr><th>Mode</th><th>Role of TAUDnMD0 Bit</th></tr>
<tr><td>Interval timer mode
Capture mode
Count capture mode</td><td>Specifies whether INTTAUDnIm is generated at the beginning of count operation (when a start trigger is entered) or not.
0: INTTAUDnIm is not generated.
1: INTTAUDnIm is generated.</td></tr>
<tr><td>Event count mode
Up/down count mode</td><td>This bit should be set to 0 (INTTAUDnIm signal is not output at the beginning of count operation).</td></tr>
<tr><td>One-count mode
Pulse one-count mode</td><td>Enables/disables start trigger detection during counting.
0: Disables detection.
1: Enables detection.</td></tr>
<tr><td>Capture and one-count mode
Capture and gate count mode</td><td>This bit should be set to 0.

CAUTION: INTTAUDnIm signal is not output at the beginning of count operation. In addition, start trigger detected during counting is disabled.</td></tr>
<tr><td>Judge mode
Judge and one-count mode</td><td>Specifies INTTAUDnIm output timing.
0: When TAUDnCNTm ≤ TAUDnCDRm
1: When TAUDnCNTm > TAUDnCDRm</td></tr>
</table>
</td></tr>
</table>

18.3.7 TAUDnCMURm — TAUDn Channel Mode User Register

This register specifies a type of effective edge detection used for TAUDTTINm input.

Access: Readable/writable in 8-bit units.

Address: <TAUDn_base> + $C0_H$ + m × 4_H

Value after reset: 00_H This register is initialized by any reset source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.17 TAUDnCMURm Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	These bits specify an effective edge of TAUDTTINm input signal.

TAUDn TIS1	TAUDn TIS0	Functional Description
0	0	Falling edge
0	1	Rising edge
1	0	Detection of rising and falling edges (selects low width measurement) • Start trigger: Falling edge • Stop trigger (capture): Rising edge
1	1	Detection of rising and falling edges (selects high width measurement) • Start trigger: Rising edge • Stop trigger (capture): Falling edge

Edge detection of TAUDTTINm input signal is based on the operation clock selected by TAUDnCMORm.TAUDnCKS[1:0].

18.3.8 TAUDnCSRm — TAUDn Channel Status Register

This register indicates the count direction and overflow status of channel m counter.

Access: Readable in 8-bit units.

Address: <TAUDn_base> + 140_H + m × 4_H

Value after reset: 00_H This register is initialized by any reset source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnCSF	TAUDnOVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 18.18 TAUDnCSRm Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read.
1	TAUDnCSF	Indicates a count direction. 0: Count-up 1: Count-down The read value of this bit is valid only in the following mode: • Up/down count mode
0	TAUDnOVF	Indicates counter overflow status. 0: No overflow occurs. 1: Overflow occurs. This bit is used only in the following modes: • Capture mode • Capture and one-count mode The function of this bit depends on the setting of control bit TAUDnCMORm.TAUDnCOS[1:0].

18.3.9 TAUDnCSCm — TAUDn Channel Status Clear Register

This is a trigger register for clearing the overflow flag TAUDnCSRm.TAUDnOVF of channel m.

Access: Writable in 8-bit units. This value is always read as 00_H.

Address: <TAUDn_base> + 180_H + m × 4_H

Value after reset: 00_H This register is initialized by any reset source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAUDnCLOV
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 18.19 TAUDnCSCm Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TAUDnCLOV	0: No function 1: Clears overflow flag TAUDnCSRm.TAUDnOVF.

18.3.10 TAUDnTS — TAUDn Channel Start Trigger Register

This register enables the counter operation of each channel.

Access: Writable in 16-bit units. This value is always read as 0000_H.

Address: <TAUDn_base> + $1C4_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTS15	TAUDnTS14	TAUDnTS13	TAUDnTS12	TAUDnTS11	TAUDnTS10	TAUDnTS09	TAUDnTS08	TAUDnTS07	TAUDnTS06	TAUDnTS05	TAUDnTS04	TAUDnTS03	TAUDnTS02	TAUDnTS01	TAUDnTS00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 18.20 TAUDnTS Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTSm	These bits enable the counter operation of channel m. 0: No function 1: Enables the counter operation and sets TAUDnTE.TAUDnTEm to 1. The counter operation is only enabled when TAUDnTE.TAUDnTEm is set to 1. Whether counting is started or not depends on a selected operating mode.

18.3.11 TAUDnTE — TAUDn Channel Enable Status Register

This register enables/disables a counter operation.

Access: Readable in 16-bit units.

Address: <TAUDn_base> + $1C0_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTE15	TAUDnTE14	TAUDnTE13	TAUDnTE12	TAUDnTE11	TAUDnTE10	TAUDnTE09	TAUDnTE08	TAUDnTE07	TAUDnTE06	TAUDnTE05	TAUDnTE04	TAUDnTE03	TAUDnTE02	TAUDnTE01	TAUDnTE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.21 TAUDnTE Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTEm	These bits enable or disable the counter operation of channel m. 0: Disables counter operation. 1: Enables counter operation. This bit is set to 1 when trigger input of TAUDTSSTm (synchronous channel start trigger signal) is detected or when TAUDnTS.TAUDnTSm is set to 1. This bit is set to 0 when TAUDnTT.TAUDnTTm is set to 1.

NOTE

For details on how to make a simultaneous start between the units, see **Section 24.2.3.1, Simultaneous Start Trigger Function.**

18.3.12 TAUDnTT — TAUDn Channel Stop Trigger Register

This register stops the counter operation of each channel.

Access: Writable in 16-bit units. This value is always read as 0000_H.

Address: <TAUDn_base> + $1C8_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTT15	TAUDnTT14	TAUDnTT13	TAUDnTT12	TAUDnTT11	TAUDnTT10	TAUDnTT09	TAUDnTT08	TAUDnTT07	TAUDnTT06	TAUDnTT05	TAUDnTT04	TAUDnTT03	TAUDnTT02	TAUDnTT01	TAUDnTT00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 18.22 TAUDnTT Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTTm	These bits are used to stop the counter operation of channel m. 0: No function 1: Stops the counter operation and resets TAUDnTE.TAUDnTEm. TAUDnCNTm, TAUDnTO.TAUDnTOm, and TAUDTTOUTm retain the values provided before the counter is stopped.

18.3.13 TAUDnRDE — TAUDn Channel Reload Data Enable Register

This register enables/disables simultaneous rewrite of TAUDnCDRm/TAUDnTOLm data register.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + 260_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RDE15	TAUDn RDE14	TAUDn RDE13	TAUDn RDE12	TAUDn RDE11	TAUDn RDE10	TAUDn RDE09	TAUDn RDE08	TAUDn RDE07	TAUDn RDE06	TAUDn RDE05	TAUDn RDE04	TAUDn RDE03	TAUDn RDE02	TAUDn RDE01	TAUDn RDE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.23 TAUDnRDE Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnRDEm	These bits enable or disable simultaneous rewrite of the data register of channel m. 0: Disables simultaneous rewrite 1: Enables simultaneous rewrite

18.3.14 TAUDnRDS — TAUDn Channel Reload Data Control Channel Select Register

This register selects a channel that controls simultaneous rewrite.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + 268_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RDS15	TAUDn RDS14	TAUDn RDS13	TAUDn RDS12	TAUDn RDS11	TAUDn RDS10	TAUDn RDS09	TAUDn RDS08	TAUDn RDS07	TAUDn RDS06	TAUDn RDS05	TAUDn RDS04	TAUDn RDS03	TAUDn RDS02	TAUDn RDS01	TAUDn RDS00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.24 TAUDnRDS Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnRDSm	These bits select a channel that controls a simultaneous rewrite trigger. 0: Master channel 1: Another upper channel

18.3.15 TAUDnRDM — TAUDn Channel Reload Data Mode Register

This register selects the timing for generating a simultaneous rewrite control signal.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + 264_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RDM15	TAUDn RDM14	TAUDn RDM13	TAUDn RDM12	TAUDn RDM11	TAUDn RDM10	TAUDn RDM09	TAUDn RDM08	TAUDn RDM07	TAUDn RDM06	TAUDn RDM05	TAUDn RDM04	TAUDn RDM03	TAUDn RDM02	TAUDn RDM01	TAUDn RDM00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.25 TAUDnRDM Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnRDMm	These bits select the timing for generating a simultaneous rewrite trigger signal. 0: When the master channel counter starts to count 1: At the peak of cycle of triangular wave These bit settings are applied only when TAUDnRDE.TAUDnRDEm = 1 and TAUDnRDS.TAUDnRDSm = 0.

18.3.16 TAUDnRDC — TAUDn Channel Reload Data Control Register

This register specifies a channel which generates an INTTAUDnlm signal to trigger simultaneous rewrite.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + $26C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RDC15	TAUDn RDC14	TAUDn RDC13	TAUDn RDC12	TAUDn RDC11	TAUDn RDC10	TAUDn RDC09	TAUDn RDC08	TAUDn RDC07	TAUDn RDC06	TAUDn RDC05	TAUDn RDC04	TAUDn RDC03	TAUDn RDC02	TAUDn RDC01	TAUDn RDC00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.26 TAUDnRDC Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnRDCm	These bits specify whether the channel generates a simultaneous rewrite trigger signal or not. 0: Not operate as a simultaneous rewrite trigger channel. 1: Operates as a simultaneous rewrite trigger channel. These bit settings are applied only when TAUDnRDE.TAUDnRDEm = 1 and TAUDnRDS.TAUDnRDSm = 1.

18.3.17 TAUDnRDT — TAUDn Channel Reload Data Trigger Register

This register triggers a simultaneous rewrite enabling state.

Access: Writable in 16-bit units. This value is always read as 0000_H.

Address: <TAUDn_base> + 44_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RDT15	TAUDn RDT14	TAUDn RDT13	TAUDn RDT12	TAUDn RDT11	TAUDn RDT10	TAUDn RDT09	TAUDn RDT08	TAUDn RDT07	TAUDn RDT06	TAUDn RDT05	TAUDn RDT04	TAUDn RDT03	TAUDn RDT02	TAUDn RDT01	TAUDn RDT00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 18.27 TAUDnRDT Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnRDTm	These bits are used to trigger a simultaneous rewrite enabling state. 0: No function 1: Triggers a simultaneous rewrite enabling state. The simultaneous rewrite pending flag (TAUDnRSFm) is set to 1. The system waits for a simultaneous rewrite trigger.

18.3.18 TAUDnRSF — TAUDn Channel Reload Status Register

This flag register indicates simultaneous rewrite status.

Access: Readable in 16-bit units.

Address: <TAUDn_base> + 48_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RSF15	TAUDn RSF14	TAUDn RSF13	TAUDn RSF12	TAUDn RSF11	TAUDn RSF10	TAUDn RSF09	TAUDn RSF08	TAUDn RSF07	TAUDn RSF06	TAUDn RSF05	TAUDn RSF04	TAUDn RSF03	TAUDn RSF02	TAUDn RSF01	TAUDn RSF00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.28 TAUDnRSF Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnRSFm	These bits indicate simultaneous rewrite status. 0: Indicates that a simultaneous rewrite trigger has started simultaneous rewrite. 1: Indicates that simultaneous rewrite is in the enabling state (TAUDnRDTm = 1) and that the system waits for a simultaneous rewrite trigger.

18.3.19 TAUDnTOE — TAUDn Channel Output Enable Register

This register enables/disables the independent channel output mode controlled by software.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + $5C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTOE15	TAUDnTOE14	TAUDnTOE13	TAUDnTOE12	TAUDnTOE11	TAUDnTOE10	TAUDnTOE09	TAUDnTOE08	TAUDnTOE07	TAUDnTOE06	TAUDnTOE05	TAUDnTOE04	TAUDnTOE03	TAUDnTOE02	TAUDnTOE01	TAUDnTOE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.29 TAUDnTOE Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTOEm	These bits enable or disable the independent channel output function. 0: Disables the independent timer output function. 1: Enables the independent timer output function.

18.3.20 TAUDnTO — TAUDn Channel Output Register

This register specifies and reads a TAUDTTOUTm level.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 58_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTO15	TAUDnTO14	TAUDnTO13	TAUDnTO12	TAUDnTO11	TAUDnTO10	TAUDnTO09	TAUDnTO08	TAUDnTO07	TAUDnTO06	TAUDnTO05	TAUDnTO04	TAUDnTO03	TAUDnTO02	TAUDnTO01	TAUDnTO00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.30 TAUDnTO Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTOm	These bits specify and read a TAUDTTOUTm level. 0: Low level 1: High level Only TAUDnTOm bits for which Independent Channel Output function is disabled (TAUDnTOEm = 0) can be written.

18.3.21 TAUDnTOM — TAUDn Channel Output Mode Register

This register specifies the output mode of each channel.

Access: Readable/writable in 16-bit units. Writable only while the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + 248_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTOM15	TAUDnTOM14	TAUDnTOM13	TAUDnTOM12	TAUDnTOM11	TAUDnTOM10	TAUDnTOM09	TAUDnTOM08	TAUDnTOM07	TAUDnTOM06	TAUDnTOM05	TAUDnTOM04	TAUDnTOM03	TAUDnTOM02	TAUDnTOM01	TAUDnTOM00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.31 TAUDnTOM Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTOMm	These bits specify an output mode. 0: Independent channel operation 1: Synchronous channel operation As described in **Section 18.4.4, Channel Output Modes**, the output mode depends on the setting of each channel output control bit.

18.3.22 TAUDnTOC — TAUDn Channel Output Configuration Register

This register specifies the output mode of each channel in combination with TAUDnTOMm.

Access: Readable/writable in 16-bit units. Writable only while the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + $24C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTOC15	TAUDnTOC14	TAUDnTOC13	TAUDnTOC12	TAUDnTOC11	TAUDnTOC10	TAUDnTOC09	TAUDnTOC08	TAUDnTOC07	TAUDnTOC06	TAUDnTOC05	TAUDnTOC04	TAUDnTOC03	TAUDnTOC02	TAUDnTOC01	TAUDnTOC00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.32 TAUDnTOC Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTOCm	These bits specify an output mode. 0: Operating mode 1 1: Operating mode 2 As listed below, the output mode depends on the setting of TAUDnTOM.TAUDnTOMm.

TOMm	TOCm	Functional Description
0	0	Toggle mode: Toggle operation is conducted when INTTAUDnIm occurs.
0	1	Set/reset mode: Set when INTTAUDnIm occurs at the beginning of count operation, and reset when INTTAUDnIm is caused by detection of a match between TAUDnCNTm and TAUDnCDRm.
1	0	Synchronous channel operating mode 1: Set when INT occurs on master channels, and reset when INT occurs on slave channels.
1	1	Synchronous channel operating mode 2: Set when INTTAUDnIm occurs in count-down status, and reset when INTTAUDnIm occurs in count-up status.

18.3.23 TAUDnTOL — TAUDn Channel Output Active Level Register

This register specifies the output logic of channel output bit (TAUDnTO.TAUDnTOm).

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 040_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTOL15	TAUDnTOL14	TAUDnTOL13	TAUDnTOL12	TAUDnTOL11	TAUDnTOL10	TAUDnTOL09	TAUDnTOL08	TAUDnTOL07	TAUDnTOL06	TAUDnTOL05	TAUDnTOL04	TAUDnTOL03	TAUDnTOL02	TAUDnTOL01	TAUDnTOL00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.33 TAUDnTOL Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTOLm	Specifies the output logic of channel m output bit (TAUDnTO.TAUDnTOm). 0: Positive logic (active high) 1: Negative logic (active low) These bits apply in all channel output modes except independent channel output mode controlled by software and independent channel output mode 1.

18.3.24 TAUDnTDE — TAUDn Channel Dead Time Output Enable Register

This register enables/disables the dead time operation of every channel.

Access: Readable/writable in 16-bit units. Writable only while the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + 250_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTDE15	TAUDnTDE14	TAUDnTDE13	TAUDnTDE12	TAUDnTDE11	TAUDnTDE10	TAUDnTDE09	TAUDnTDE08	TAUDnTDE07	TAUDnTDE06	TAUDnTDE05	TAUDnTDE04	TAUDnTDE03	TAUDnTDE02	TAUDnTDE01	TAUDnTDE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.34 TAUDnTDE Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTDEm	These bits enable or disable the dead time control operation of channel m. 0: Disables dead time operation 1: Enables dead time operation. The same setting should be made for both even and odd slave channels in pairs. These bit settings are applied when: • TAUDnTOE.TAUDnTOEm, TAUDnTOM.TAUDnTOMm, TAUDnTOC.TAUDnTOCm = 1

18.3.25 TAUDnTDM — TAUDn Channel Dead Time Output Mode Register

This register specifies the timing to add dead time during dead time output.

Access: Readable/writable in 16-bit units. Writable only while the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + 254_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTDM15	TAUDnTDM14	TAUDnTDM13	TAUDnTDM12	TAUDnTDM11	TAUDnTDM10	TAUDnTDM09	TAUDnTDM08	TAUDnTDM07	TAUDnTDM06	TAUDnTDM05	TAUDnTDM04	TAUDnTDM03	TAUDnTDM02	TAUDnTDM01	TAUDnTDM00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.35 TAUDnTDM Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTDMm	These bits specify the timing to add dead time during dead time output. 0: When detecting the duty cycle of an upper even-numbered channel (duty dead time output). 1: When detecting the TIN input edge of a lower odd-numbered channel (one-phase dead time output). The same setting should be made for both even and odd slave channels in pairs. These bit settings are applied when: • TAUDnTOE.TAUDnTOEm, TAUDnTOM.TAUDnTOMm, TAUDnTOC.TAUDnTOCm, TAUDnTDE.TAUDnTDEm = 1

18.3.26 TAUDnTDL — TAUDn Channel Dead Time Output Level Register

This register selects a phase in which dead time is added.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 54_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTDL15	TAUDnTDL14	TAUDnTDL13	TAUDnTDL12	TAUDnTDL11	TAUDnTDL10	TAUDnTDL09	TAUDnTDL08	TAUDnTDL07	TAUDnTDL06	TAUDnTDL05	TAUDnTDL04	TAUDnTDL03	TAUDnTDL02	TAUDnTDL01	TAUDnTDL00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.36 TAUDnTDL Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTDLm	These bits select a phase in which dead time is added. 0: Normal phase 1: Reverse phase These bit settings are applied when: • TAUDnTOE.TAUDnTOEm, TAUDnTOM.TAUDnTOMm, TAUDnTOC.TAUDnTOCm, TAUDnTDE.TAUDnTDEm = 1

18.3.27 TAUDnTRE — TAUDn Channel Real-time Output Enable Register

This register enables/disables real-time output.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + 258_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTRE15	TAUDnTRE14	TAUDnTRE13	TAUDnTRE12	TAUDnTRE11	TAUDnTRE10	TAUDnTRE09	TAUDnTRE08	TAUDnTRE07	TAUDnTRE06	TAUDnTRE05	TAUDnTRE04	TAUDnTRE03	TAUDnTRE02	TAUDnTRE01	TAUDnTRE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.37 TAUDnTRE Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTREm	These bits enable or disable real-time output of channel m. 0: Disables real-time output 1: Enables real-time output. These bit settings are applied only when TAUDnTOE.TAUDnTOEm = 1.These bit settings are applied only when TAUTnTOE.TAUDnTOEm = 1. When TAUDnTRE.TREm = 0, TAUDTTOUTm is not affected by real-time output. When TAUDnTRE.TREm = 1, TAUDTTOUTm outputs the value of real-time output bit TAUDnTRO.TAUDnTROm in response to a timer operation.

18.3.28 TAUDnTRC — TAUDn Channel Real-time Output Control Register

This register controls the real-time output trigger of each channel.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + $25C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTRC15	TAUDnTRC14	TAUDnTRC13	TAUDnTRC12	TAUDnTRC11	TAUDnTRC10	TAUDnTRC09	TAUDnTRC08	TAUDnTRC07	TAUDnTRC06	TAUDnTRC05	TAUDnTRC04	TAUDnTRC03	TAUDnTRC02	TAUDnTRC01	TAUDnTRC00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.38 TAUDnTRC Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTRCm	These bits specify a channel on which the real-time output trigger for channel m is generated. 0: Next upper channel with this bit set to 1 1: Channel m These bit settings are applied only when TAUTnTRE.TAUDnTREm = 1.

18.3.29 TAUDnTRO — TAUDn Channel Real-time Output Register

This register sets a value which is output to TAUDTTOUTm.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + $4C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn TRO15	TAUDn TRO14	TAUDn TRO13	TAUDn TRO12	TAUDn TRO11	TAUDn TRO10	TAUDn TRO09	TAUDn TRO08	TAUDn TRO07	TAUDn TRO06	TAUDn TRO05	TAUDn TRO04	TAUDn TRO03	TAUDn TRO02	TAUDn TRO01	TAUDn TRO00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.39 TAUDnTRO Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTROm	These bits set a value which is output to TAUDTTOUTm. 0: Low level 1: High level TAUDnTROm value is not output to TAUDTTOUTm when TAUDnTRE.TAUDnTREm = 0, even if a real-time output trigger occurs.

18.3.30 TAUDnTME — TAUDn Channel Modulation Output Enable Register

This register enables/disables modulation output for timer output and real-time output.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 50_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn TME15	TAUDn TME14	TAUDn TME13	TAUDn TME12	TAUDn TME11	TAUDn TME10	TAUDn TME09	TAUDn TME08	TAUDn TME07	TAUDn TME06	TAUDn TME05	TAUDn TME04	TAUDn TME03	TAUDn TME02	TAUDn TME01	TAUDn TME00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.40 TAUDnTME Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnTMEm	These bits enable or disable modulation output for timer output and real-time output of channel m. 0: Disables modulation 1: Enables modulation These bit settings are applied only when TAUDnTOE.TAUDnTOEm and TAUDnTRE.TAUDnTREm = 1.

18.4 Function

18.4.1 General Operating Procedure

The following lists the general operation procedure for the TAUDn:

After reset release, the operation of each channel is stopped. Clock supply is started and writing to each register is enabled. All circuits and registers of all channels are initialized. The control register of TAUDTTOUTm is also initialized and outputs a low level.

(1) Set the TAUDnTPS and TAUDnBRS registers to specify the clock frequency of CK0 to CK3.

(2) Configure the desired TAUDn function:

- Set the operation mode
- Set the channel output mode
- Set any other control bits

(3) Enable the counter by setting the TAUDnTS.TAUDnTSm bit to 1.
The counter starts to count immediately, or when an appropriate trigger is detected, depending on the bit settings.

(4) Stop the counter or perform a forced restart operation during count operation. The counter can be stopped by setting the TAUDnTT.TAUDnTTm bit to 1. The counter can be forcibly restarted by setting the TAUDnTS.TAUDnTSm bit to 1.

(5) Stop the function by setting the TAUDnTT.TAUDnTTm bit to 1.

NOTE

A detailed description of the required control bits and the operation of the individual functions is given below:

- **Section 18.4.9, Independent Channel Operation Functions**
- **Section 18.4.12, Synchronous Channel Operation Functions**

18.4.2 Concepts of Synchronous Channel Operation

The synchronous channel operation function is implemented using a combination of channel groups (consist of master and slave channels). Several rules apply to the settings of channels. These rules are detailed in **Section 18.4.2.1, Rules of Synchronous Channel Operation.**

Two special features for synchronous channel operation are detailed in the following sections:

- **Section 18.4.2.2, Simultaneous Start and Stop of Synchronous Channel Counters**
- **Section 18.4.3, Simultaneous Rewrite**

18.4.2.1 Rules of Synchronous Channel Operation

Number of masters and slaves

- Only even-numbered channels (CH0, CH2, CH4, …) can be set as master channels. Any channel apart from CH0 can be set as a slave channel.
- Only channels lower than the master channel can be set as slave channels, and several slave channels can be set for one master channel.

  Example: If CH2 is a master channel, CH3 and the lower channels (CH4, CH5, …) can be set as slave channels.
- If multiple master channels are used, slave channels cannot cross the master channels. Example: If CH0 and CH4 are master channels, CH1 to CH3 can be set as slave channels for CH0, but CH5 to CH15 cannot.

Operation clock

- The same operation clock must be set for the slave channel and the master channel. This is achieved using the TAUDnCMORm.TAUDnCKS[1:0] bits of the slave and master channel.

The basic concepts of master/slave usage and operation clocks are illustrated in **Figure 18.3**.

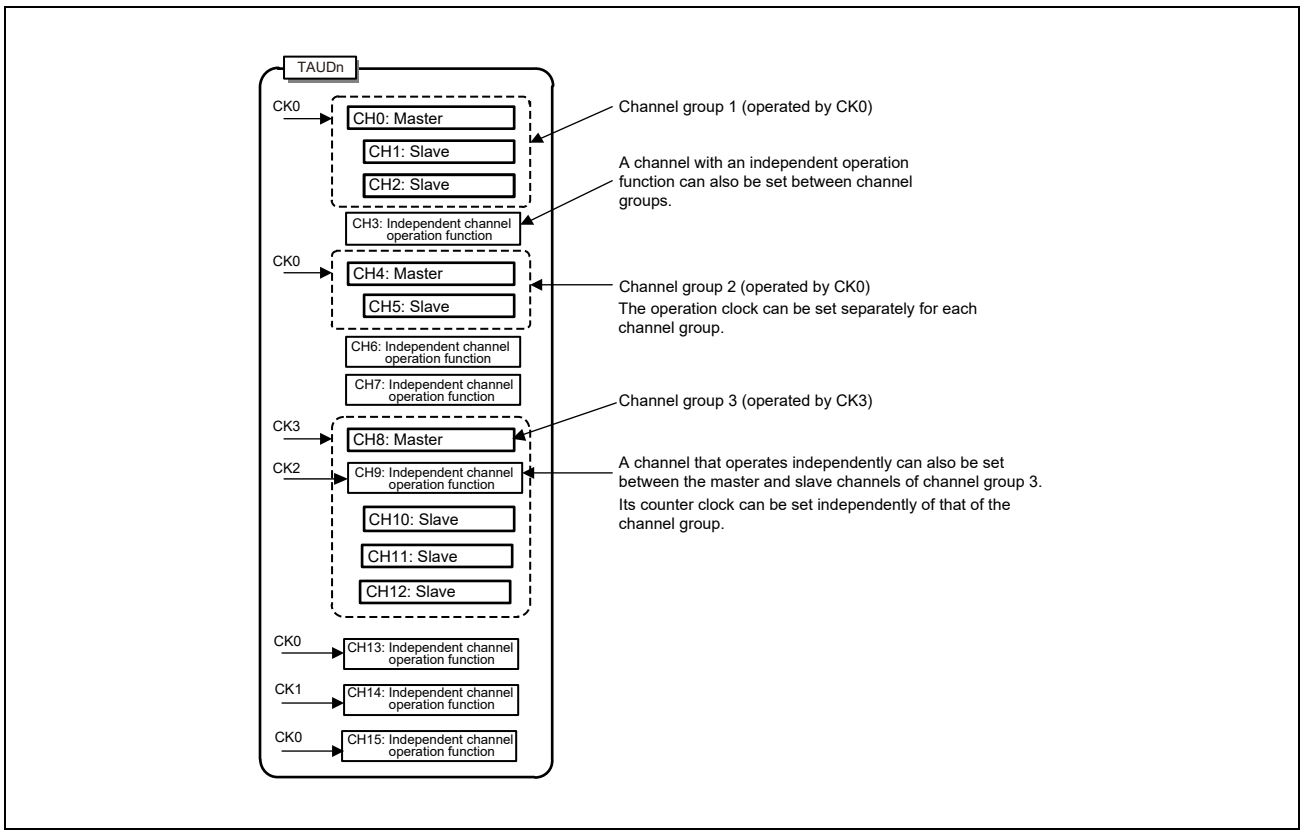


Figure 18.3 Grouping of Channels and Assignment of Count Clocks

Control trigger signal for master/salve channels

- Master channels can output control trigger signals to slave channels.
- Slave channels can use control trigger signals from master channels but cannot output control trigger signals for their own to lower channels.
- Master channels cannot use control trigger signals from upper master channels.

18.4.2.2 Simultaneous Start and Stop of Synchronous Channel Counters

Channels that are operated synchronously can be started and stopped simultaneously within the same unit and between the units.

(1) Simultaneous Start and Stop within the Same Unit

- To simultaneously start synchronized channels, the TAUDnTS.TAUDnTSm bits of the channels should be set at the same time.
- To simultaneously stop synchronized channels, the TAUDnTT.TAUDnTTm bits of the channels should be set at the same time.

Setting to the TAUDnTS.TAUDnTSm bits to 1 also sets the corresponding TAUDnTE.TAUDnTEm bits to 1, enabling counting. The count start timing depends on operating mode.

(2) Simultaneous Start for Multiple Units

Counters in different units can also be started simultaneously if the corresponding counters are enabled before receiving the simultaneous trigger signal.

For details on how to make a simultaneous start between the units, see **Section 24.2.3.1, Simultaneous Start Trigger Function**.

18.4.3 Simultaneous Rewrite

18.4.3.1 Overview of Operations

Simultaneous rewrite describes the ability to change the compare/start value and the output logic of multiple channels at the same time.

The corresponding data and control registers (TAUDnCDRm and TAUDnTOLm) can nevertheless be written at any time. The new value does not affect the counter operation or the output signal until simultaneous rewrite is triggered.

Simultaneous rewrite can be triggered by:

- The counter on the master channel or upper channel (depending on the selected operation mode) reaching a certain value
- INTTAUDnIm being issued on the upper channel specified by TAUDnRDC.TAUDnRDCm

There are four methods for simultaneous rewrite. These are listed in **Table 18.41**, along with how to specify them and when they cause simultaneous rewrite to be triggered.

Table 18.41 Simultaneous Rewrite Methods and when They are Triggered

Method	Simultaneous Rewrite Triggered when	TAUDnRDE. TAUDnRDEm	TAUDnRDS. TAUDnRDSm	TAUDnRDM. TAUDnRDMm
—	No simultaneous rewrite	0	0	0
A	The master channel (re)starts counting	1	0	0
B	Counting is started in the master channel. The master channel starts counting down at the peak of triangular cycle of the corresponding slave channel.	1	0	1
C1	INTTAUDnIm is generated on an upper channel specified by TAUDnRDC.TAUDnRDCm	1	1	0/1
C2	INTTAUDnIm is generated on an upper channel specified by TAUDnRDC.TAUDnRDCm that in turn is triggered by an external signal	1	1	0/1

Table 18.42 lists which of these four methods is available for each channel operation function. For more information about the individual channel operation functions, see the corresponding sections in **Section 18.4.9, Independent Channel Operation Functions** and **Section 18.4.12, Synchronous Channel Operation Functions**.

Table 18.42 Channel Operation Functions and Methods They Use

Function	A	B	C1	C2
Simultaneous Rewrite Trigger Output Function Type 1			√	
PWM Output Function	√		√	
One-Shot Pulse Output Function	√			
Trigger Start PWM Output Function	√			√
Delay Pulse Output Function	√			
Triangle PWM Output Function		√	√	
Triangle PWM Output Function with Dead Time		√	√	
Interrupt Request Signals Culling Function	√	√	√	
AD Conversion Trigger Output Function Type 1	√		√	
AD Conversion Trigger Output Function Type 2		√	√	
Non-Complementary Modulation Output Function Type 1	√		√	
Non-Complementary Modulation Output Function Type 2		√	√	
Complementary Modulation Output Function		√	√	

Note: √: Available, (Blank): Unavailable

18.4.3.2 How to Control Simultaneous Rewrite

Figure 18.4 shows the general procedure for simultaneous rewrite. The three main blocks (initial settings, start and counter count operation, and simultaneous rewrite) are explained afterwards.

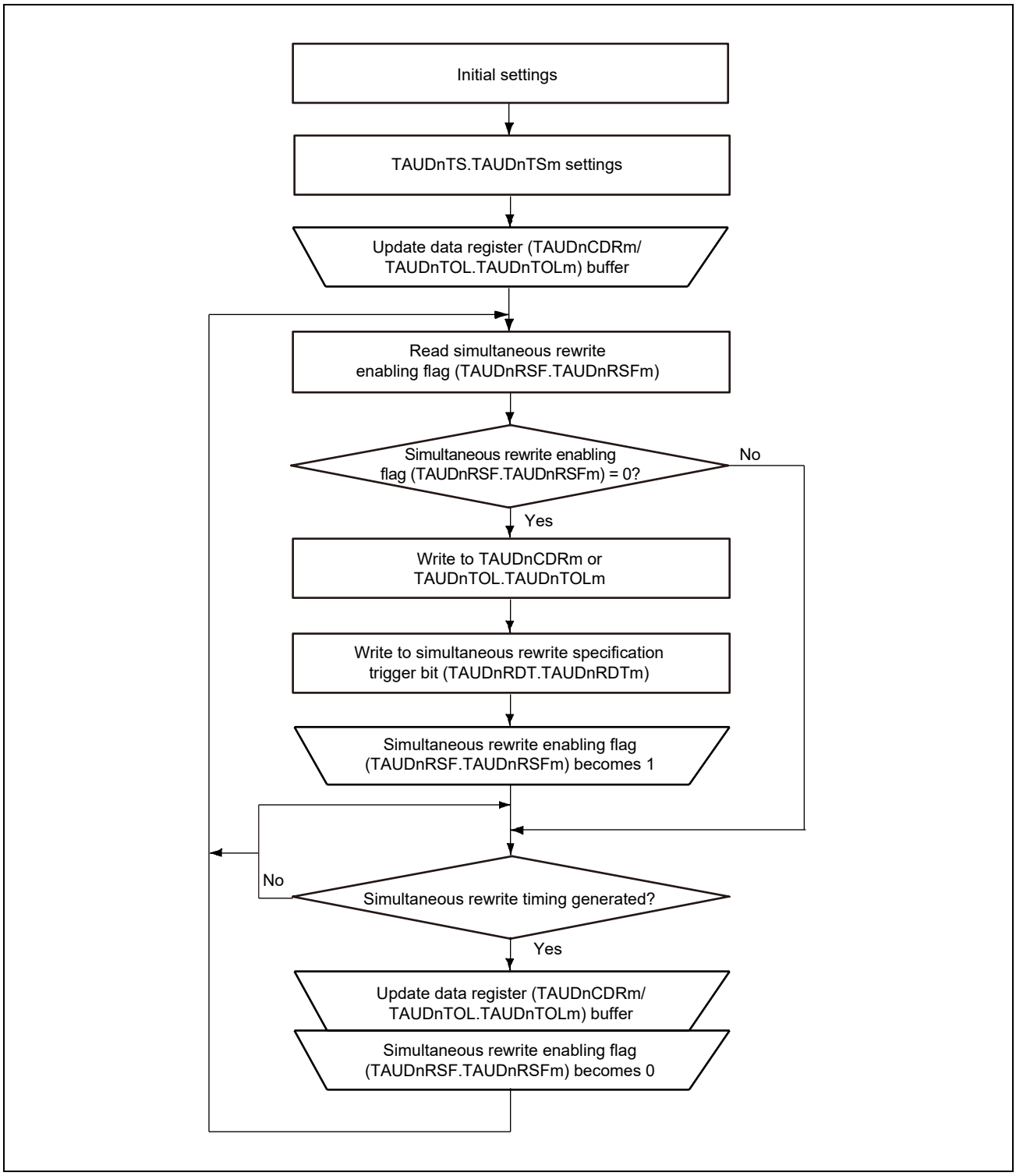


Figure 18.4 General Procedure for Simultaneous Rewrite

(1) Initial Settings

- To enable simultaneous rewrite in channel m, set TAUDnRDE.TAUDnRDEm = 1
- To select the type of simultaneous rewrite, set TAUDnRDM.TAUDnRDMm and TAUDnRDS.TAUDnRDSm according to the values listed in **Table 18.41, Simultaneous Rewrite Methods and when They are Triggered**.
- To select which upper channel is monitored for simultaneous rewrite triggers, use TAUDnRDC.TAUDnRDCm (prerequisite: TAUDnRDS.TAUDnRDSm is set in upper channel.)

(2) Start Counter and Count Operation

- To start all the TAUDnCNTm counters of the channel group, set the corresponding TAUDnTS.TAUDnTSm bits to1. The values of TAUDnTOL.TAUDnTOLm and the data registers (TAUDnCDRm) are loaded into the corresponding TAUDnTOL.TAUDnTOLm buffer (TAUDnTOL.TAUDnTOLm buf) and data buffer registers (TAUDnCDRm buf) and the counters start.
- Setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1 sets the reload flag (TAUDnRSF.TAUDnRSFm) to 1, enabling simultaneous rewrite. TAUDnRSF.TAUDnRSFm remains set to 1 until simultaneous rewrite is completed.
- When the specified trigger for simultaneous rewrite is detected, the TAUDnRSF.TAUDnRSFm bit is checked to see if simultaneous rewrite is enabled (TAUDnRSF.TAUDnRSFm = 1). If it is, simultaneous rewrite is carried out. Otherwise the simultaneous rewrite is not carried out and waits for the next trigger detection.

(3) Simultaneous Rewrite

- When simultaneous rewrite is enabled (TAUDnRSF.TAUDnRSFm = 1) and the simultaneous rewrite trigger is detected, the current values of the data registers are copied to their buffers. These values are then loaded into the corresponding counters and are applied the next time the counter starts or restarts.
- The TAUDnRSF.TAUDnRSFm bit is set to 0, and the system awaits the next simultaneous rewrite trigger.

18.4.3.3 Other General Rules of Simultaneous Rewrite

The following rules also apply:

- TAUDnRDE.TAUDnRDEm, TAUDnRDS.TAUDnRDSm, TAUDnRDM.TAUDnRDMm, and TAUDnRDC.TAUDnRDCm cannot be changed while the counter is in operation (TAUDnTE.TAUDnTEm = 1).
- TAUDnTOL.TAUDnTOLm can only be rewritten during operation with PWM output function or triangle PWM output function. For all other output functions, TAUDnTOL.TAUDnTOLm should be written before the counter starts. If it is rewritten while any other function is used, TAUDTTOUTm outputs an invalid wave.
- When an upper channel is used as a channel issuing the simultaneous rewrite trigger (TAUDnRDS.TAUDnRDSm = 1), the TAUDnRDC.TAUDnRDCm bit controls all the lower channels. This means that if the TAUDnRDC.TAUDnRDCm bits of CH2 and CH7 are set to 1 and the TAUDnRDC.TAUDnRDCm bits of other channels are set to 0, CH2 and CH7 serve as simultaneous rewrite trigger generation channels. CH2 controls the lower channels CH3 to CH6, and CH7 controls the lower channels CH8 to CH15.
- If simultaneous rewrite is enabled and an upper channel is selected for the simultaneous rewrite trigger (TAUDnRDE.TAUDnRDEm and TAUDnRDS.TAUDnRDSm = 1) but no upper channel is set (TAUDnRDC.TAUDnRDC[15:0] = 0), simultaneous rewrite cannot take place.

18.4.3.4 Types of Simultaneous Rewrite

In the following section, the four simultaneous rewrite methods are explained using timing diagrams.

(1) Simultaneous Rewrite when the Master Channel (Re)starts Counting (Method A)

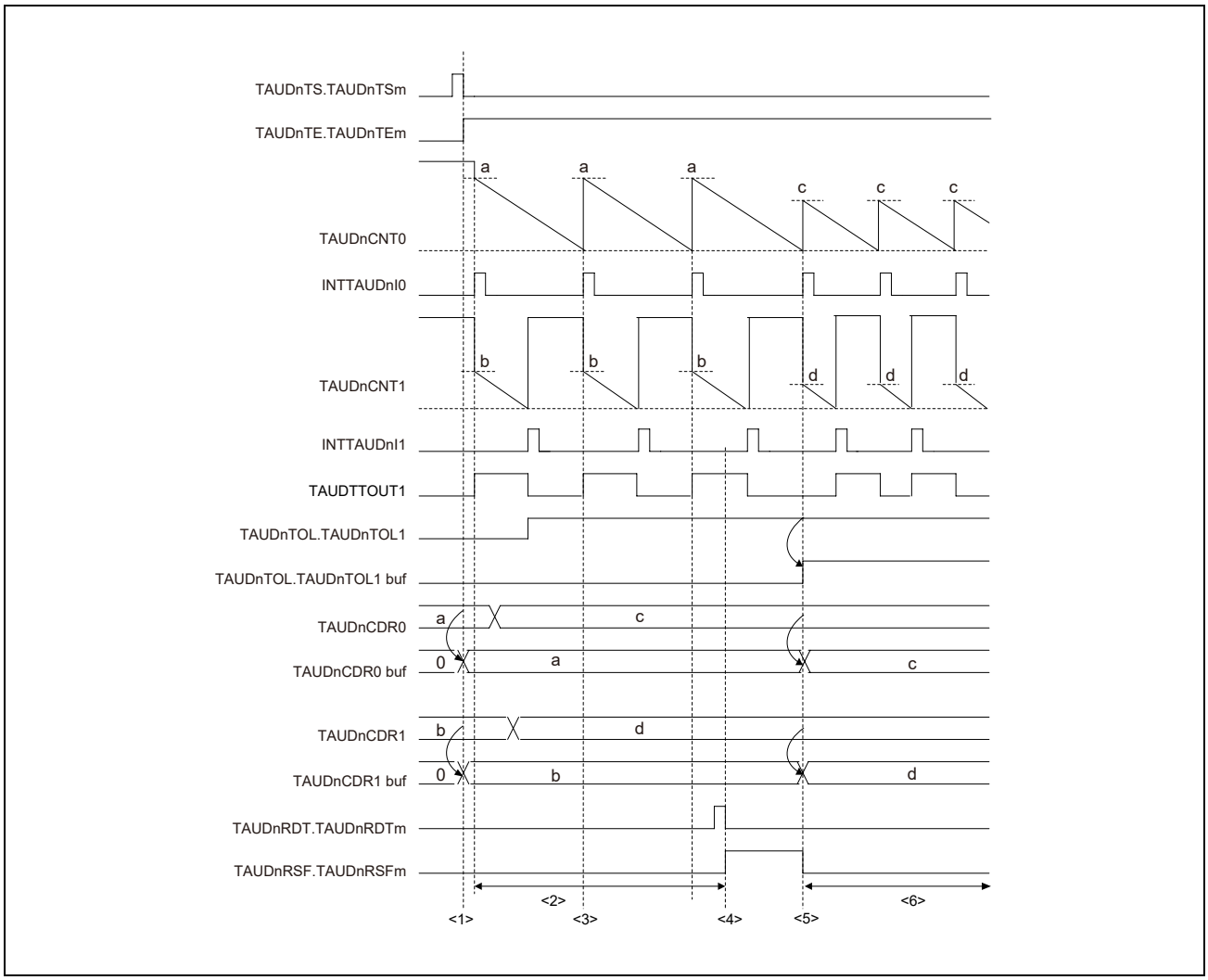


Figure 18.5 Simultaneous Rewrite when the Master Channel (Re) Starts Counting

Setting

- CH0 is the master channel, which counts down, and CH1 represents an arbitrary slave channel. The simultaneous rewrite method A is applied.

Description:

(1) When TAUDnTS.TAUDnTSm = 1 is set, the value of TAUDnCDRm is copied to the TAUDnCDRm buffer.

(2) The TAUDnCDRm and TAUDnTOL.TAUDnTOLm registers can be written at any time.

(3) CH0 restarts counting, but simultaneous rewrite does not occur because it is disabled (TAUDnRSF.TAUDnRSFm = 0)

(4) The reload data trigger bit (TAUDnRDT.TAUDnRDTm) is set to 1 which sets the status flag (TAUDnRSF.TAUDnRSFm = 1), enabling simultaneous rewrite.

(5) Because simultaneous rewrite is enabled, it is triggered when CH0 restarts counting. The TAUDnCDRm value is loaded into the TAUDnCDRm buffer and the TAUDnTOL.TAUDnTOLm value is loaded into the TAUDnTOL.TAUDnTOLm buffer.

(6) The counters count down and await the next simultaneous rewrite trigger. The values of TAUDnCDRm and TAUDnTOL.TAUDnTOLm can be changed again.

(2) Simultaneous Rewrite at the Peak of a Triangular Cycle of Slave Channel (Method B)

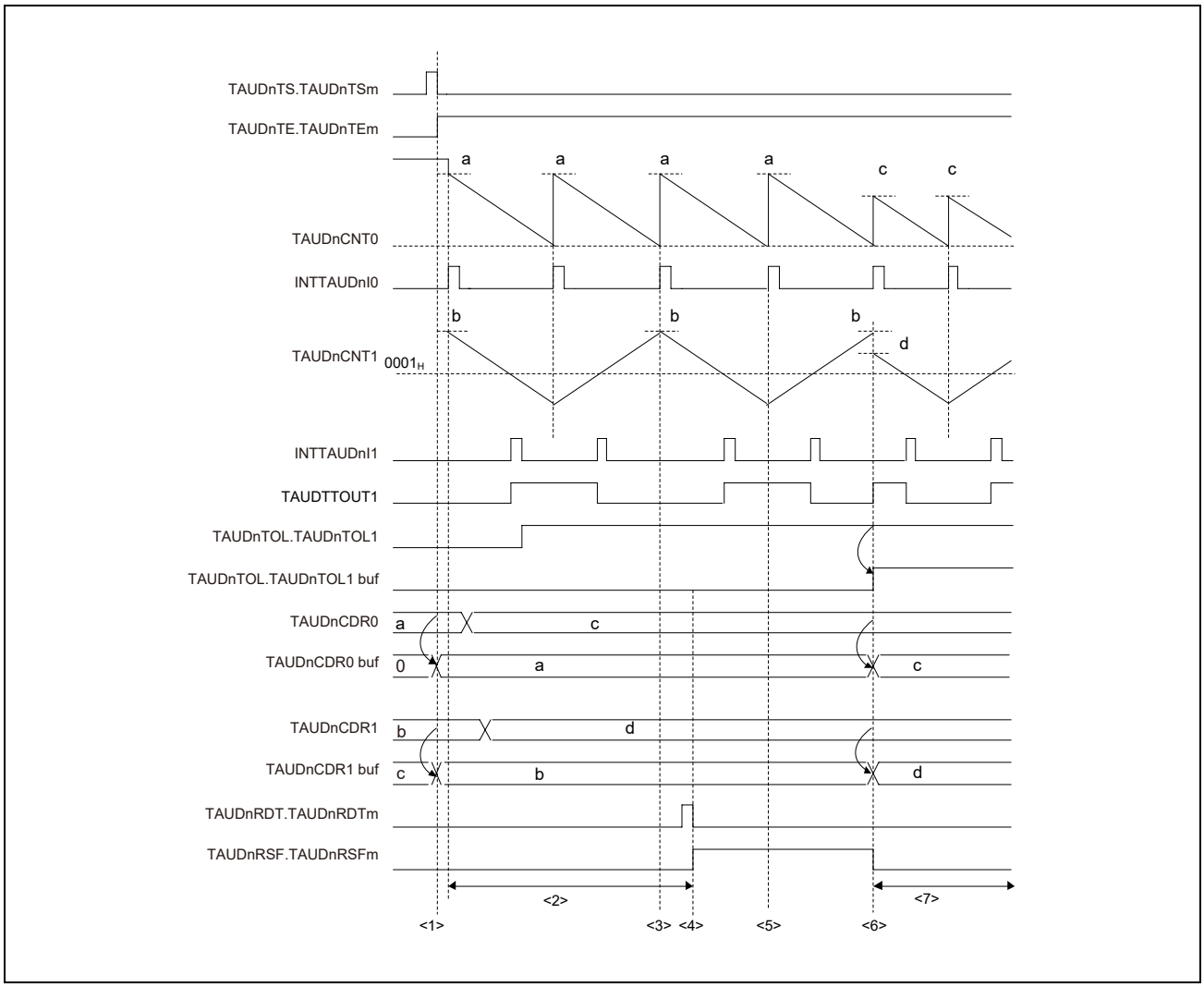


Figure 18.6 Simultaneous Rewrite at the Peak of a Triangular Cycle of Slave Channel

Setting

- CH0 is the master channel which performs counting down, and CH1 represents an arbitrary slave channel. The simultaneous rewrite method B is applied.

Description:

(1) When TAUDnTS.TAUDnTSm = 1 is set, the value of TAUDnCDRm is copied to the TAUDnCDRm buffer.

(2) The TAUDnCDRm and TAUDnTOL registers can be written at any time.

(3) Simultaneous rewrite does not occur because it is disabled (TAUDnRSF.TAUDnRSFm = 0).

(4) The reload data trigger bit (TAUDnRDT.TAUDnRDTm) is set to 1 which sets the status flag (TAUDnRSF.TAUDnRSFm = 1), enabling simultaneous rewrite.

(5) Simultaneous rewrite does not take place at the bottom of the triangular cycle.

(6) Simultaneous rewrite takes place at the top of the triangular cycle. The TAUDnCDRm value is loaded into the TAUDnCDRm buffer, the TAUDnTOL.TAUDnTOLm value is loaded into the TAUDnTOL.TAUDnTOLm buffer.

(7) The counters count down and await the next simultaneous rewrite trigger. The values of TAUDnCDRm and TAUDnTOL.TAUDnTOLm can be changed again.

(3) Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm (Method C1)

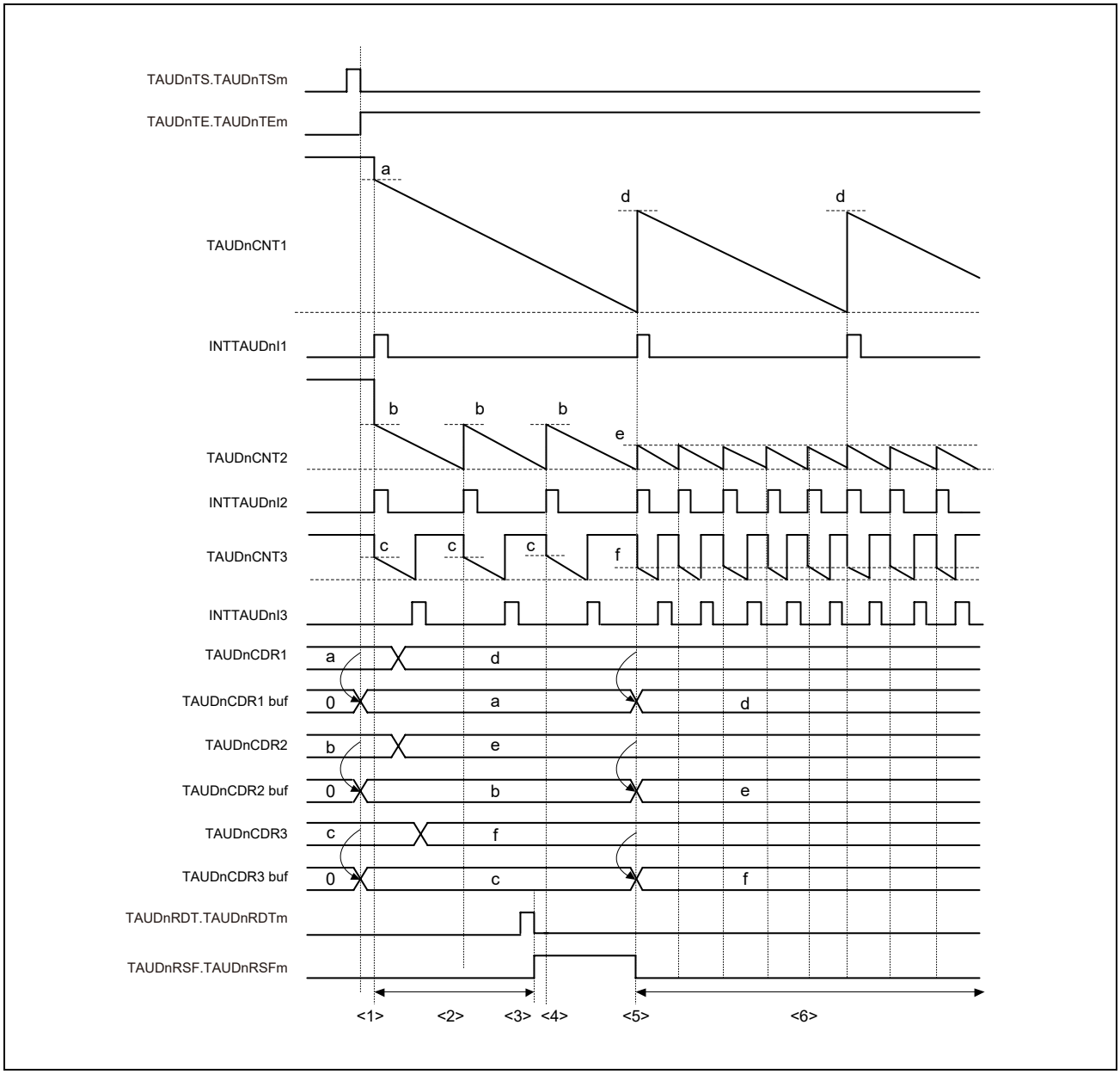


Figure 18.7 Simultaneous Rewrite When INTTAUDnIm Is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm

Setting

- CH1 is an upper channel which performs counting down, CH2 is a master channel, and CH3 is the slave channel. The simultaneous rewrite method C1 is applied. The TAUDnRDC register specifies a channel which generates simultaneous rewrite triggers.

Description:

(1) When TAUDnTS.TAUDnTSm is set to 1, the TAUDnCDRm value is copied to the TAUDnCDRm buffer.

(2) The TAUDnCDRm register is always ready to write.

(3) By setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1, the status flag is set (TAUDnRSF.TAUDnRSFm = 1) to enable simultaneous rewrite.

(4) Simultaneous rewrite is triggered only by a CH1 interrupt. Therefore, simultaneous rewrite is not conducted even if enabled.

(5) Simultaneous rewrite is triggered by INT1 which is generated when counter 1 reaches 0000_H. The TAUDnCDRm values are loaded into the corresponding TAUDnCDRm buffers.

(6) The counter counts down and awaits the next simultaneous rewrite trigger. The values of the TAUDnCDRm registers can be changed again.

(4) Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal (Method C2)

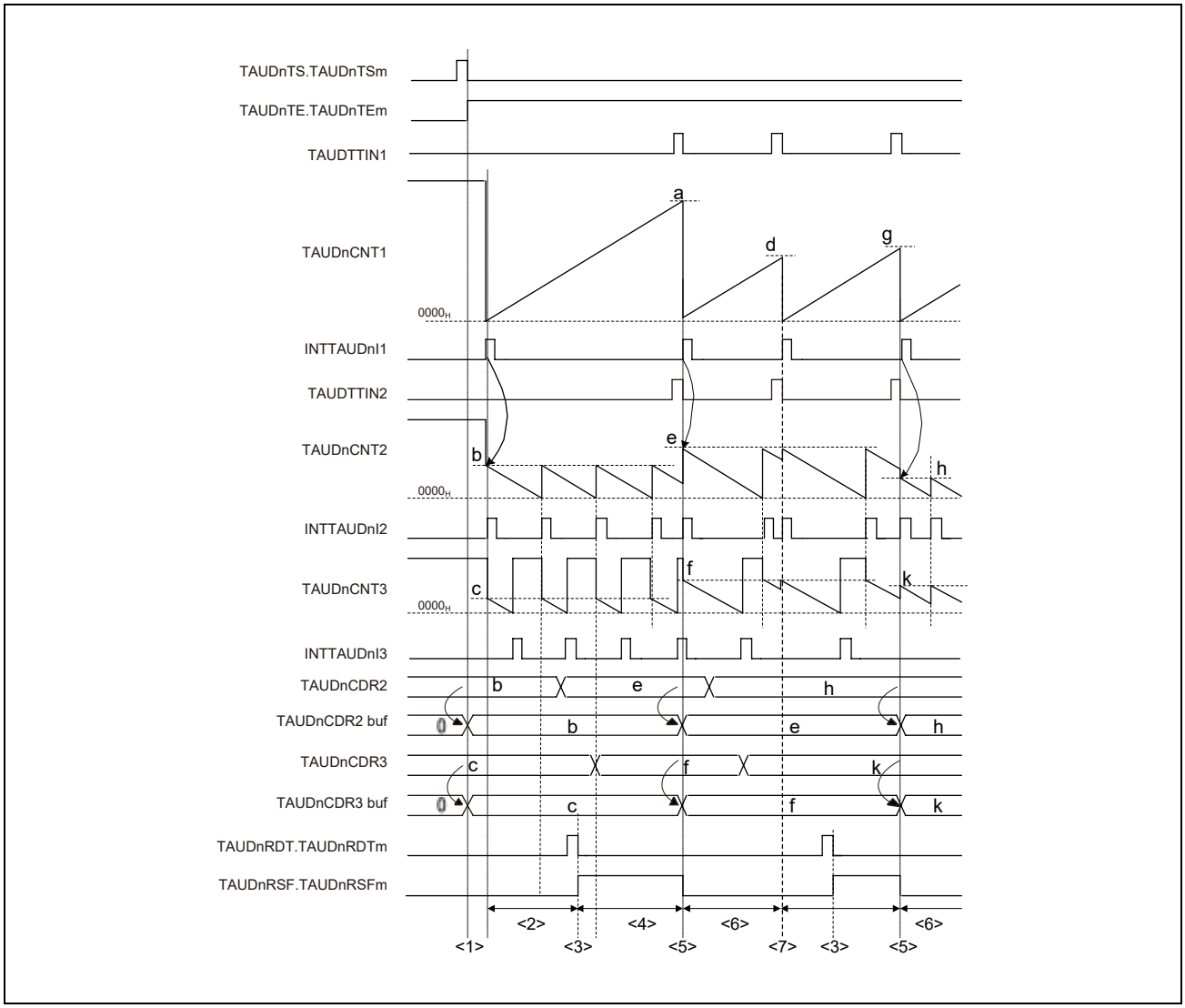


Figure 18.8 Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal

Setting

CH1 is an upper channel which performs counting up, CH2 is a master channel, and CH3 is the slave channel. The synchronous channel operation method C2 is applied. The TAUDnRDC register specifies which upper channel is monitored for an INTTAUDnIm trigger.

Description:

(1) When TAUDnTS.TAUDnTSm is set to 1, the TAUDnCDRm value is copied to the TAUDnCDRm buffer. However, as TAUDnCDR1 operates in capture mode, the TAUDnCDR1 value is not copied to the TAUDnCDR1 buffer.

(2) The TAUDnCDRm register is always ready to write.

(3) By setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1, the status flag is set (TAUDnRSF.TAUDnRSFm = 1) to enable simultaneous rewrite.

(4) Simultaneous rewrite is triggered only by a CH1 interrupt. Therefore, simultaneous rewrite is not conducted even if enabled.

(5) Simultaneous rewrite is triggered by INT1 which is caused by external signal TIN1. The TAUDnCDRm values are written to the corresponding TAUDnCDRm buffers.

(6) The counters count down and await the next simultaneous rewrite trigger. The values of the TAUDnCDRm registers can be changed again.

(7) An external signal occurs at TIN2 but simultaneous rewrite does not take place because it is disabled (TAUDnRSF.TAUDnRSFm = 0).

18.4.4 Channel Output Modes

The output of the TAUDTTOUTm pin can be controlled in two ways, the latter of which can be further split into individual modes.

- By software (TAUDnTOE.TAUDnTOEm = 0)
  When controlled by software, the value written in the output register bit (TAUDnTO.TAUDnTOm) is sent out of the output pin (TAUDTTOUTm).
- By TAUD signals (TAUDnTOE.TAUDnTOEm = 1)
  When controlled by TAUD signals, the output level of TAUDTTOUTm is set or reset or toggled by internal signals. The value of TAUDnTO.TAUDnTOm is updated accordingly to reflect the value of TAUDTTOUTm
  - Independently (TAUDnTOM.TAUDnTOMm = 0)
    In case of independent operation, the output of the TAUDTTOUTm pin is only affected by settings of channel m. Therefore, independent channel operation should be selected (TAUDnTOM.TAUDnTOMm = 0).
  - Synchronously (TAUDnTOM.TAUDnTOMm = 1)
    In case of synchronous operation, the output of the TAUDTTOUTm pin is affected by settings of channel m and those of other channels. Therefore, synchronous channel operation should be selected for all synchronized channels (TAUDnTOM.TAUDnTOMm = 1).

The TAUDnTO.TAUDnTOm bit can always be read to determine the current value of TAUDTTOUTm, regardless of whether the pin is controlled by software, operated independently, or operated synchronously.

Control bits

The settings of the control bits required to select a specific channel output mode are listed in **Table 18.43, Channel Output Modes**.

The channel output modes are described in details below.

- **Section 18.4.4.2, Channel Output Modes Controlled Independently by TAUDn Signals**
- **Section 18.4.4.3, Channel Output Modes Controlled Synchronously by TAUDn Signals**

Batch operation of TAUDnTOm bit

Whether a set value is reflected to the TAUDnTOm bit or not is controlled by the TAUDnTOE.TAUDnTOEm bit.

The TAUDnTOm setting is written only to the bit (channel) set with TAUDnTOE.TAUDnTOEm bit = 0 when a write to the TAUDnTO register is attempted. No TAUDnTOm setting is reflected to the bit (channel) set with TAUDnTOE.TAUDnTOEm bit = 1.

NOTE

TAUDnTO.TAUDnTOm bit is placed so that its bit number corresponds to a channel number.

Output logic

Positive or negative logic of the output is specified by control bit TAUDnTOL.TAUDnTOLm.

The value of TAUDnTOL.TAUDnTOLm bit should be set before the counter is started. It can only be changed during operation with PWM output function or triangle PWM output function. Otherwise, changes to TAUDnTOL.TAUDnTOLm result in an invalid TAUDTTOUTm signal output.

See **Section 18.4.3, Simultaneous Rewrite.**

The various channel output modes and the channel output control bits are listed in **Table 18.43**.

Table 18.43 Channel Output Modes

Channel Output Mode	TAUDn TOE. TAUDn TOEm	TAUDn TOM. TAUDn TOMm	TAUDn TOC. TAUDn TOCm	TAUDn TDE. TAUDn TDEm	TAUDn TRE. TAUDn TREm	TAUDn TME. TAUDn TMEm	TAUDn TDM. TAUDn TDMm
By software							
Independent channel output mode controlled by software	0	X	X	X	X	X	X
By TAUD signals, independently							
Independent channel output mode 1	1	0	0	0	0	0	0
with real-time output	1	0	0	0	1	0	0
Independent channel output mode 2	1	0	1	0	0	0	0
By TAUD signals, synchronously							
Synchronous channel output mode 1	1	1	0	0	0	0	0
with non-complementary modulation output	1	1	0	0	1	X	0
Synchronous channel output mode 2	1	1	1	0	0	0	0
with dead time output	1	1	1	1	0	0	0
with one-phase PWM output	1	1	1	1	0	0	1
with complementary modulation output	1	1	1	1	1	1	0
with non-complementary modulation output	1	1	1	0	1	1	0

Note: All combinations not listed in this table are forbidden.
Bits marked with an X can be set to any value.

NOTES

1. The following bits cannot be changed during count operation (TAUDnTE.TAUDnTE = 1):
   - TAUDnTOE.TAUDnTOEm
   - TAUDnTOM.TAUDnTOMm
   - TAUDnTOC.TAUDnTOCm
   - TAUDnTDE.TAUDnTDEm
   - TAUDnTRE.TAUDnTREm
   - TAUDnTDM.TAUDnTDMm
2. The following bits cannot be changed during count operation (TAUDnTE.TAUDnTEm = 1) except in channel output modes with modulation output:
   - TAUDnTME.TAUDnTMEm
   - TAUDnTDL.TAUDnTDLm

18.4.4.1 General Procedures for Specifying a Channel Output Mode

This section describes the general procedures for specifying a TAUDTTOUTm channel output mode. The prerequisite is that timer output operation is disabled (TAUDnTOE.TAUDnTOEm = 0).

(1) Set TAUDnTO.TAUDnTOm to specify the initial level of the TAUDTTOUTm output.

(2) Set channel output mode according to **Table 18.43, Channel Output Modes**, and the output logic using the TAUDnTOL.TAUDnTOLm bit.

(3) Start the counter (TAUDnTS.TAUDnTSm = 1).

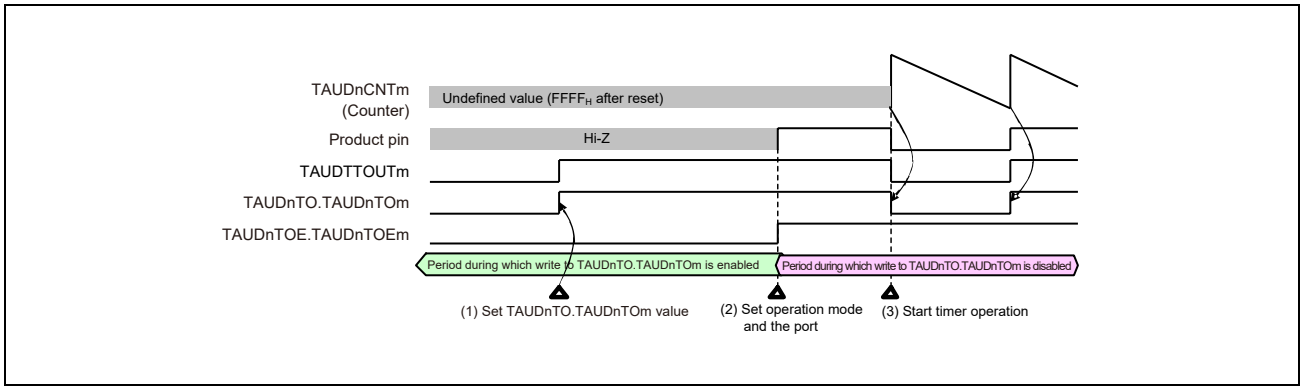


Figure 18.9 General Procedure for Specifying a TAUDTTOUTm Channel Output Mode

18.4.4.2 Channel Output Modes Controlled Independently by TAUDn Signals

This section lists the channel output modes that are controlled independently by TAUDn signals. The control bits used to specify a mode are listed in **Table 18.43, Channel Output Modes**.

(1) Independent Channel Output Mode 1

Set/reset conditions

In this output mode, TAUDTTOUTm toggles when INTTAUDnIm is detected. The value of TAUDnTOL.TAUDnTOLm is ignored.

Prerequisites

There are no prerequisites other than those shown in **Table 18.43, Channel Output Modes**.

(2) Independent Channel Output Mode 1 with Real-Time Output

In this output mode, the value of TAUDnTRO.TAUDnTROm bit of the trigger channel is output to TAUDTTOUTm. The trigger channel is specified by setting the corresponding TAUDnTRC.TAUDnTRCm bit to 1. It controls all lower channels for which TAUDnTRC.TAUDnTRCm = 0.

Set/reset conditions

The value of TAUDnTRO.TAUDnTROm bit is sent to TAUDTTOUTm only when an INTTAUDnIm interrupt occurs on the trigger channel. The interrupt is generated either:

- at certain specified intervals or
- on detection of an effective TAUDTTINm input edge/counter start

The type of trigger is set using the TAUDnCMORm.TAUDnMD[4:1] bits.

Prerequisites

Both master and slave channels can be set as a trigger generation channel. A channel for which TAUDnTRC.TRCm is set to 1 serves as a trigger generation channel even if TAUDnTRE.TAUDnTREm is set to 0. If there is no channel for which TAUDnTRC.TAUDnTRCm is set to 1 or if TAUDnTRC.TAUDnTRC0 = 0, real-time output cannot take place.

This can be seen in **Figure 18.10**.

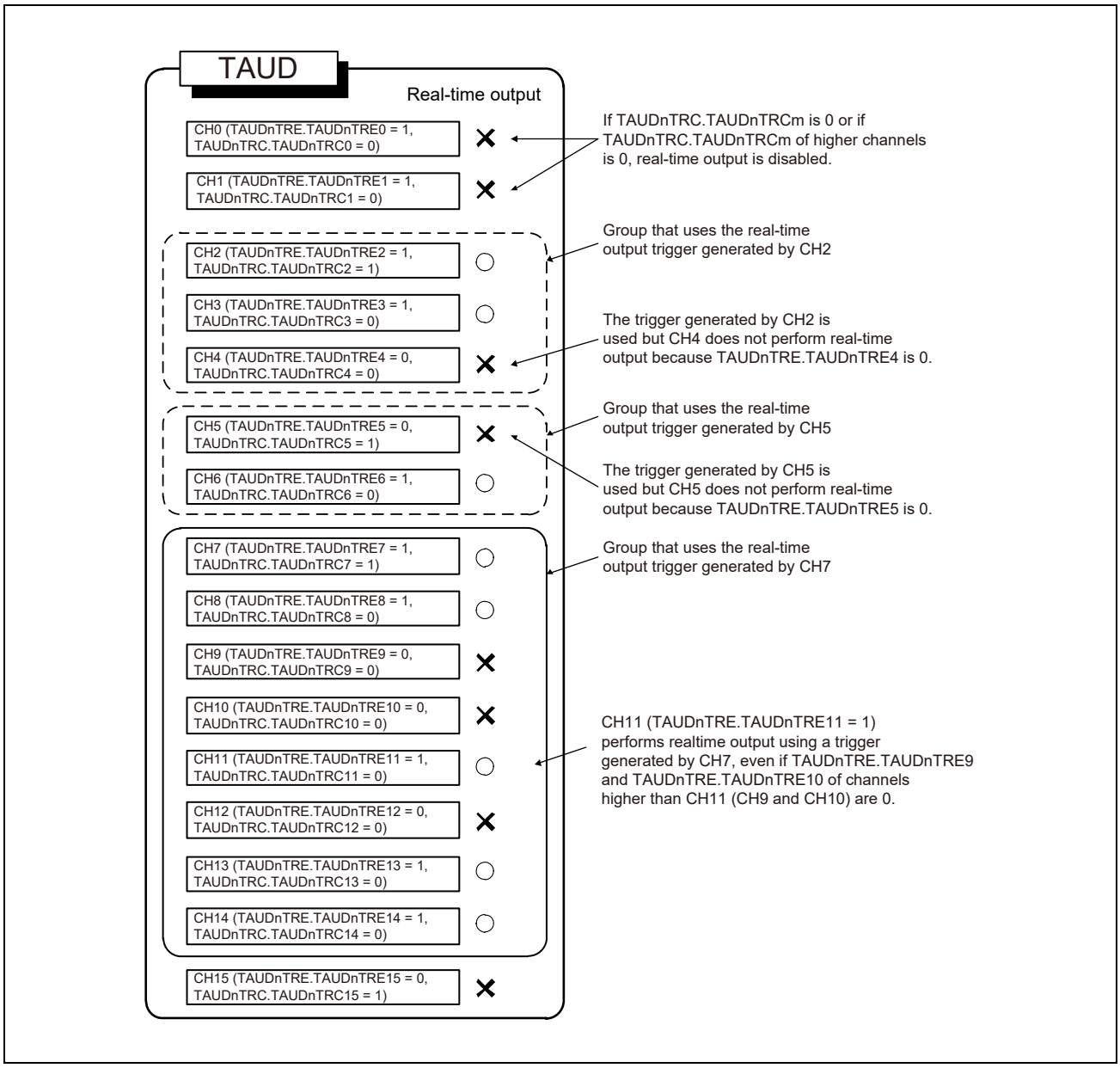


Figure 18.10 Real-Time Output

(3) Independent Channel Output Mode 2

Set/reset conditions

In this output mode, TAUDTTOUTm is set when INTTAUDnIm occurs at the time of count start, and reset when INTTAUDnIm occurs due to a match between TAUDnCNTm and TAUDnCDRm.

Prerequisites

There are no prerequisites other than those shown in **Table 18.43, Channel Output Modes.**

18.4.4.3 Channel Output Modes Controlled Synchronously by TAUDn Signals

This section lists the channel output modes that are controlled synchronously by TAUDn signals. The control bits used to specify a mode are listed in **Table 18.43, Channel Output Modes**.

(1) Synchronous Channel Output Mode 1

Set/reset conditions

In this output mode, INTTAUDnIm of master channel serves as a set signal and INTTAUDnIm of the slave channel as a reset signal. If INTTAUDnIm of master channel and INTTAUDnIm of the slave channel are generated at the same time, INTTAUDnIm of the slave channel (reset signal) has priority over INTTAUDnIm (set signal) of master channel, i.e., the master channel is ignored.

Prerequisites

There are no prerequisites other than those shown in **Table 18.43, Channel Output Modes**.

(2) Synchronous Channel Output Mode 1 with Non-Complementary Modulation Output

Set/reset conditions

In this output mode, TAUDTTOUTm outputs the result of an AND operation between the PWM output and the real-time output bit (TAUDnTRO.TAUDnTROm) of a channel.

The phase period to which the dead time is added is specified using the TAUDnTDL.TDLm bit; for positive phase set TAUDnTDL.TAUDnTDLm = 0 and for negative phase set TAUDnTDL.TAUDnTDLm = 1.

Prerequisites

A set of at least three channels is required to generate the PWM output. The master channel and slave channel 1 generate a period, and slave channel 2 generates the duty cycle. In typical applications, five more slave channels are also used that operate in the same manner as slave channel 2.

Only the PWM output and the real-time output bit of the same channel can be combined.

TAUDnTRO.TAUDnTROm, TAUDnTME.TAUDnTMEm, and TAUDnTDL.TAUDnTDLm can only be changed during count operation.

- If TAUDnTME.TAUDnTMEm is changed, its new value is applied upon detection of INTTAUDnIm on the specified channel.
- If TAUDnTME.TAUDnTMEm and TAUDnTDL.TAUDnTDLm are changed, their new values are applied upon detection of INTTAUDnIm on the master channel.

(3) Synchronous Channel Output Mode 2

In this output mode, the operating mode should be set to up/down count mode. The result is a triangle PWM wave at TAUDTTOUTm. For details, see **Section 18.4.12.7, Triangle PWM Output Function**.

Set/reset conditions

TAUDnCNTm of the slave channel counts down and up alternatively. When it passes 0001_Hit generates an interrupt, causing TAUDTTOUTm to toggle.

Prerequisites

A set of two channels is required to generate the triangle PWM output. TAUDTTOUTm should be set to 0 before the function starts.

(4) Synchronous Channel Output Mode 2 with Dead Time Output

In this output mode, a dead time delay is added to TAUDTTOUTm. The set/reset conditions are shown in **Figure 18.11**.

Set/reset conditions

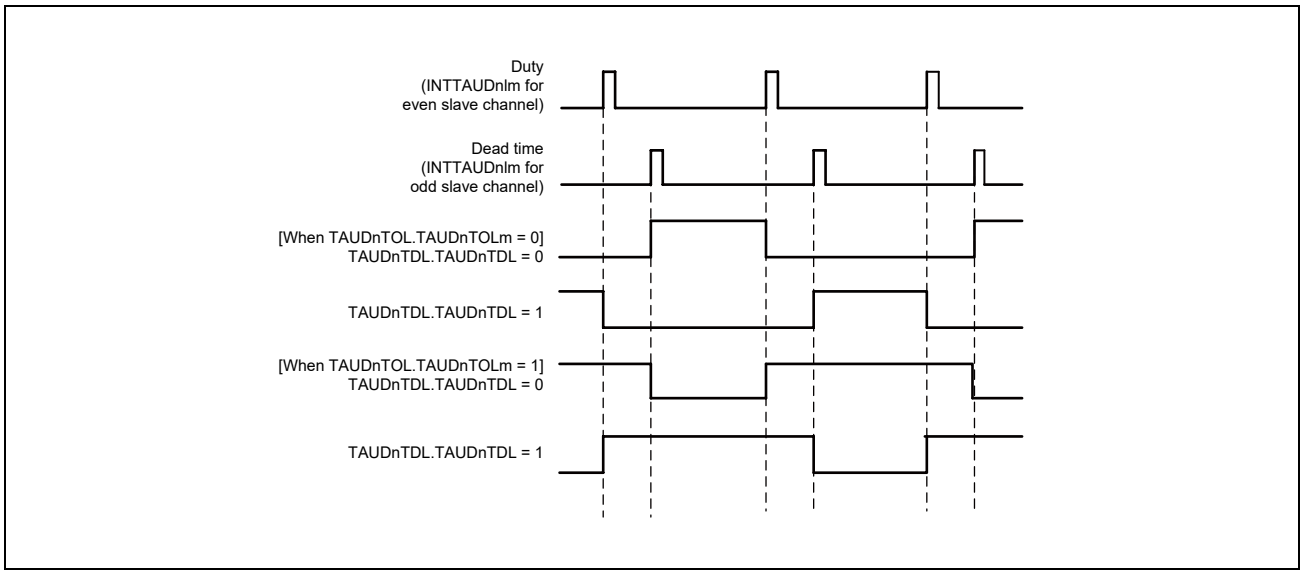


Figure 18.11 Set/Reset Conditions for Synchronous Channel Output Mode 2 with Dead Time Output

With regard to the edge to which dead time is added, set TAUDnTDL.TAUDnTDLm = 0 for rising edges and TAUDnTDL.TAUDnTDLm = 1 for falling edges.

Prerequisites

Dead time control requires a set of three channels, each operating in the following modes:

- One master channel
  The master channel should be set to interval timer mode.
- One even slave channel
  The even slave channel should be set up/down count mode.
- One odd slave channel (even-numbered channel + 1)
  The odd slave channel should be set to one-count mode.

The values of the following bits should be the same for the odd-numbered channel and the even-numbered channel:

- TAUDnTOE.TAUDnTOEm
- TAUDnTME.TAUDnTMEm
- TAUDnTRE.TAUDnTREm
- TAUDnTOM.TAUDnTOMm
- TAUDnTOC.TAUDnTOCm
- TAUDnTDE.TAUDnTDEm
- TAUDnTDM.TAUDnTDMm

(5) Synchronous Channel Output Mode 2 with One-Phase PWM Output

In this output mode, a dead time delay is added to TAUDTTOUTm. The set/reset conditions are shown in **Figure 18.12**.

Set/reset conditions

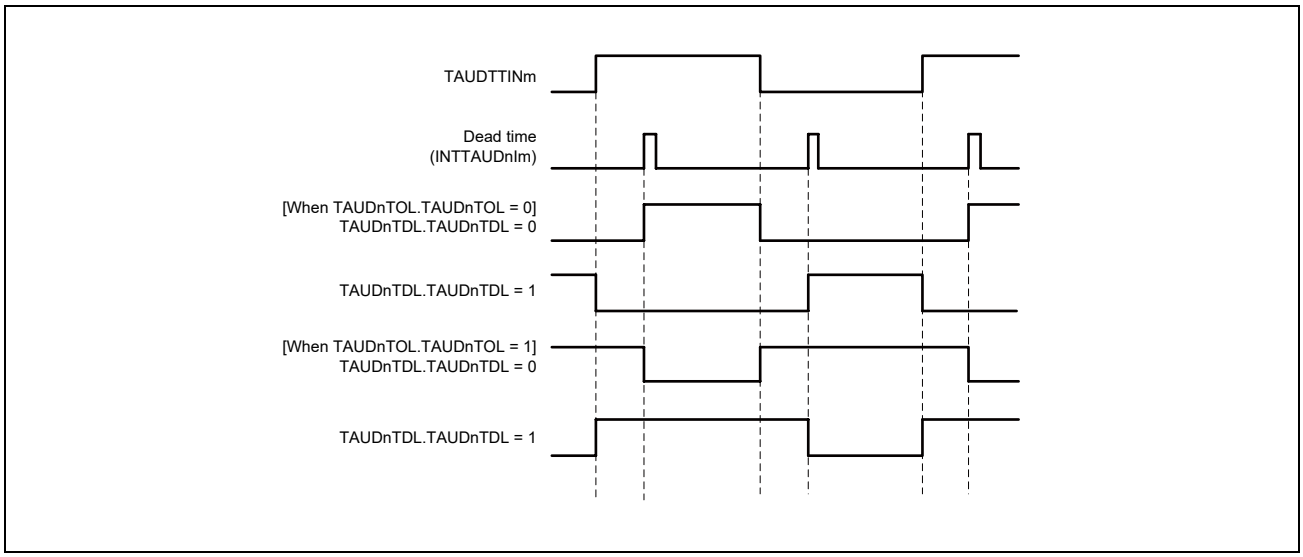


Figure 18.12 Set/Reset Conditions for Synchronous Channel Output Mode 2 with One-Phase PWM Output

With regard to the edge to which dead time is added, set TAUDnTDL.TAUDnTDLm = 0 for rising edges and TAUDnTDL.TAUDnTDLm = 1 for falling edges.

Prerequisites

One-phase PWM output control requires a set of two channels:

- One even slave channel.
- One odd slave channel (even-numbered channel + 1)
  The odd slave channel should be set to one-count mode.

The values of the following bits should be the same for the odd-numbered channel and the even-numbered channel:

- TAUDnTOE.TAUDnTOEm
- TAUDnTME.TAUDnTMEm
- TAUDnTRE.TAUDnTREm
- TAUDnTOM.TAUDnTOMm
- TAUDnTOC.TAUDnTOCm
- TAUDnTDE.TAUDnTDEm
- TAUDnTDM.TAUDnTDMm

(6) Synchronous Channel Output Mode 2 with Complementary Modulation Output

Set/reset conditions

In this output mode, TAUDTTOUTm outputs a PWM signal, a high signal, or a low signal depending on the value of real-time output bit (TAUDnTRO.TAUDnTROm), the modulation output bit (TAUDnTME.TAUDnTMEm), and the output level bit (TAUDnTOL.TAUDnTOLm) of a pair of slave channels.

For details, see **Section 18.4.13.3, Complementary Modulation Output Function**.

Prerequisites

A set of at least four channels is required for this mode. The master channel and slave channel 1 generate a period, slave channel 2 generates a duty cycle, and slave channel 3 generates dead time. Slave channels 2 and 3 are a pair. In typical applications, 4 more channels are also used, which operates in the same manner as slave channels 2 and 3 respectively.

TAUDnTRO.TAUDnTROm, TAUDnTME.TAUDnTMEm, and TAUDnTDL.TAUDnTDLm can only be changed during count operation.

- If TAUDnTME.TAUDnTMEm is changed during operation, its new value is applied upon detection of INTTAUDnIm at the specified channel.
- If TAUDnTME.TAUDnTMEm and TAUDnTDL.TAUDnTDLm are changed, their new values are applied upon detection of INTTAUDnIm on an even slave channel.

(7) Synchronous Channel Output Mode 2 with Non-Complementary Modulation Output

The difference from synchronous channel output mode 1 with non-complementary modulation output is the PWM wave shape.

Mode 1 has a rectangular wave while mode 2 has a triangular wave.

18.4.5 Start Timing in Each Operating Modes

This section describes the timing at which the counter starts after TAUDnTS.TAUDnTSm is set to 1 in each operating mode.

In all modes, the value of data register and whether or not an interrupt occurs depends on mode and register settings.

CAUTION

The count start timing described in this section is for your reference. Actually, the count start timing depends on the count clock timing.

18.4.5.1 Interval Timer Mode, Judge Mode, Capture Mode, Up/Down Count Mode, and Count Capture Mode

The counter starts operating with the next count clock after TAUDnTS.TAUDnTSm is set to 1. The value of data register is also loaded when the counter starts.

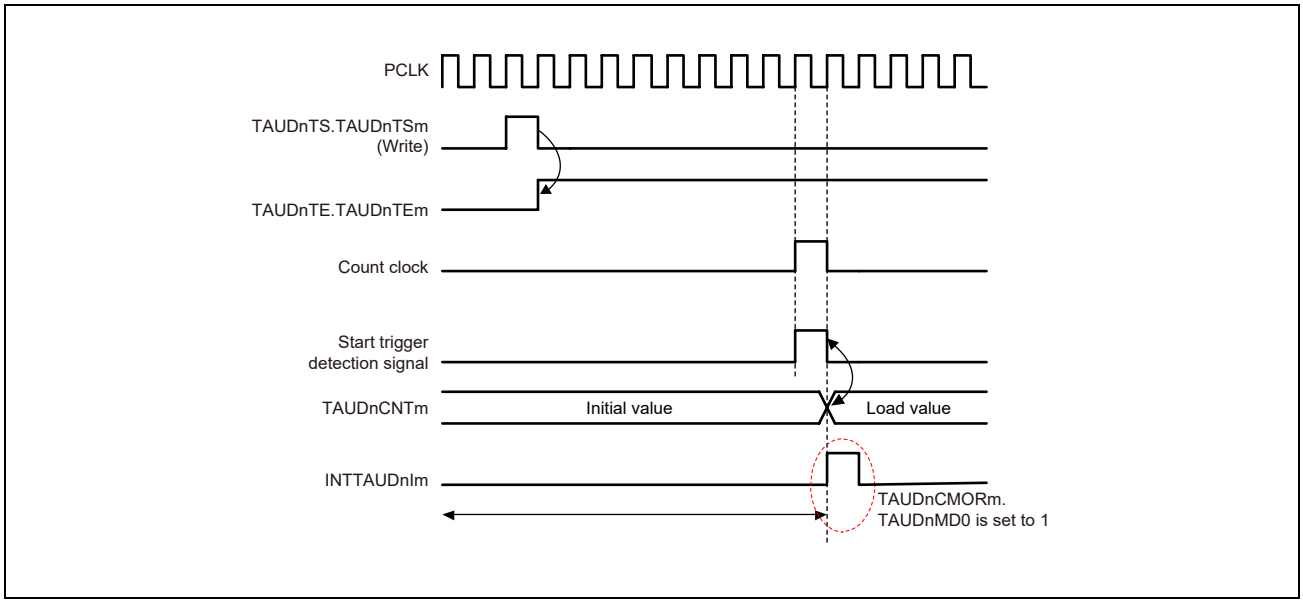


Figure 18.13 Start Timing in Interval Timer Mode, Judge Mode, Capture Mode, Up/Down Count Mode, and Count Capture Mode

NOTE

Make sure to set TAUDnCMORm.TAUDnMD0 to 0 when using the up/down count mode.

18.4.5.2 Event Count Mode

The value of data register is loaded as soon as TAUDnTS.TAUDnTSm is set to 1. The counter also starts immediately. The value of data register decrements with subsequent count clocks.

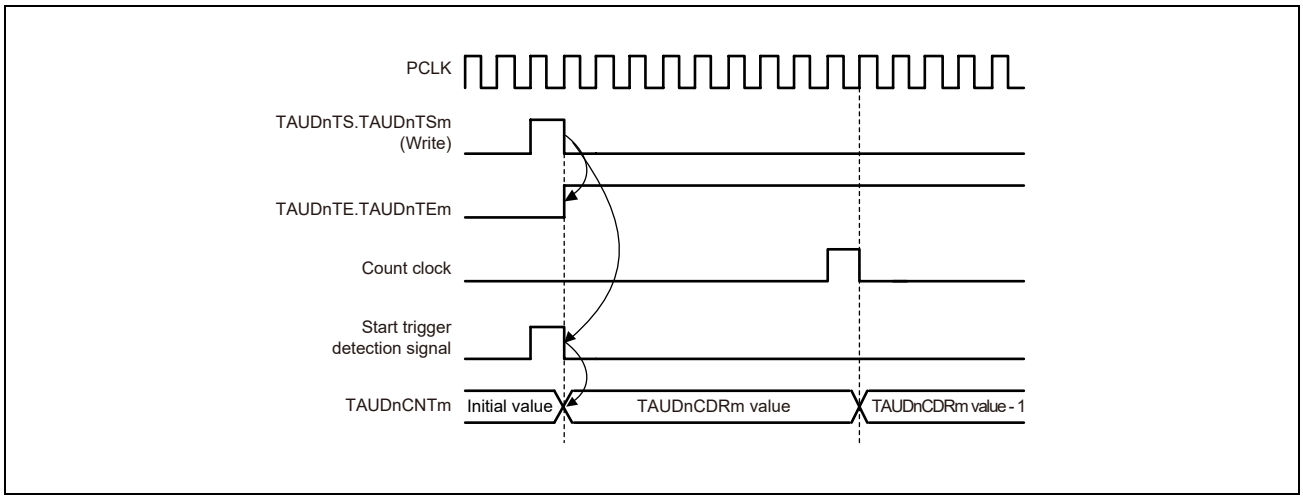


Figure 18.14 Start Timing in Event Count Mode

18.4.5.3 Other Operating Modes

In other operating modes, the counter operation start timing is triggered only upon detection of an effective edge of TAUDTTINm. Once the counter starts, the value of data register is also loaded. The count clock cycles, which is irrelevant to start of counter operation, determine the frequency with which all operations take place.

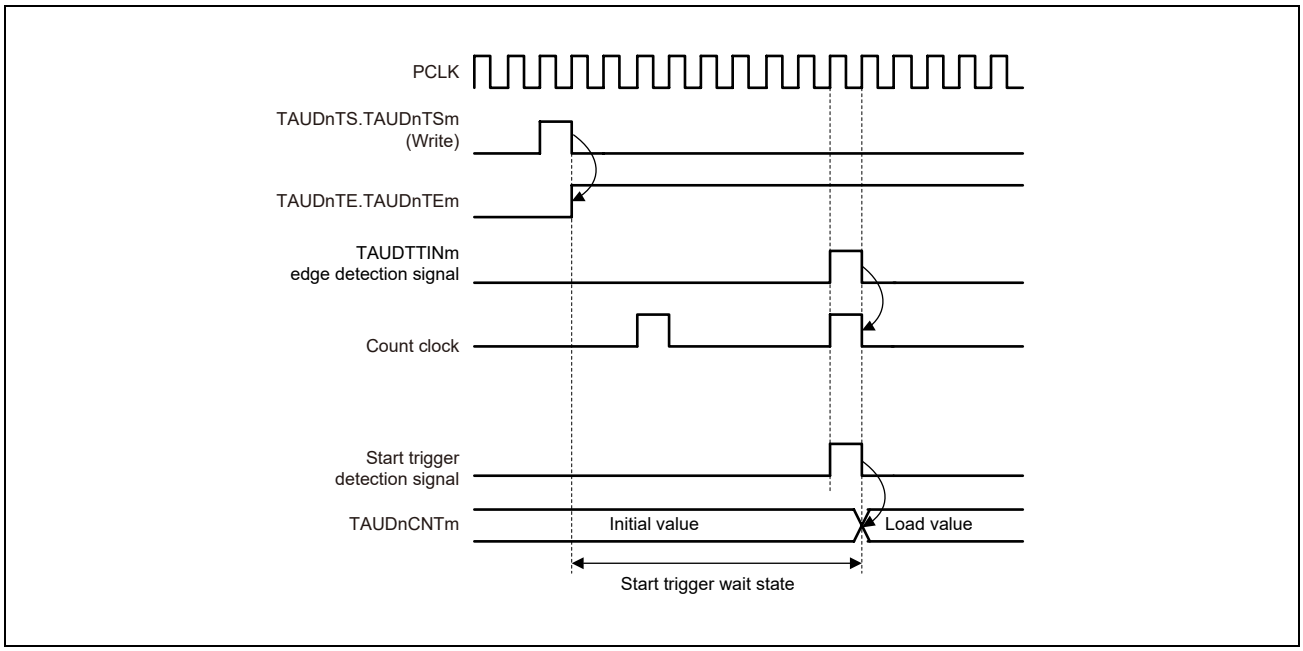


Figure 18.15 Start Timing in Other Operating Modes

18.4.6 TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts

When the counter starts, it is possible to specify whether an INTTAUDnIm is generated using the TAUDnCMOR.TAUDnMD0 bit. The effect of the bit depends on the selected mode, as shown in **Table 18.44**. The effects of INTTAUDnIm on TAUDTTOUTm depend on the selected channel operation function.

Table 18.44 Effect of TAUDnCMORm.TAUDnMD0 Bit on Generation of INTTAUDnIm when Counter is Triggered

Mode	TAUDnCMORm.TAUDnMD0 Bit	INTTAUDnIm Generated when Counter Starts
Interval timer mode	0	No
Capture Mode Count Capture Mode	1	Yes
Capture & One Count Mode Capture & Gate Count Mode Event Count Mode Up Down Count Mode	0	No
One Count Mode	0/1	No, regardless of setting of TAUDnCMORm.TAUDnMD0 bit.
Pulse One Count Mode	0/1	Yes, regardless of setting of TAUDnCMORm.TAUDnMD0 bit.

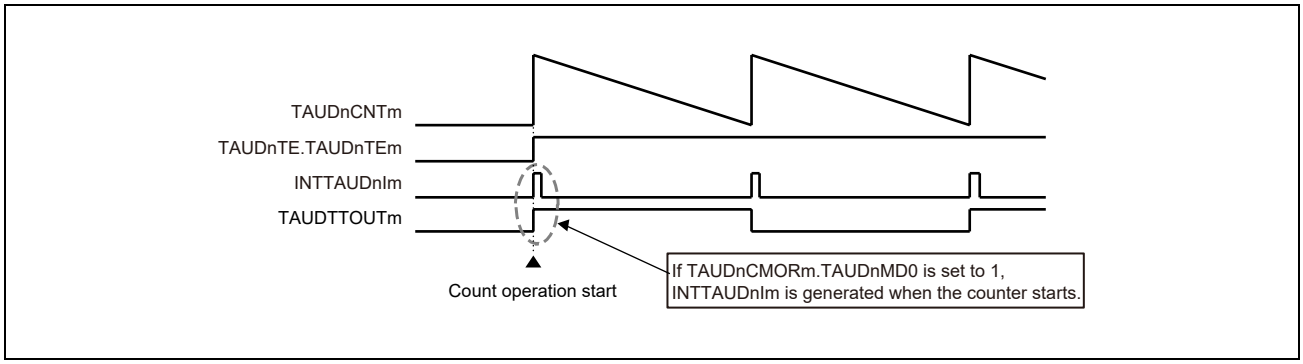


Figure 18.16 INTTAUDnIm Generated when Counter Starts

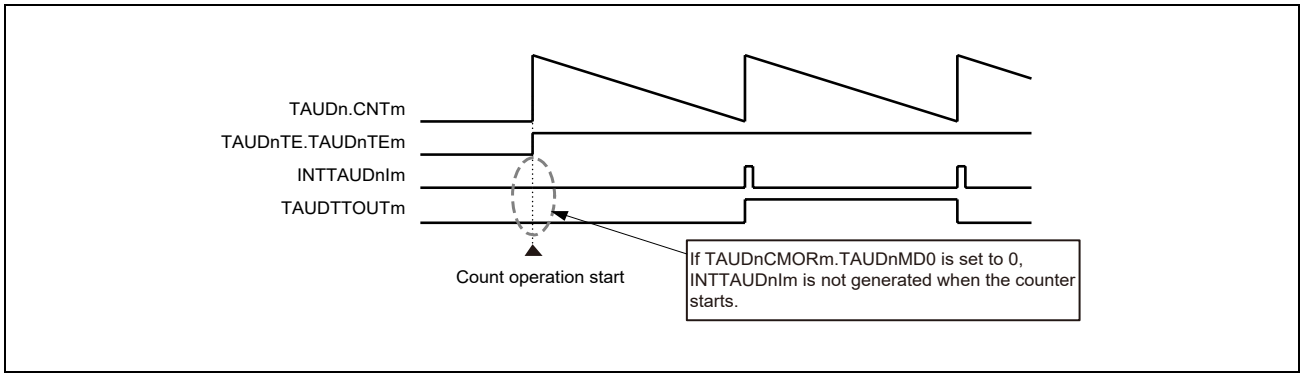


Figure 18.17 INTTAUDnIm not Generated when Counter Starts

18.4.7 Interrupt Generation upon Overflow

Certain independent functions that count up, overflow without generating an interrupt when they reach $FFFF_H$. This section describes how it is possible to generate an interrupt, by combining a channel operating in one of these modes with a channel in a different operation mode which counts down.

The appropriate operation mode for the second channel depends on the operation mode of the first channel. Nevertheless, the principle is the same for all combinations:

- Find an operation mode for the second channel that counts down in such a manner, that it reaches 0000_H at the same time as the first channel overflows (TAUDnCNTm = $FFFF_H$).
- Set TAUDnCDRm of the second channel to $FFFF_H$.
- The two channels must count at the same speed (i.e. they must have the same count clock).

Result:

The down-counter of the second channel reaches 0000_H at exactly the same time as the up-counter of the first channel overflows (TAUDnCNTm = $FFFF_H$). Thus the second channel generates the desired interrupt.

The following sections list the operating modes that count down that are required to match specific operating modes that count up, as well as example timing diagrams.

18.4.7.1 Count Capture Mode

Applies to

- TAUDTTINm Input Position Detection Function

Combine with

Interval Timer Mode

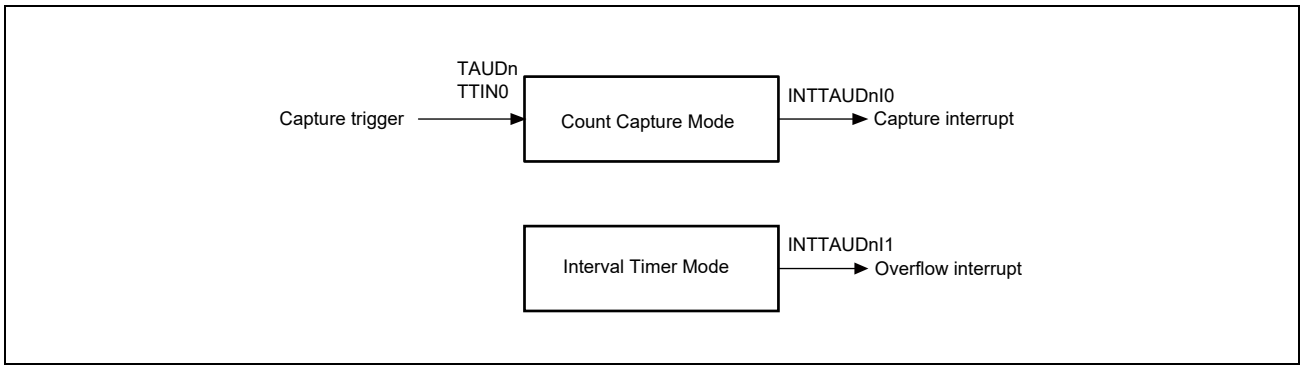


Figure 18.18 Combination of Count Capture Mode and Interval Timer Mode

Timing diagram

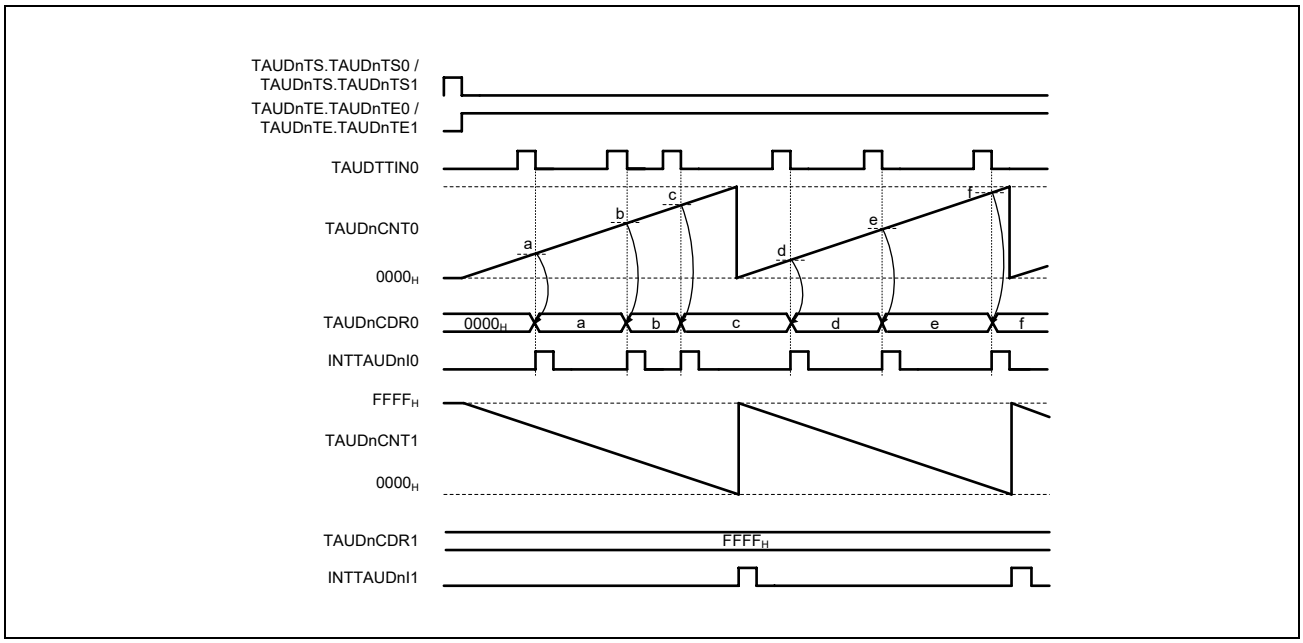


Figure 18.19 Interrupt Generation via Combination of Count Capture Mode and Interval Timer Mode

18.4.8 TAUDTTINm Edge Detection

Edge detection is based on the operation clock. This means that an edge can only be detected at the next rising edge of the operation clock. This can lead to a maximum delay of one operation clock cycle.

Figure 18.20 shows when edge detection takes place.

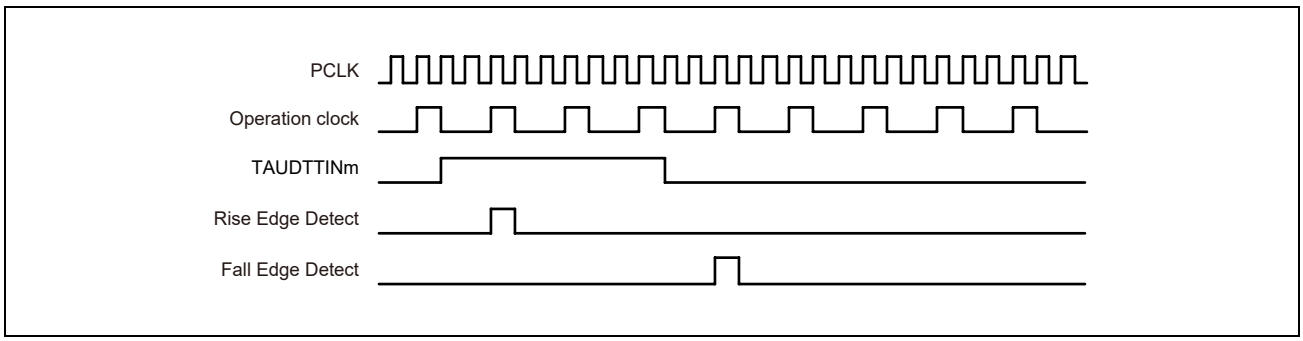


Figure 18.20 Basic Edge Detection Timing

Figure 18.20 shows an operation timing image. Actually, a noise filter or synchronization circuit which is located between the TAUDnlm pin and TAUDn causes a delay time.

18.4.9 Independent Channel Operation Functions

The following sections list the independent channel operation functions provided by the Timer Array Unit D. For a general overview of independent channel operation, see **Section 18.2, Overview**.

This section describes functions that generate interrupts at regular intervals or with a specified delay.

- **Section 18.4.9.1, Interval Timer Function**
- **Section 18.4.9.2, TAUDTTINm Input Interval Timer Function**
- **Section 18.4.9.3, Clock Divide Function**
- **Section 18.4.9.4, External Event Count Function**
- **Section 18.4.9.5, Delay Count Function**
- **Section 18.4.9.6, One-Pulse Output Function**
- **Section 18.4.9.7, TAUDTTINm Input Pulse Interval Measurement Function**
- **Section 18.4.9.8, TAUDTTINm Input Signal Width Measurement Function**
- **Section 18.4.9.9, TAUDTTINm Input Position Detection Function**
- **Section 18.4.9.10, TAUDTTINm Input Period Count Detection Function**
- **Section 18.4.9.11, TAUDTTINm Input Pulse Interval Judgment Function**
- **Section 18.4.9.12, TAUDTTINm Input Signal Width Judgment Function**

18.4.9.1 Interval Timer Function

(1) Overview

Summary

This function is used as a reference timer for generating timer interrupts (INTTAUDnIm) at regular intervals. When an interrupt is generated, the TAUDTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operation mode must be set to Interval Timer Mode, refer to **Table 18.45, Contents of TAUDnCMORm Register for Interval Timer Function**.
- The channel output mode must be set to Independent Channel Output Mode 1, refer to **Section 18.4.4, Channel Output Modes**.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is written to TAUDnCNTm and the counter starts to count down from this value.

When the counter reaches 0000_H, INTTAUDnIm is generated and the TAUDTTOUTm signal toggles. TAUDnCNTm then reloads the TAUDnCDRm value and subsequently continues to operate.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1, which in turn sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm stop but retain their values. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. The counter can also be forcibly restarted (without stopping it first) by setting TAUDnTS.TAUDnTSm to 1 during operation.

Conditions

If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not generated, and therefore TAUDTTOUTm does not toggle. This results in an inverted TAUDTTOUTm signal compared to when TAUDnCMORm.TAUDnMD0 is set to 1. For details refer to **Section 18.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

(2) Equations

INTTAUDnIm cycle = count clock cycle × (TAUDnCDRm + 1)

TAUDTTOUTm square wave cycle = count clock cycle × (TAUDnCDRm + 1) × 2

(3) Block Diagram and General Timing Diagram

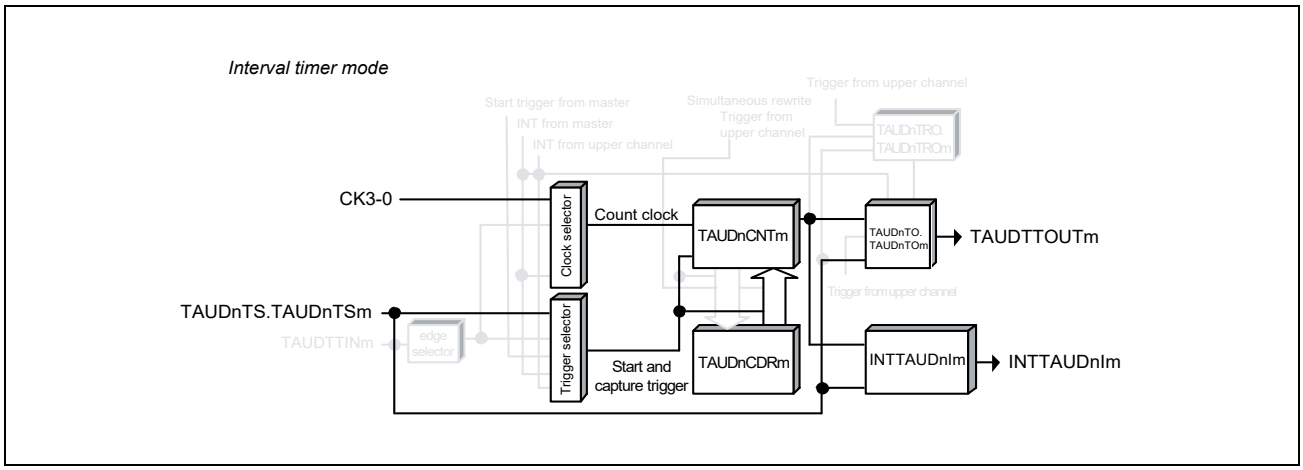


Figure 18.21 Block Diagram of Interval Timer Function

The following settings apply to the general timing diagram.

- INTTAUDnIm is generated at the beginning of operation (TAUDnCMORm.TAUDnMD0 = 1)

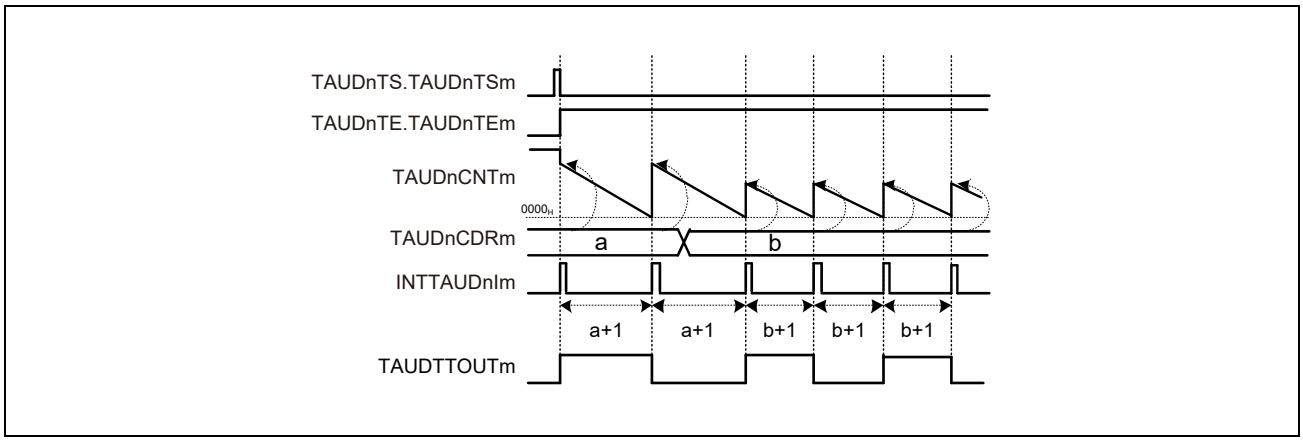


Figure 18.22 General Timing Diagram of Interval Timer Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.45 Contents of TAUDnCMORm Register for Interval Timer Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the sampling clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUDnCCS[1:0]	00: Uses the sampling clock as a counter clock.
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Triggers the counter by software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated to toggle TAUDTTOUTm at the beginning of an operation. 1: INTTAUDnIm is generated to toggle TAUDTTOUTm at the beginning of an operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.46 Contents of TAUDnCMURm Register for Interval Timer Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	Unused. Set to 00.

(c) Channel output mode

Table 18.47 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

NOTE

The channel output mode can also be set to Channel Output Mode Controlled by Software by setting TAUDnTOE.TAUDnTOEm = 0. TAUDTTOUTm can then be controlled independently of the interrupts. For details, see **Section 18.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the interval timer function. Therefore, these registers should be set to 0.

Table 18.48 Simultaneous Rewrite Settings for Interval Timer Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for Interval Timer Function

Table 18.49 Operating Procedure for Interval Timer Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.45, Contents of TAUDnCMORm Register for Interval Timer Function**, and **Table 18.46, Contents of TAUDnCMURm Register for Interval Timer Function**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 18.47, Control Bit Settings in Independent Channel Output Mode 1**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. The TAUDnCDRm value is loaded in TAUDnCNTm. When TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm is generated and TAUDTTOUTm toggles.
During Operation	The TAUDnCDRm register value can be changed at any time. The TAUDnCNTm register can be read at all times.	TAUDnCNTm counts down. When the counter reaches 0000_H: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated and TAUDTTOUTm toggles.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(6) Specific Timing Diagrams

(a) TAUDnCDRm = 0000_H, count clock = PCLK/2

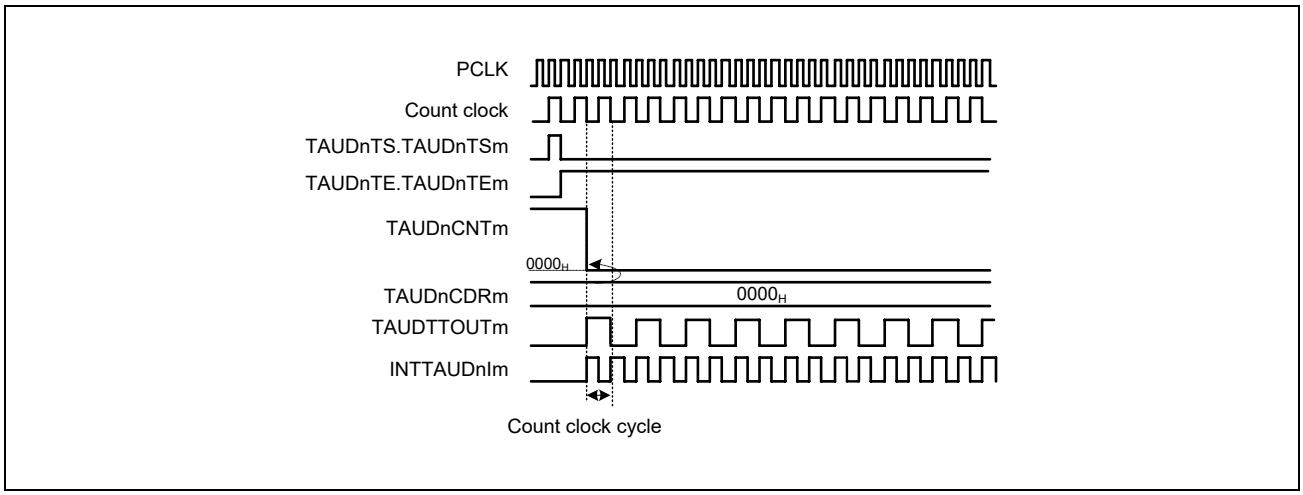


Figure 18.23 TAUDnCDRm = 0000_H, Count Clock = PCLK/2

- If TAUDnCDRm = 0000_H and the count clock = PCLK/2, the TAUDnCDRm value is loaded into TAUDnCNTm every count clock, meaning that TAUDnCNTm is always 0000_H.
- INTTAUDnIm is generated every count clock, resulting in TAUDTTOUTm toggling every count clock.

(b) TAUDnCDRm = 0000_H, count clock = PCLK

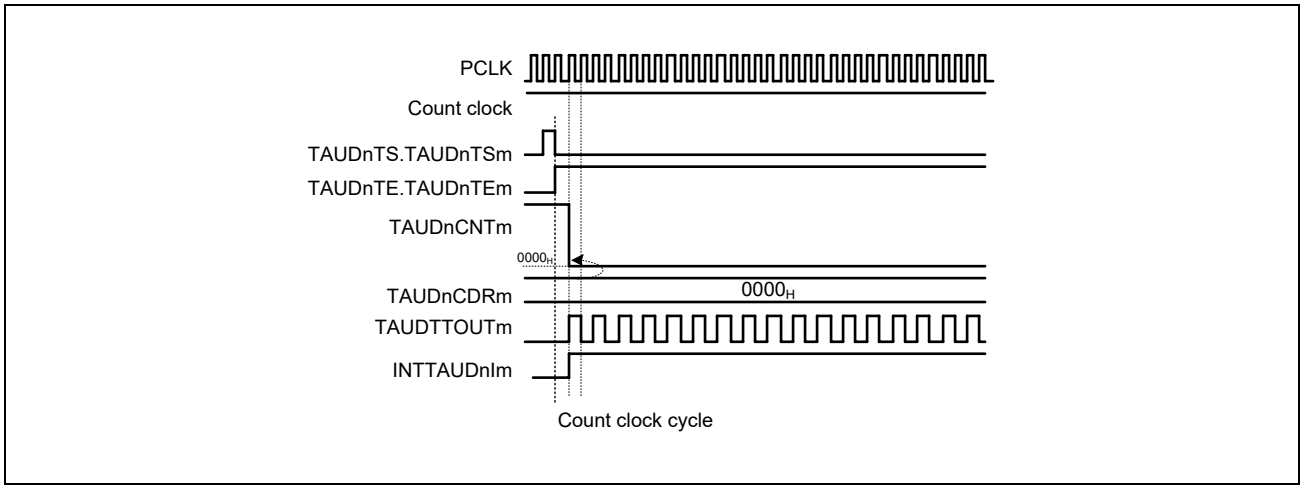


Figure 18.24 TAUDnCDRm = 0000_H, Count Clock = PCLK

- If TAUDnCDRm = 0000_H and the count clock = PCLK, the TAUDnCDRm value is loaded into TAUDnCNTm every PCLK clock, meaning that TAUDnCNTm is always 0000_H.
- INTTAUDnIm is generated continuously, resulting in TAUDTTOUTm toggling every PCLK clock.

(c) Operation stop and restart

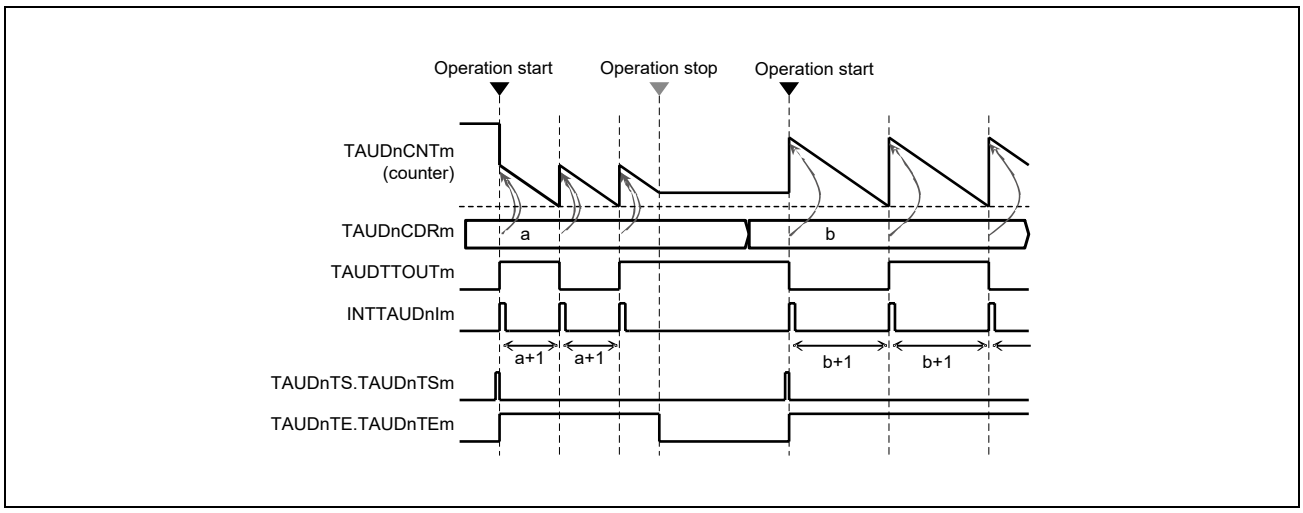


Figure 18.25 Operation Stop and Restart (TAUDnCMORm.TAUDnMD0 = 1)

- The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm and TAUDTTOUTm stop but retain their values.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1.

(d) Forced restart (TAUDnCMORm.TAUDnMD0 = 1)

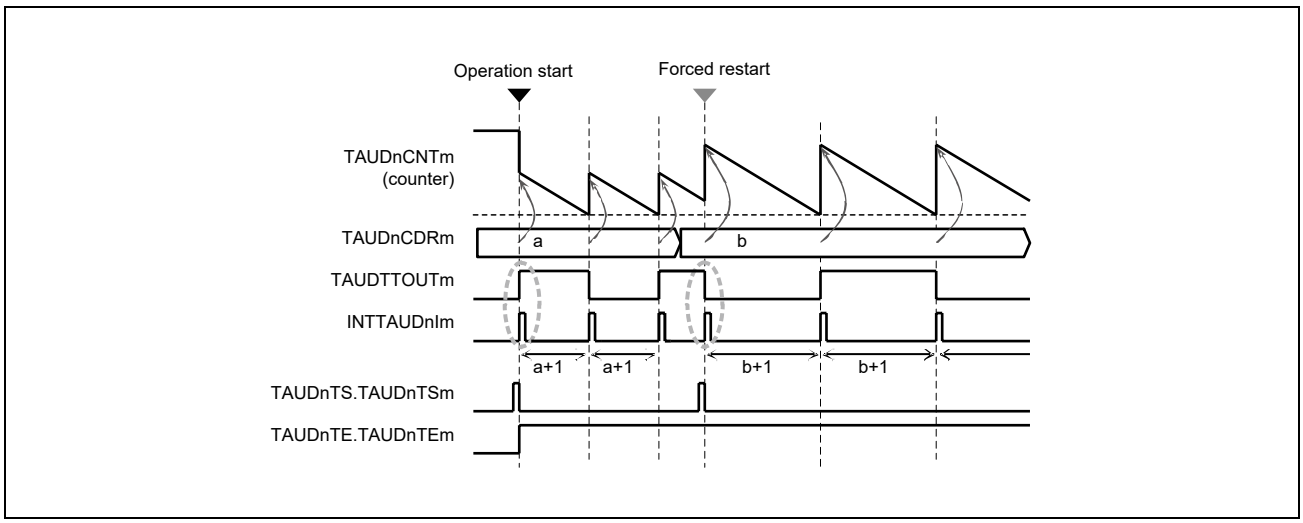


Figure 18.26 Forced Restart Operation (TAUDnCMORm.TAUDnMD0 = 1)

- The counter can be forcibly restarted (without stopping it first) by setting TAUDnTS.TAUDnTSm to 1 during operation.
- If the TAUDnCMORm.TAUDnMD0 bit is set to 1, the first interrupt after a start or restart is generated.
- When a forced restart is made, the TAUDnCDRm value is reflected to TAUDnCNTm and counting starts. Execute a forced restart to reflect the changed TAUDnCDRm value immediately.
- When a forced restart is made, an interrupt (INTTAUDnlm) is generated and TAUDTTOUTm is inverted.

(e) Forced restart (TAUDnCMORm.TAUDnMD0 = 0)

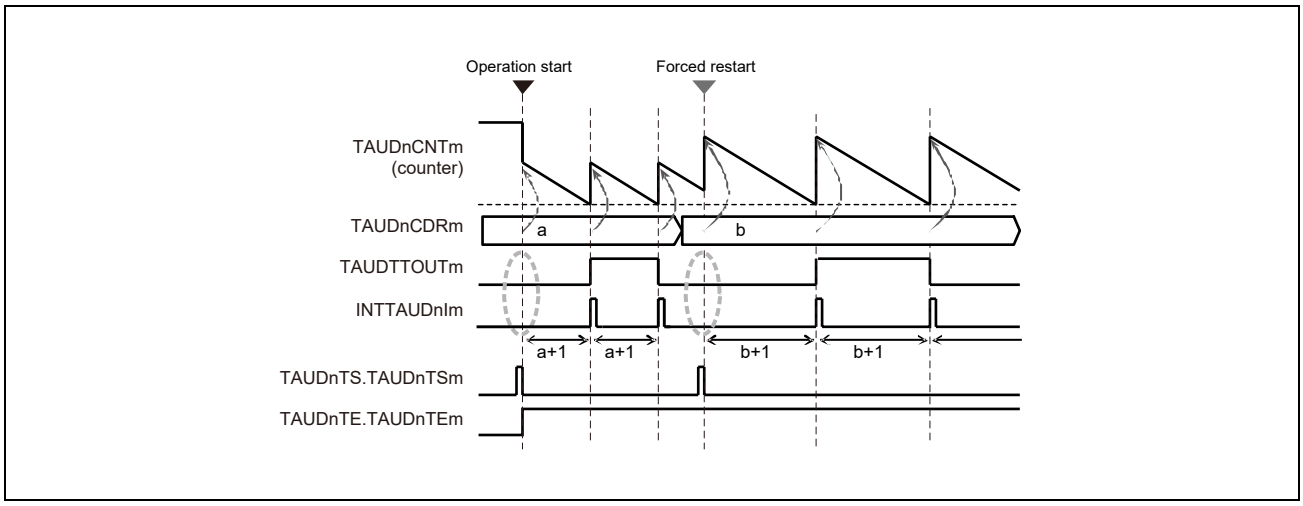


Figure 18.27 Forced Restart Operation (TAUDnCMORm.TAUDnMD0 = 0)

- When a forced restart is made, an interrupt (INTTAUDnlm) is not generated and TAUDTTOUTm is not inverted.

18.4.9.2 TAUDTTINm Input Interval Timer Function

(1) Overview

Summary

This function is used as a reference timer for generating timer interrupts (INTTAUDnIm) at regular intervals or when an effective TAUDTTINm input edge is detected. When an interrupt is generated, the TAUDTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operating mode should be set to interval timer mode. See **Table 18.50, Contents of TAUDnCMORm Register for TAUDTTINm Input Interval Timer Function**.
- The channel output mode should be set to independent channel output mode 1. See **Section18.4.4, Channel Output Modes**.

Functional description

This function operates in an identical manner to the interval timer function (see **Section 18.4.9.1,** Interval Timer Function) except that this function is restarted by an effective TAUDTTINm input edge. The type of edge used as a trigger is specified using the TAUDnCMURm.TAUDnTIS[1:0] bits. Either rising edge, falling edge, or rising and falling edge can be selected.

(2) Equations

INTTAUDnIm cycle = count clock cycle × (TAUDnCDRm + 1)

TAUDTTOUTm square wave cycle = count clock cycle × (TAUDnCDRm + 1) × 2

(3) Block Diagram and General Timing Diagram

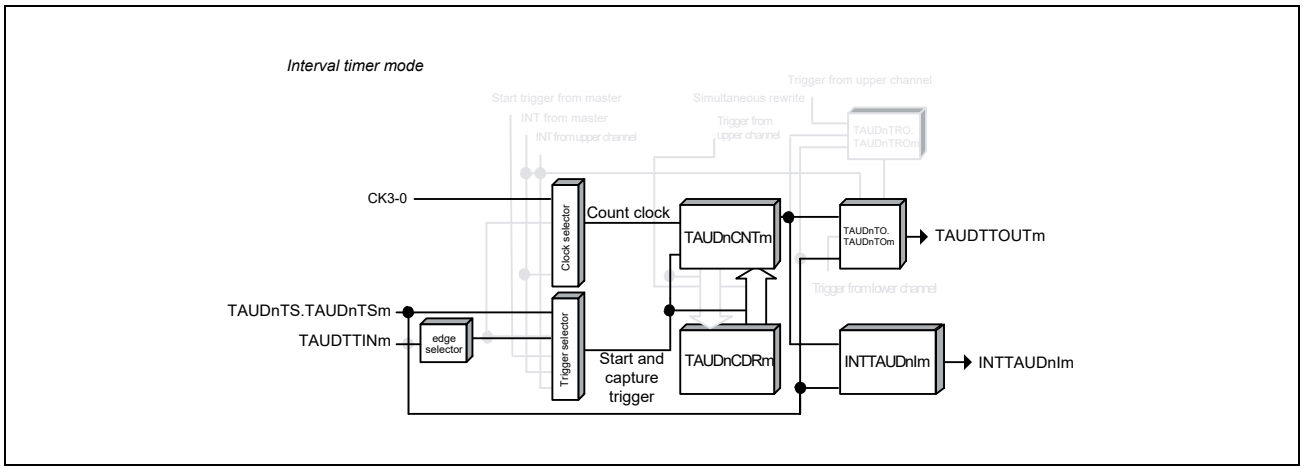


Figure 18.28 Block Diagram of TAUDTTINm Input Interval Timer Function

The following settings apply to the general timing diagram.

- INTTAUDnIm is generated at the beginning of operation (TAUDnCMORm.TAUDnMD0 = 1)
- Rising edge detection (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

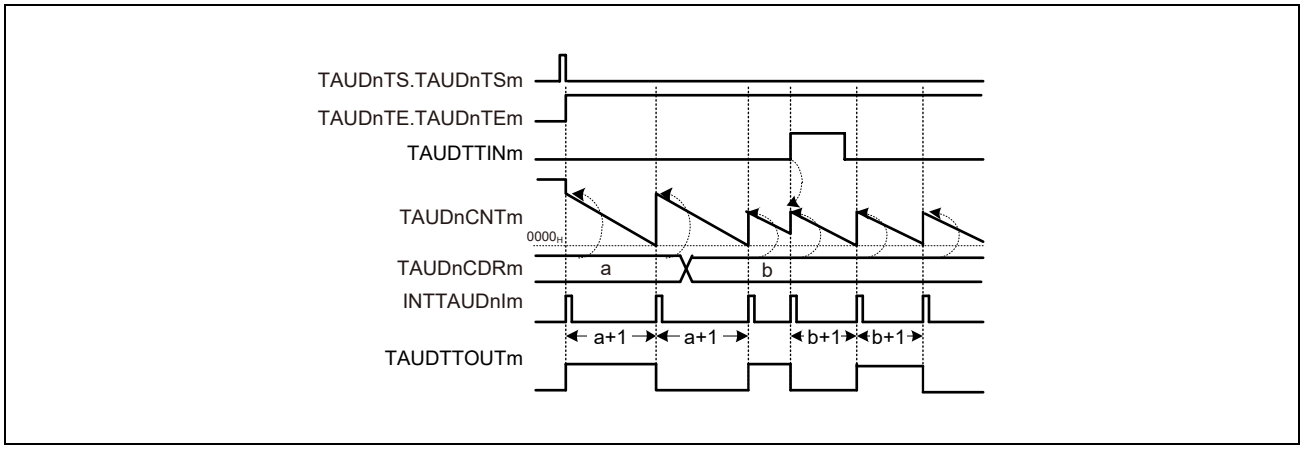


Figure 18.29 General Timing Diagram of TAUDTTINm Input Interval Timer Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.50 Contents of TAUDnCMORm Register for TAUDTTINm Input Interval Timer Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as an external start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated to toggle TAUDTTOUTm at the beginning of an operation. 1: INTTAUDnIm is generated to toggle TAUDTTOUTm at the beginning of an operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.51 Contents of TAUDnCMURm Register for TAUDTTINm Input Interval Timer Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

Table 18.52 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Setting prohibited

NOTE

The channel output mode can also be set to Channel Output Mode Controlled by Software by setting TAUDnTOE.TAUDnTOEm = 0. TAUDTTOUTm can then be controlled independently of the interrupts.

For details, see **Section 18.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the TAUDTTINm Input Interval Timer Function. Therefore, these registers should be set to 0.

Table 18.53 Simultaneous Rewrite Settings for TAUDTTINm Input Interval Timer Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Interval Timer Function

Table 18.54 Operating Procedure for TAUDTTINm Input Interval Timer Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.50 Contents of TAUDnCMORm Register for TAUDTTINm Input Interval Timer Function**, and **Table 18.51 Contents of TAUDnCMURm Register for TAUDTTINm Input Interval Timer Function**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 18.52, Control Bit Settings in Independent Channel Output Mode 1**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. The TAUDnCDRm value is loaded in TAUDnCNTm. When TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm is generated and TAUDTTOUTm toggles.
During Operation	The values of the TAUDnCMURm.TAUDnTIS[1:0] and the TAUDnCDRm register are changeable at any time. The TAUDnCNTm register can be read at all times. Detection of TAUDTTINm edge	TAUDnCNTm counts down. When the counter reaches 0000_H: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated and TAUDTTOUTm toggles. When an effective TAUDTTINm input edge is detected during count operation, the TAUDnCDRm value is loaded in TAUDnCNTm and count operation continues. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(6) Specific Timing Diagrams

The timing diagrams in **Section 18.4.9.1, Interval Timer Function** apply, and in addition the counter can also be restarted by an effective TAUDTTINm input edge.

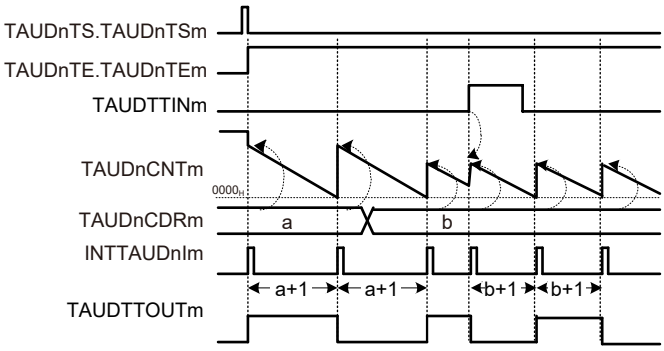


Figure 18.30 Counter Triggered by Rising TAUDTTINm Input Edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B), TAUDnCMORm.TAUDnMD0 = 1

- If an effective TAUDTTINm input edge is detected, an interrupt is generated which causes TAUDTTOUTm to toggle. In this example, the effective edge is a rising edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B).

18.4.9.3 Clock Divide Function

(1) Overview

Summary

This function is used as a frequency divider. The frequency of the input signal TAUDTTINm is divided by a factor related to TAUDnCDRm, and the resulting signal is output to TAUDTTOUTm.

Prerequisites

- TAUDTTINm should have a fixed frequency.
- The operating mode should be set to interval timer mode (see **Table 18.55, Contents of TAUDnCMORm Register for Clock Divide Function**).
- The channel output mode should be set to independent channel output mode 1 (see **Section 18.4.4, Channel Output Modes**).

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from this value, using TAUDTTINm as a count clock.

When the counter value reaches 0000_H, INTTAUDnIm occurs and TAUDTTOUTm signal is toggled. Then, TAUDnCDRm value is loaded into TAUDnCNTm to continue operation subsequently.

The value of TAUDnCDRm can be rewritten at any time. The changed value of TAUDnCDRm is applied when the counter starts to count down next time.

The counter can be stopped by setting TAUDnTT.TAUDnTTm = 1. This sets TAUDnTE.TAUDnTEm = 0. TAUDnCNTm and TAUDTTOUTm stop but retain their values. The function can be restarted by setting TAUDnTS.TAUDnTSm = 1. The counter can also be forcibly restarted without making a stop by setting TAUDnTS.TAUDnTSm = 1 during operation (forced restart).

Conditions

If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not generated, and therefore TAUDTTOUTm does not toggle. This results in an inverted TAUDTTOUTm signal compared to when TAUDnCMORm.TAUDnMD0 is set to 1. For details, see **Section 18.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

NOTE

TAUDTTINm input signals are sampled at the frequency of the operation clock set by TAUDnCMORm.TAUDnCKS[1:0] bits. Therefore, the TAUDTTOUTm output clock cycle has an error of ± 1 operation clock cycle.

(2) Equations

- When rising edge detection is selected:
  TAUDTTOUTm frequency = TAUDTTINm frequency/[(TAUDnCDRm + 1) × 2]
- When falling edge detection is selected:
  TAUDTTOUTm frequency = TAUDTTINm frequency/[(TAUDnCDRm + 1) × 2]
- When falling and rising edge detection is selected:
  TAUDTTOUTm frequency = TAUDTTINm frequency/(TAUDnCDRm + 1)

(3) Block Diagram and General Timing Diagram

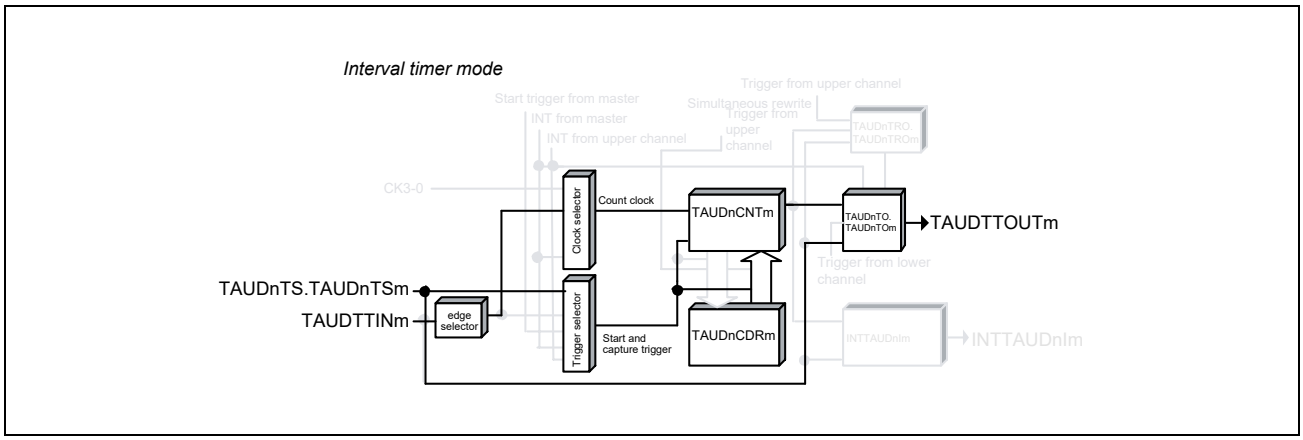


Figure 18.31 Block Diagram of Clock Divide Function

The following settings apply to the general timing diagram.

- INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)
- Detection of rising edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

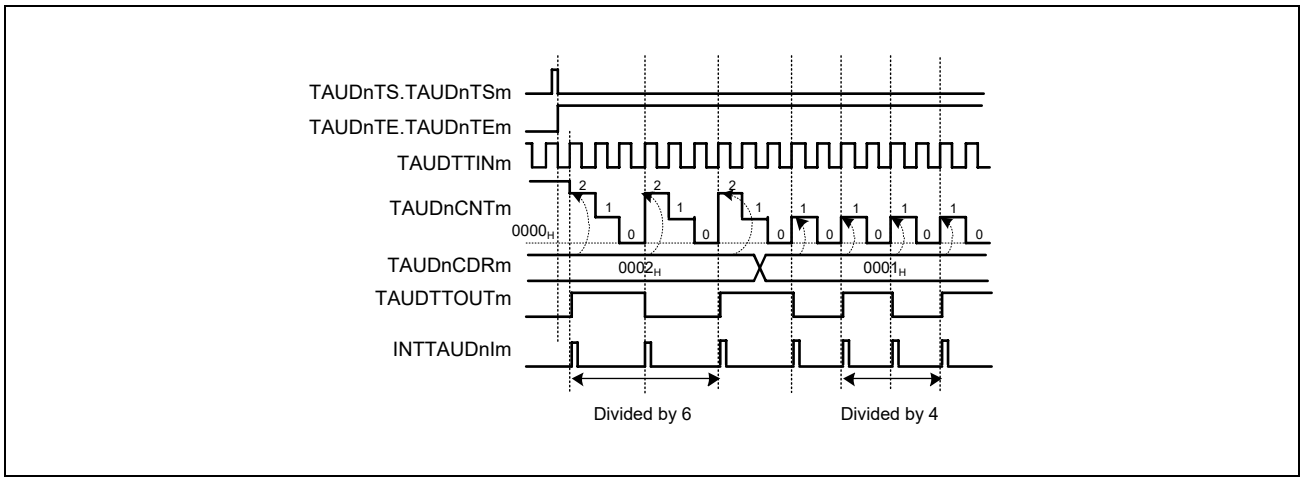


Figure 18.32 General Timing Diagram of Clock Divide Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.55 Contents of TAUDnCMORm Register for Clock Divide Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	01: An effective TAUDTTINm input edge is used as a count clock.
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated to toggle TAUDTTOUTm at the beginning of an operation. 1: INTTAUDnIm is generated and TAUDTTOUTm is toggled at the beginning of operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.56 Contents of TAUDnCMURm Register for Clock Divide Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

Table 18.57 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.58 Simultaneous Rewrite Settings for Clock Divide Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for Clock Divide Function

Table 18.59 Operating Procedure for Clock Divide Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.55, Contents of TAUDnCMORm Register for Clock Divide Function**, and **Table 18.56, Contents of TAUDnCMURm Register for Clock Divide Function**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 18.57, Control Bit Settings in Independent Channel Output Mode 1**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCNTm loads TAUDnCDRm value. If TAUDnCMORm.TAUDnMD0 is set to 1, INTTAUDnIm occurs and TAUDTTOUTm is toggled.
During Operation	The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at all times.	TAUDnCNTm counts down each time TAUDTTINm input edge is detected. When the counter reaches 0000_H: • TAUDnCDRm value is loaded in TAUDnCNTm and count operation continues. • INTTAUDnIm is generated. • TAUDTTOUTm is toggled. Afterwards, this procedure is repeated.
Stop Operations	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm and TAUDTTOUTm retain their current values.

Restart

(6) Specific Timing Diagrams

(a) TAUDnCDRm = 0000_H

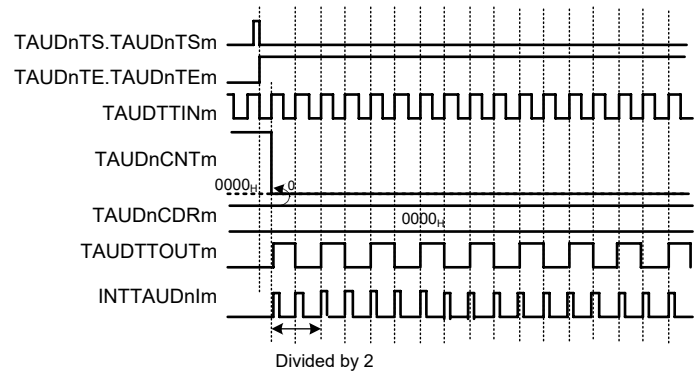


Figure 18.33 TAUDnCDRm = 0000_H, TAUDnCMORm.TAUDnMD0 = 1, TAUDnCMURm.TAUDnTIS[1:0] = 01_B

- If TAUDnCDRm is 0000_H, TAUDnCNTm is always 0000_H.
- INTTAUDnIm is generated every count clock, resulting in TAUDTTOUTm toggling every count clock.

Figure 18.33 shows an operation timing example. Actually, there is a delay from TINm detection until TOUTm output because of the delay time of a noise filter or synchronization circuit placed between the TAUDnlm pin and TAUDn.

(b) Restart

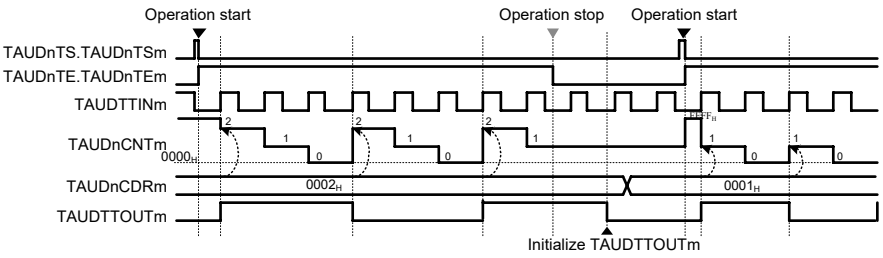


Figure 18.34 Restart (TAUDnCMORm.TAUDnMD0 = 1, TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

To reset the value of TAUDTTOUTm:

- Set TAUDnTOE.TAUDnTOEm = 0 when the counter is stopped (TAUDnTE.TAUDnTEm = 0).
- Then, write either 0 or 1 to TAUDTO.TAUDnTOm to set the new start value of TAUDTTOUTm.

(c) Forced restart

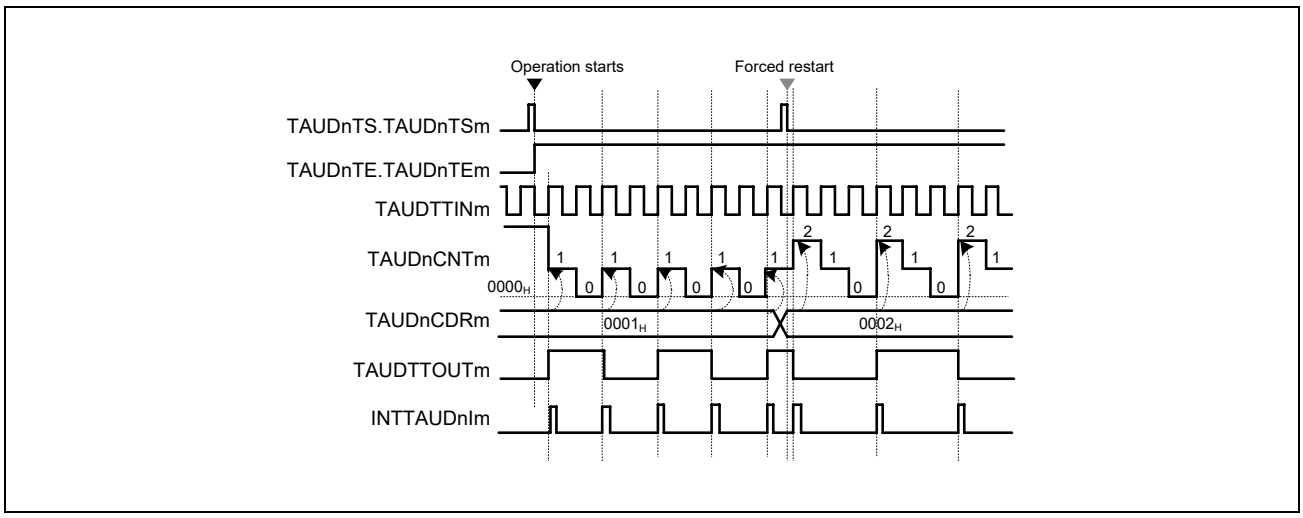


Figure 18.35 Forced Restart Operation (TAUDnCMORm.TAUDnMD0 = 1, TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

- The counter can be forcibly restarted (without stopping it first) by setting TAUDnTS.TAUDnTSm = 1 during operation.
- The value of TAUDnCDRm is written to TAUDnCNTm and the count operation restarts.
- TAUDTTOUTm restarts at the same level as before the forced restart.

18.4.9.4 External Event Count Function

(1) Overview

Summary

This function is used as an event timer, which generates an interrupt (INTTAUDnIm) when a specific number of TAUDTTINm input pulses has occurred.

Prerequisites

- The operating mode should be set to the event count mode (see **Table 18.60, Contents of TAUDnCMORm Register for External Event Count Function**).
- TAUDTTOUTm is not used with this function.

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. When the counter starts, the current value of TAUDnCDRm is loaded into TAUDnCNTm.

When an effective TAUDTTINm input edge is detected, the value of TAUDnCNTm decrements by 1. TAUDnCNTm retains this value until an effective TAUDTTINm input edge is detected or the counter is restarted.

When the effective edge is detected for the (TAUDnCDRm + 1) times, INTTAUDnIm is generated. Then, TAUDnCDRm value is loaded into TAUDnCNTm to continue operation subsequently.

The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. The counter can also be restarted without stopping it first (forced restart) by setting TAUDnTS.TAUDnTSm to 1 during operation.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

An edge type used as a trigger is specified by TAUDnCMURm.TAUDnTIS[1:0] bits.

- When TAUDnCMURm.TAUDnTIS[1:0] = 00_B, falling edges are counted.
- When TAUDnCMURm.TAUDnTIS[1:0] = 01_B, rising edges are counted.
- When TAUDnCMURm.TAUDnTIS[1:0] = 10_B, both edges are counted.

(2) Equations

Number of effective edges detected before INTTAUDnIm generation = TAUDnCDRm + 1

(3) Block Diagram and General Timing Diagram

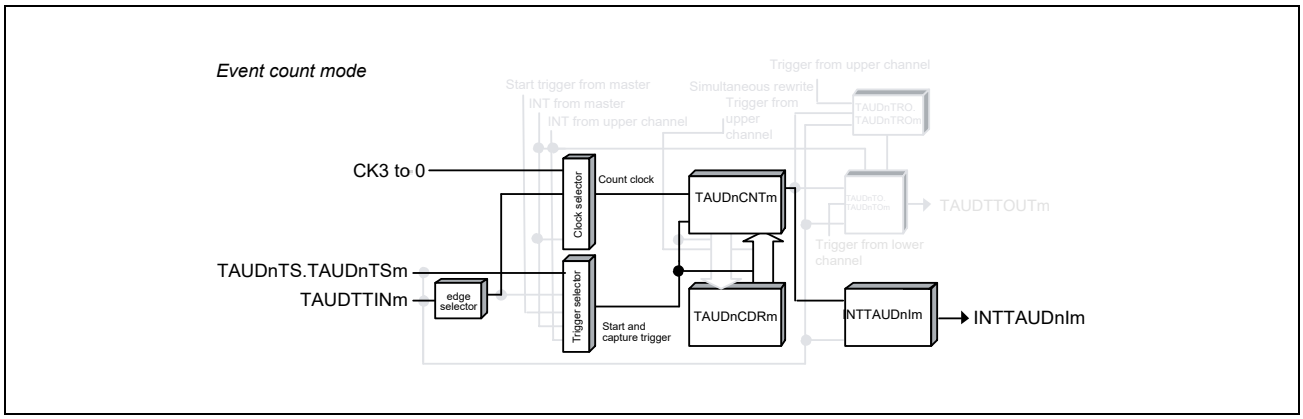


Figure 18.36 Block Diagram of External Event Count Function

The following settings apply to the general timing diagram.

- Detection of rising edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

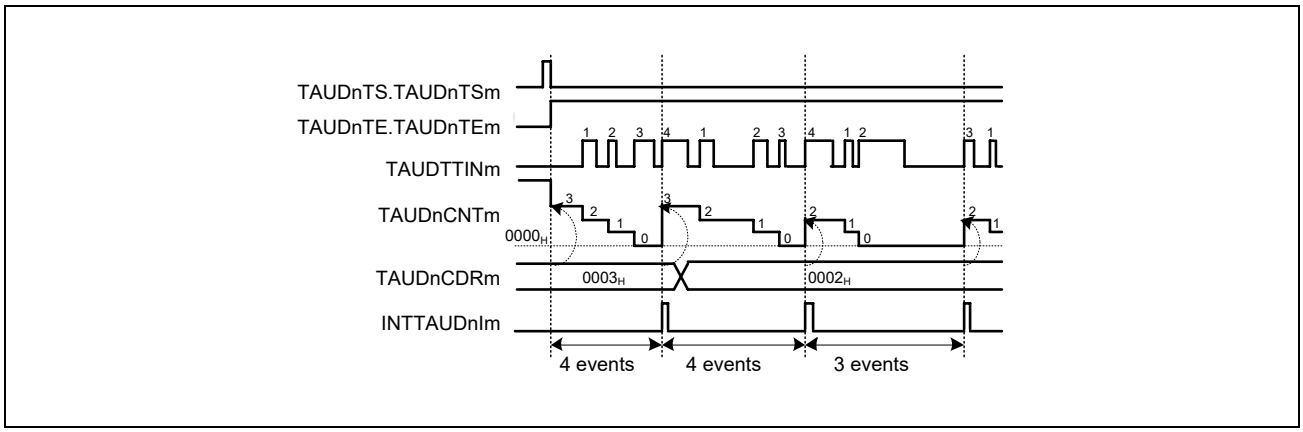


Figure 18.37 General Timing Diagram of External Event Count Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.60 Contents of TAUDnCMORm Register for External Event Count Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	01: An effective TAUDTTINm input edge is used as a count clock.
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0011: Event count mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.61 Contents of TAUDnCMURm Register for External Event Count Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

The channel output mode is not used by this function.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.62 Simultaneous Rewrite Settings for External Event Count Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for External Event Count Function

Table 18.63 Operating Procedure for External Event Count Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.60, Contents of TAUDnCMORm Register for External Event Count Function**, and **Table 18.61, Contents of TAUDnCMURm Register for External Event Count Function**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCNTm loads TAUDnCDRm value and waits for TAUDTTINm input edge detection.
During Operation	Detection of TAUDTTINm edge The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at any time.	TAUDnCNTm counts down each time TAUDTTINm input edge is detected. When the counter reaches 0000_H. When effective edges are detected (TAUDnCDRm + 1) times: • TAUDnCDRm value is loaded in TAUDnCNTm and count operation continues. • INTTAUDnIm is generated. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its current value.

Restart

(6) Specific Timing Diagrams

(a) TAUDnCDRm = 0000_H

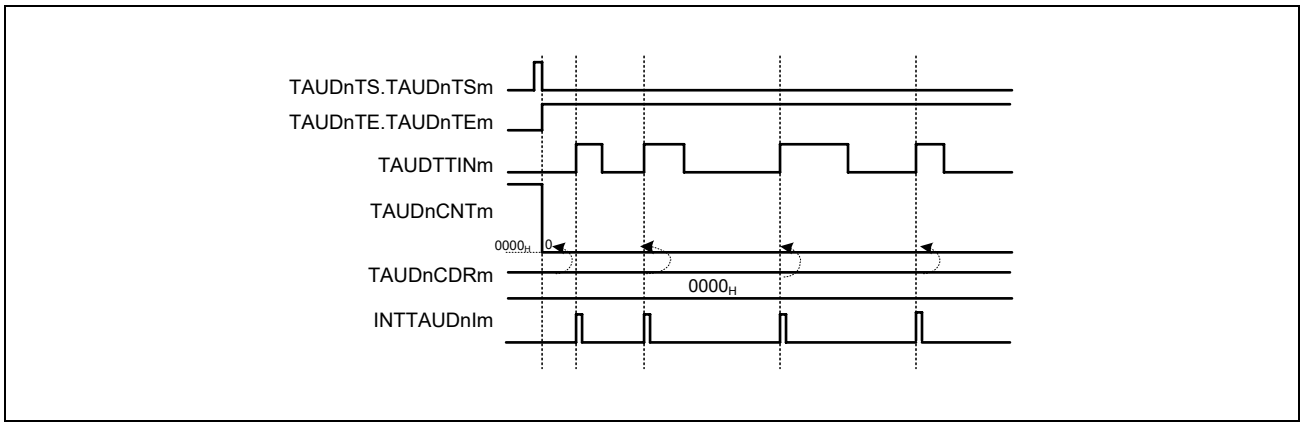


Figure 18.38 TAUDnCDRm = 0000_H, TAUDnCMURm.TAUDnTIS[1:0] = 01_B

- If TAUDnCDRm = 0000_H, 0000_H is loaded into TAUDnCNTm each time an effective TAUDTTINm input edge is detected.
  In other words, INTTAUDnlm occurs each time an effective TAUDTTINm input edge is detected.

(b) Operation stop and restart

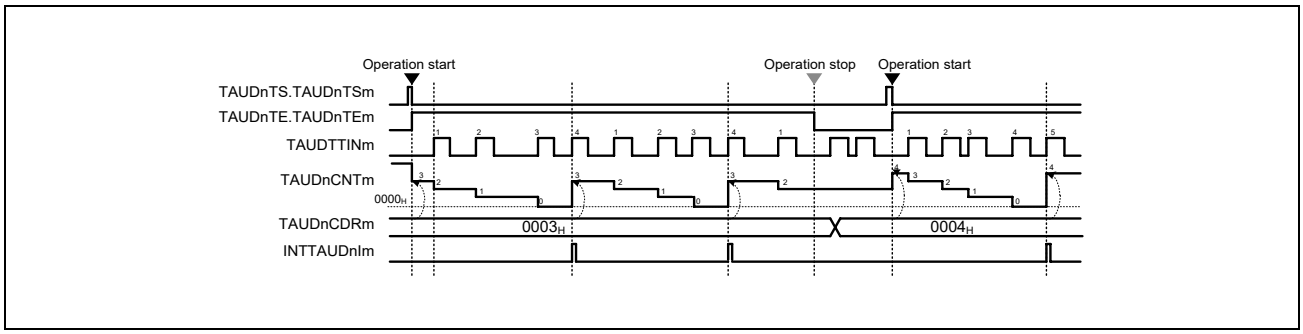


Figure 18.39 Operation Stop and Restart (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

- The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm stops and retains its current value. TAUDTTINm continues and TAUDnCNTm ignores the effective edge.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. TAUDnCNTm loads the TAUDnCDRm value and restarts count operation.

(c) Forced restart

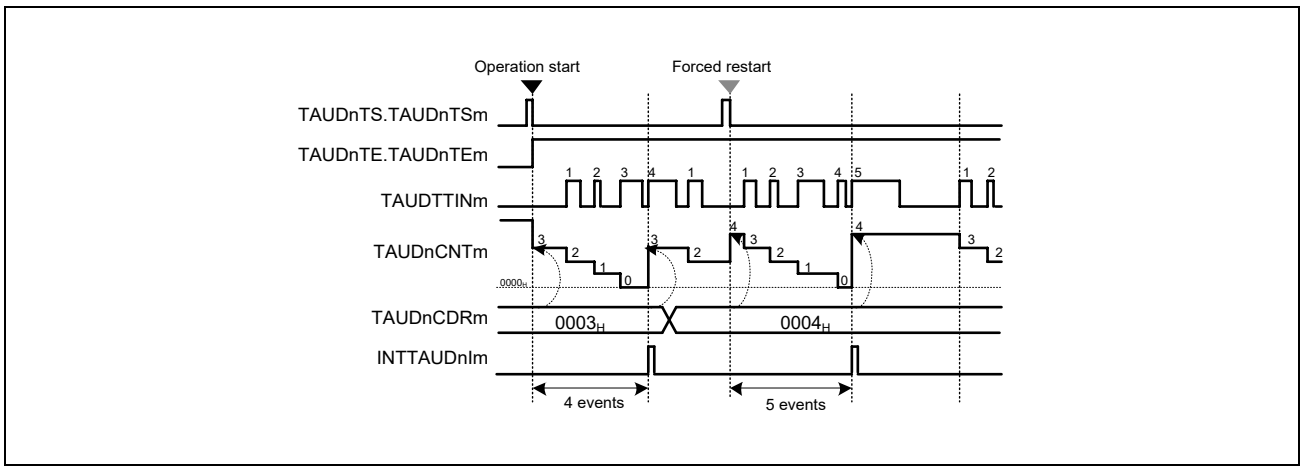


Figure 18.40 Forced Restart Operation (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

Once a forced restart is made, the changed value of TAUDnCDRm is applied to TAUDnCNTm.

- The counter can be restarted without making a stop by setting TAUDnTS.TAUDnTSm to 1 during operation.
- The value of TAUDnCDRm is loaded into TAUDnCNTm and the counter awaits the next effective TAUDTTINm input edge.

18.4.9.5 Delay Count Function

(1) Overview

Summary

This function generates interrupts (INTTAUDnIm), which have a defined delay to the TAUDTTINm input signal. TAUDTTINm input signal pulses that occur within the delay period are ignored.

Prerequisites

- The operating mode should be set to one-count mode. See **Table 18.64, Contents of TAUDnCMORm Register for Delay Count Function**.
- TAUDTTOUTm is not used with this function.
- Trigger detection should be disabled during counting (TAUDnCMORn.TAUDnMD0 = 0).

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation.

The counter starts when an effective TAUDTTINm input start edge is detected. The value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from the TAUDnCDRm value.

When the counter reaches 0000_H, an interrupt is generated. The counter returns to $FFFF_H$ and awaits the next effective TAUDTTINm input edge.

When the counter is counting down, further TAUDTTINm input signals are ignored, i.e., the counter does not reset.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

The type of edge used as a trigger is specified by the TAUDnCMURm.TAUDnTIS[1:0] bits:

- If TAUDnCMURm.TAUDnTIS[1:0] = 00_B, falling edges trigger the counter.
- If TAUDnCMURm.TAUDnTIS[1:0] = 01_B, rising edges trigger the counter.
- If TAUDnCMURm.TAUDnTIS[1:0] = 10_B, rising and falling edges trigger the counter.

(2) Equations

Delay between TAUDTTINm and INTTAUDnIm = count clock cycle × (TAUDnCDRm + 1)

(3) Block Diagram and General Timing Diagram

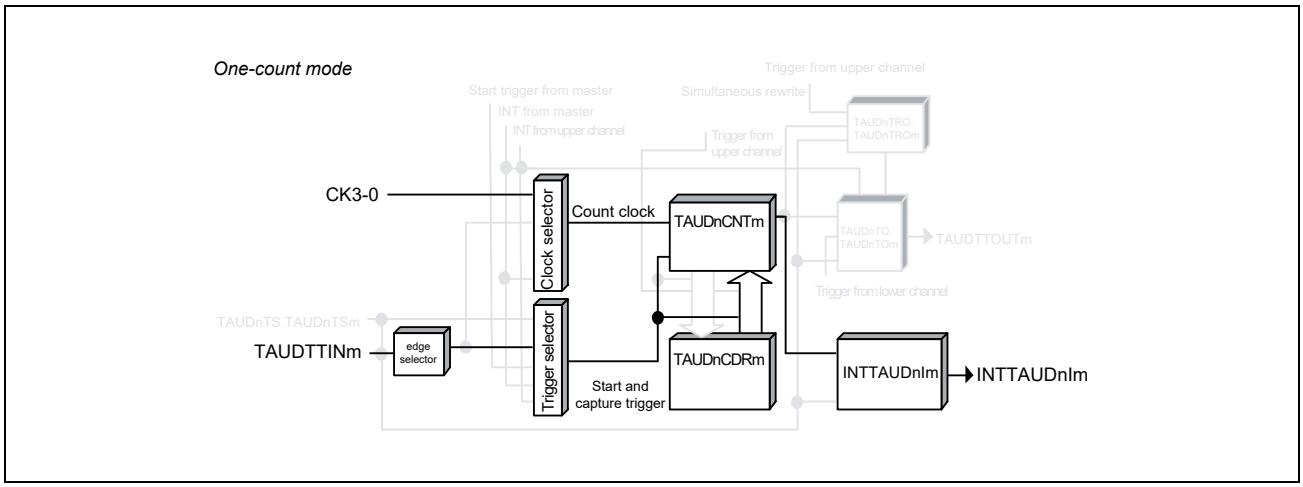


Figure 18.41 Block Diagram of Delay Count Function

The following settings apply to the general timing diagram.

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

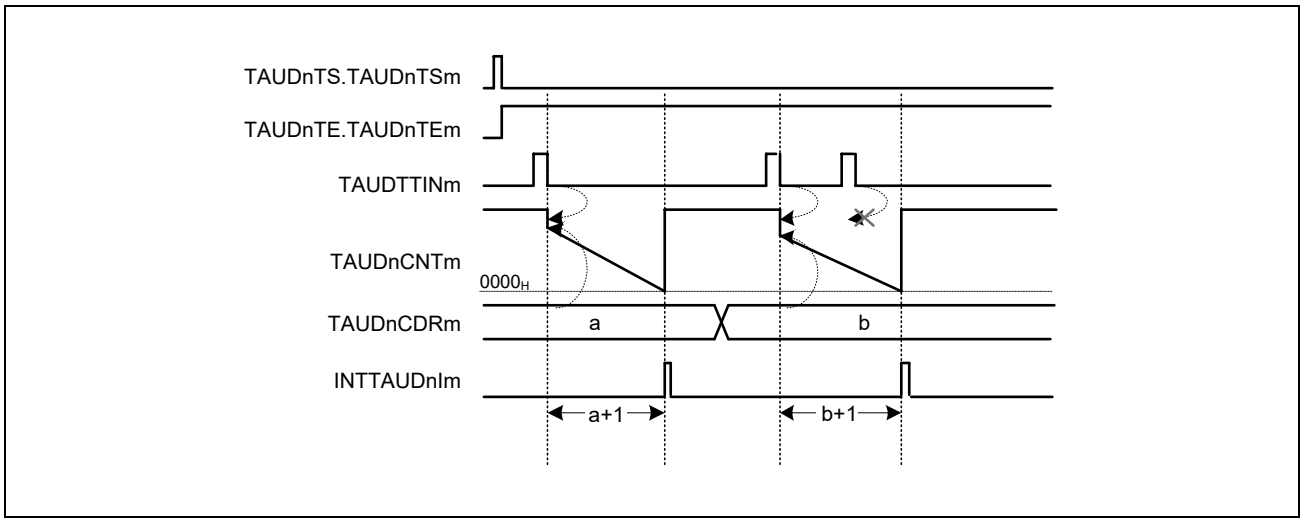


Figure 18.42 General Timing Diagram of Delay Count Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.64 Contents of TAUDnCMORm Register for Delay Count Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as an external start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	0: Disables a start trigger during operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.65 Contents of TAUDnCMURm Register for Delay Count Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.66 Simultaneous Rewrite Settings for Delay Count Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for Delay Count Function

Table 18.67 Operating Procedure for Delay Count Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.64, Contents of TAUDnCMORm Register for Delay Count Function**, and **Table 18.65, Contents of TAUDnCMURm Register for Delay Count Function**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0. Detection of TAUDTTINm start edge	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When a start edge is detected, the TAUDnCDRm value is loaded in TAUDnCNTm.
During Operation	The TAUDnCDRm register value can be changed at any time. The TAUDnCNTm register can be read at all times.	TAUDnCNTm counts down. When the counter reaches 0000_H. INTTAUDnIm is generated. TAUDnCNTm stops counting, returns $FFFF_H$, and waits for a trigger. If a trigger occurs while TAUDnCNTm is counting, the trigger is ignored. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its value.

Restart

18.4.9.6 One-Pulse Output Function

(1) Overview

Summary

This function generates an interrupt (INTTAUDnIm) when an effective TAUDTTINm input edge is detected and at a defined interval afterward. TAUDTTINm input signal pulses that occur within the defined interval are ignored. When an interrupt is generated, the TAUDTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operating mode should be set to pulse one-count mode (see **Table 18.68, Contents of TAUDnCMORm Register for One-Pulse Output Function**).
- The channel output mode should be set to independent channel output mode 1 (see **Section 18.4.4, Channel Output Modes**).
- Trigger detection should be disabled during counting (TAUDnCMORn.TAUDnMD0 = 0).

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation.

The counter starts when an effective TAUDTTINm input edge is detected. The value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from the TAUDnCDRm value. An interrupt is generated and TAUDTTOUTm toggles.

When the counter reaches 0001_H, an interrupt is generated and TAUDTTOUTm is set to the inactive level. The counter stops at 0000_H and awaits the next effective TAUDTTINm input edge.

When the counter is counting down, further TAUDTTINm input signals are ignored, i.e., the counter does not reset.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

The type of edge used as a trigger is specified by the TAUDnCMURm.TAUDnTIS[1:0] bits:

- If TAUDnCMURm.TAUDnTIS[1:0] = 00_B, falling edges trigger the counter.
- If TAUDnCMURm.TAUDnTIS[1:0] = 01_B, rising edges trigger the counter.
- If TAUDnCMURm.TAUDnTIS[1:0] = 10_B, rising and falling edges trigger the counter.

(2) Equations

Interval between TAUDTTINm and INTTAUDnIm
= TAUDTTOUTm (timer output) width = count clock cycle × TAUDnCDRm

(3) Block Diagram and General Timing Diagram

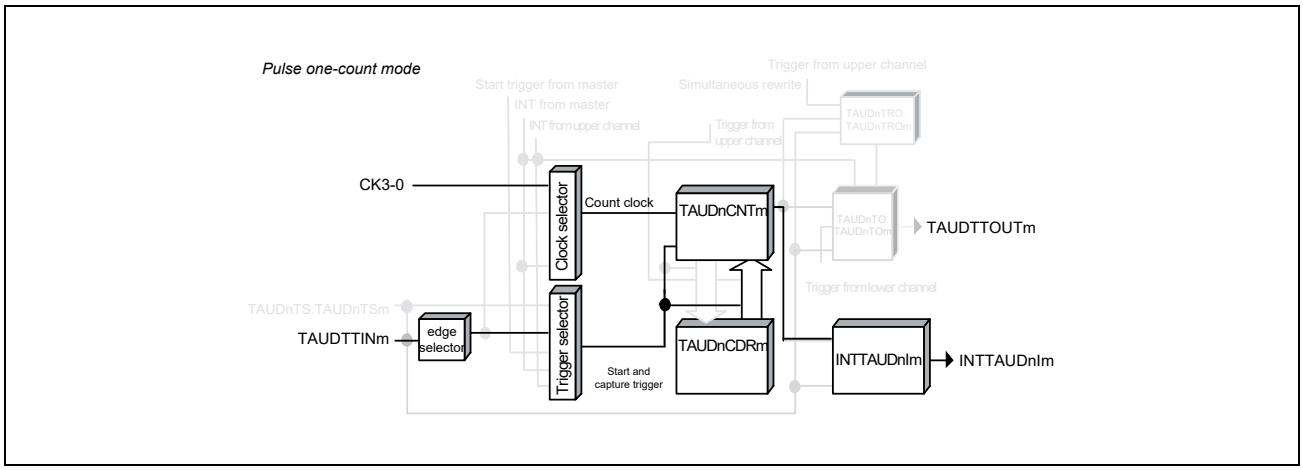


Figure 18.43 Block Diagram of One-Pulse Output Function

The following settings apply to the general timing diagram.

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

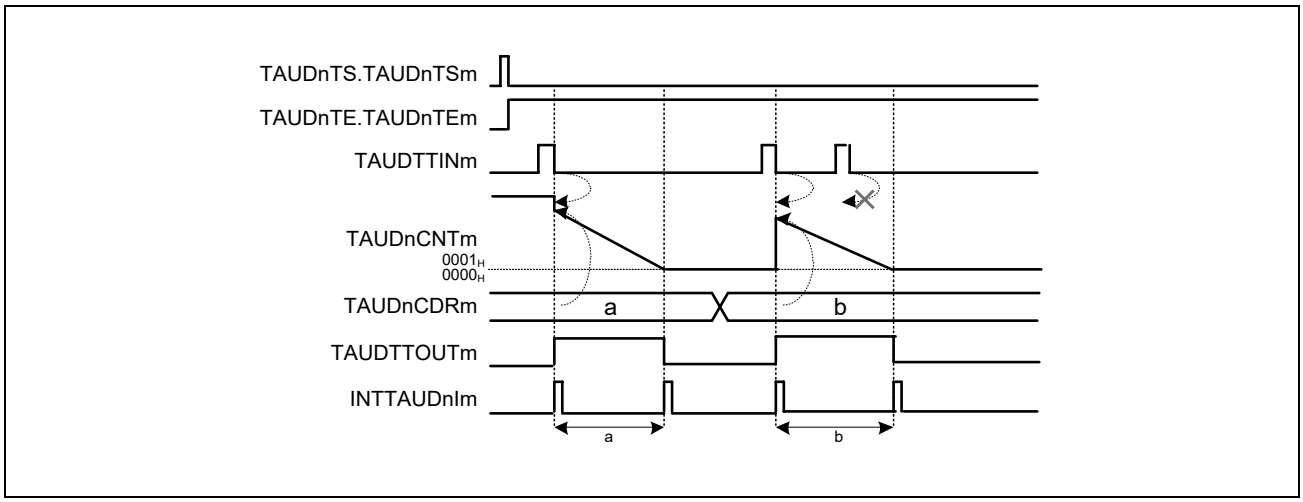


Figure 18.44 General Timing Diagram of One-Pulse Output Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.68 Contents of TAUDnCMORm Register for One-Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as an external start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1010: Pulse one-count mode
0	TAUDnMD0	0: Disables a start trigger during operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.69 Contents of TAUDnCMURm Register for One-Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

Table 18.70 Control Bit Settings in Independent Channel Output Mode 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode controlled by software.
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TAUDnTDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TAUDnTREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

NOTE

The channel output mode can also be set to channel output mode controlled by software by setting TAUDnTOE.TAUDnTOEm = 0. TAUDTTOUTm can then be controlled independently of the interrupts. For details, see **Table 18.43, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the One-Pulse Output Function. Therefore, these registers should be set to 0.

Table 18.71 Simultaneous Rewrite Settings for One-Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for One-Pulse Output Function

Table 18.72 Operating Procedure for One-Pulse Output Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.68, Contents of TAUDnCMORm Register for One-Pulse Output Function**, and **Table 18.69, Contents of TAUDnCMURm Register for One-Pulse Output Function**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 18.70, Control Bit Settings in Independent Channel Output Mode 2**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0. Detection of TAUDTTINm start edge	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When a start edge is detected, TAUDnCNTm loads the TAUDnCDRm value.
During Operation	The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at all times.	INTTAUDnIm is generated when TAUDnCNTm starts and TAUDTTOUTm is set to its active level. TAUDnCNTm counts down. When the counter reaches 0001_H: • INTTAUDnIm is generated. • TAUDTTOUTm is set to its inactive level. TAUDnCNTm stops counting and waits for a trigger. If a trigger occurs while TAUDnCNTm is counting, the trigger is ignored.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

18.4.9.7 TAUDTTINm Input Pulse Interval Measurement Function

(1) Overview

Summary

This function captures the count value and uses this value and the overflow bit TAUDnCSRm.TAUDnOVF to measure the interval of the TAUDTTINm input signal.

Prerequisites

- The operating mode should be set to capture mode. See **Table 18.74, Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Measurement Function**.
- TAUDTTOUTm is not used with this function.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The counter TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm edge is detected, the value of TAUDnCNTm is captured, transferred to TAUDnCDRm, and an interrupt INTTAUDnIm is generated. The counter resets to 0000_H and subsequently continues to operate.

If the counter reaches $FFFF_H$ before an effective TAUDTTINm edge is detected, it overflow. The counter is reset to 0000_H and subsequently continues to operate. The values transferred to TAUDnCDRm and TAUDnCSRm.TAUDnOVF respectively depend on the values of bits TAUDnCMORm.TAUDnCOS[1:0].

Table 18.73 Effects of Overflow

TAUDnCMORm. TAUDnCOS[1:0]	When Overflow Occurs		When an Effective TAUDTTINm Input is Detected	
	TAUDnCDRm	TAUDnCSRm. TAUDnOVF	TAUDnCDRm, TAUDnCNTm	TAUDnCSRm. TAUDnOVF
00	Unchanged	0	TAUDnCNTm loaded into TAUDnCDRm	1
01		1		
10	Set to $FFFF_H$	0	TAUDnCNTm set to 0, TAUDnCDRm unchanged	Unchanged
11		1		

When TAUDnCMORm.TAUDnCOS[0] = 1, the overflow bit (TAUDnCSRm.TAUDnOVF) can be cleared only by setting TAUDnCSCm.TAUDnCLOV = 1.

The combination of the value of TAUDnCDRm and TAUDnCSRm.TAUDnOVF can be used to deduce the interval of the TAUDTTINm signal. However, if an overflow occurs multiple times before an effective TAUDTTINm input is detected, the overflow bit TAUDnCSRm.TAUDnOVF cannot indicate the occurrence of multiple overflows.

The function can be stopped by setting TAUDnTT.TAUDnTTm = 1. This sets TAUDnTE.TAUDnTEm = 0. TAUDnCNTm stops but retains its value. While the function is stopped, effective TAUDTTINm input edge detection and TAUDnCNTm capture are not performed.

The counter is reset to 0000_H and subsequently continues to operate.

Conditions

If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not generated. For details, see **Section 18.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

NOTE

When TAUDnCMORm.TAUDnCOS[1:0] = 10_B or 11_B, the value of TAUDnCNTm is not loaded into TAUDnCDRm when the first effective TAUDTTINm input edge occurs after an overflow. However, an interrupt is generated.

(2) Equations

TAUDTTINm input pulse interval
= count clock cycle × [(TAUDnCSRm.TAUDnOVF × ($FFFF_H$ + 1)) + TAUDnCDRm capture value + 1]

(3) Block Diagram and General Timing Diagram

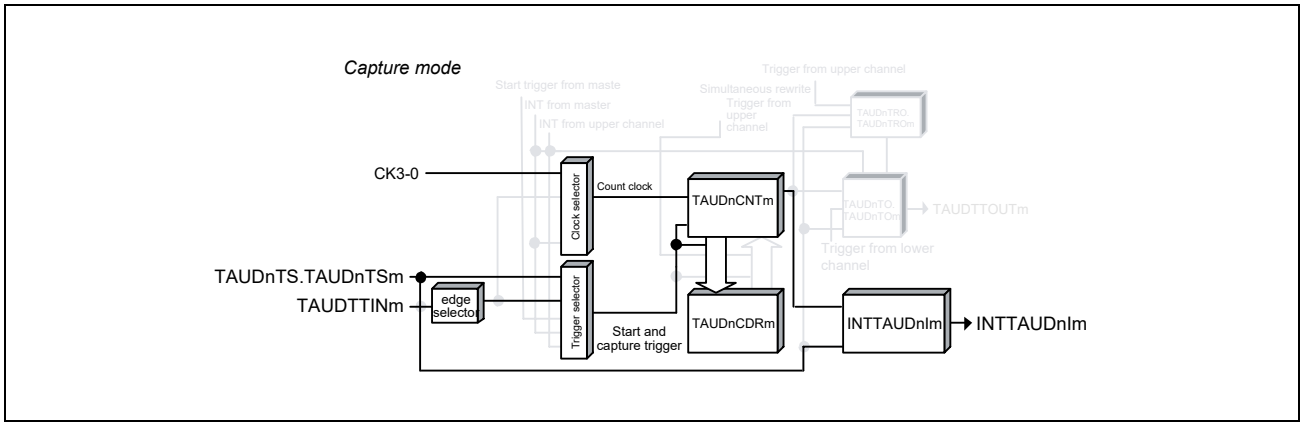


Figure 18.45 Block Diagram of TAUDTTINm Input Pulse Interval Measurement Function

The following settings apply to the general timing diagram.

- INTTAUDnIm not generated at the beginning of operation (TAUDnCMORm.TAUDnMD0 = 0)
- Falling edge detection (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)
- When an effective TAUDTTINm input is detected after an overflow, TAUDnCDRm is changed and TAUDnCSRm.TAUDnOVF is set to 1 (TAUDnCMORm.TAUDnCOS[1:0] = 00_B)

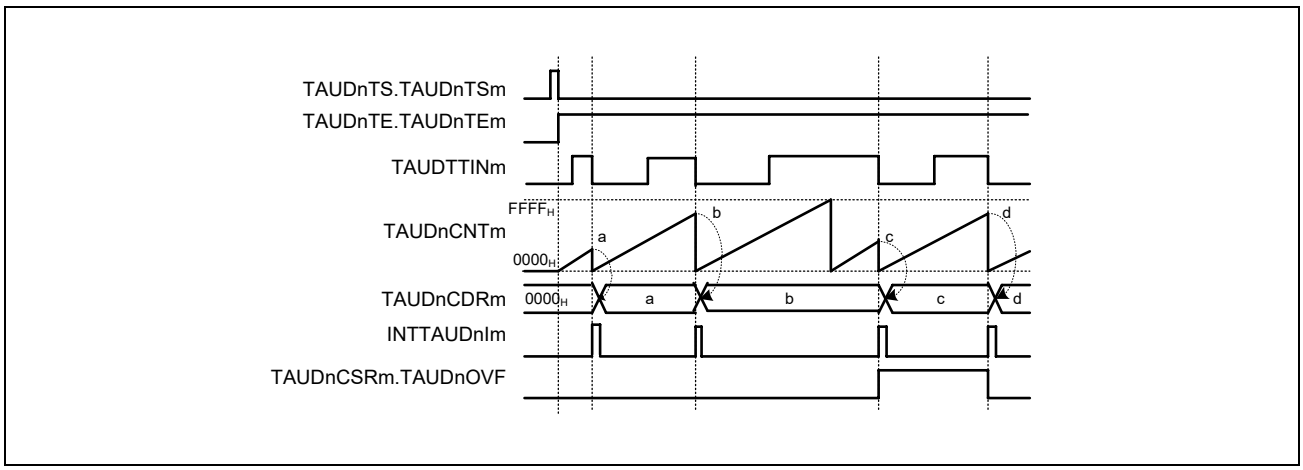


Figure 18.46 General Timing Diagram of TAUDTTINm Input Pulse Interval Measurement Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.74 Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Measurement Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: Effective edge of the TAUDTTINm input signal is the external capture trigger.
7, 6	TAUDnCOS[1:0]	See **Table 18.73, Effects of Overflow.**
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0010: Capture mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.75 Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Measurement Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the TAUDTTINm input pulse interval measurement function. Therefore, these registers should be set to 0.

Table 18.76 Simultaneous Rewrite Settings for TAUDTTINm Input Pulse Interval Measurement Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Pulse Interval Measurement Function

Table 18.77 Operating Procedure for TAUDTTINm Input Pulse Interval Measurement Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.74 Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Measurement Function**, and **Table 18.75 Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Measurement Function**. The TAUDnCDRm register functions as a capture register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCNTm is cleared to 0000_H. INTTAUDnIm is generated when TAUDnCMORm.TAUDnMD0 is set to 1.
During Operation	Detection of TAUDTTINm edge The TAUDnCMURm.TAUDnTIS[1:0] bits can be changed at any time. The TAUDnCDRm and TAUDnCSRm registers can be read at any time. TAUDnCSCm.TAUDnCLOV can be written to 1. (TAUDnCSRm.TAUDnOVF bit is cleared to 0.)	TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm edge is detected: • TAUDnCNTm transfers (captures) its value to TAUDnCDRm, and returns to 0000_H. • INTTAUDnIm is then generated. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and both it and TAUDnCSRm.TAUDnOVF retain their current values.

Restart

(6) Specific Timing Diagrams: Overflow Operation

(a) TAUDnCMORm.TAUDnCOS[1:0] = 00_B

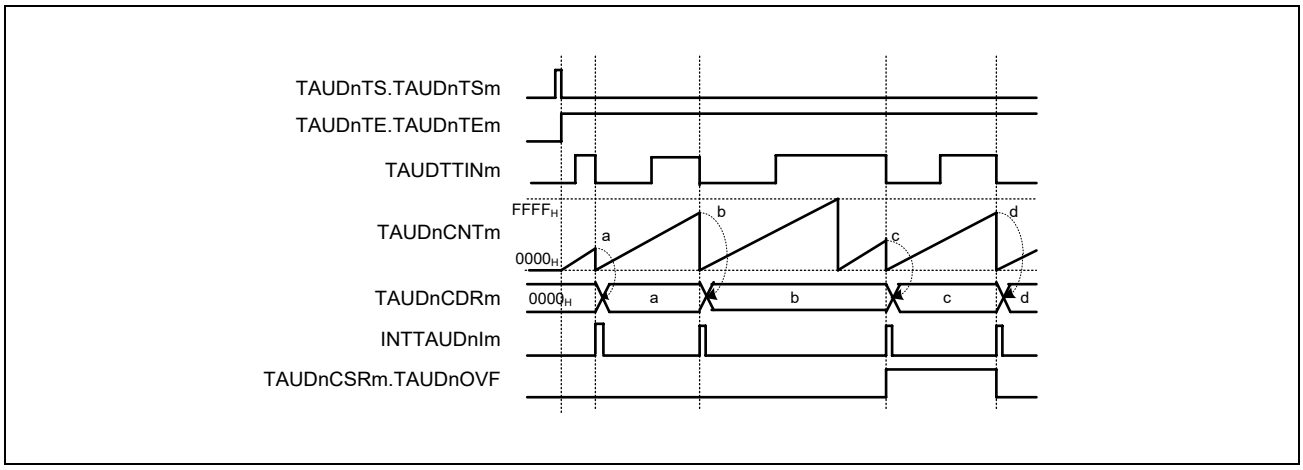


Figure 18.47 TAUDnCMORm.TAUDnCOS[1:0] = 00_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B

- When an overflow occurs, the value of TAUDnCDRm remains unchanged and TAUDnCSRm.TAUDnOVF remains = 0.
- Upon detection of the next effective TAUDTTINm input edge, the value of TAUDnCNTm is loaded into TAUDnCDRm and TAUDnCSRm.TAUDnOVF is set to1.
- Upon detection of the next effective TAUDTTINm input edge with no overflow occurring, TAUDnCSRm.TAUDnOVF is cleared to 0.

(b) TAUDnCMORm.TAUDnCOS[1:0] = 01_B

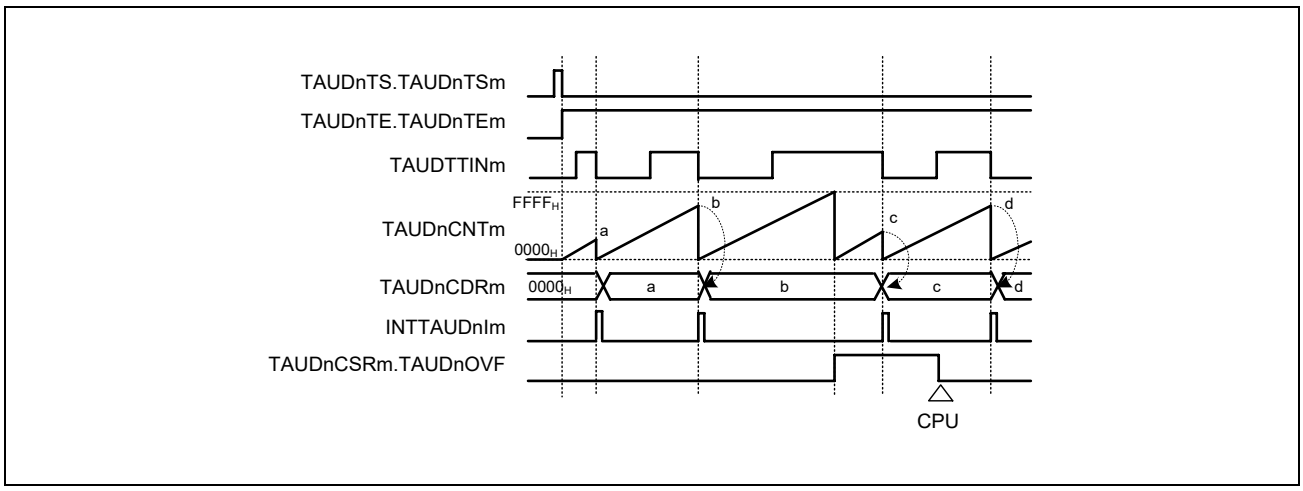


Figure 18.48 TAUDnCMORm.TAUDnCOS[1:0] = 01_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B

- When an overflow occurs, the value of TAUDnCDRm remains unchanged and TAUDnCSRm.TAUDnOVF is set to1.
- Upon detection of the next effective TAUDTTINm input edge, the value of TAUDnCNTm is loaded into TAUDnCDRm.
- TAUDnCSRm.TAUDnOVF is only cleared by a CPU command (by setting TAUDnCSCm.TAUDnCLOV bit to 1).

(c) TAUDnCMORm.TAUDnCOS[1:0] = 10_B

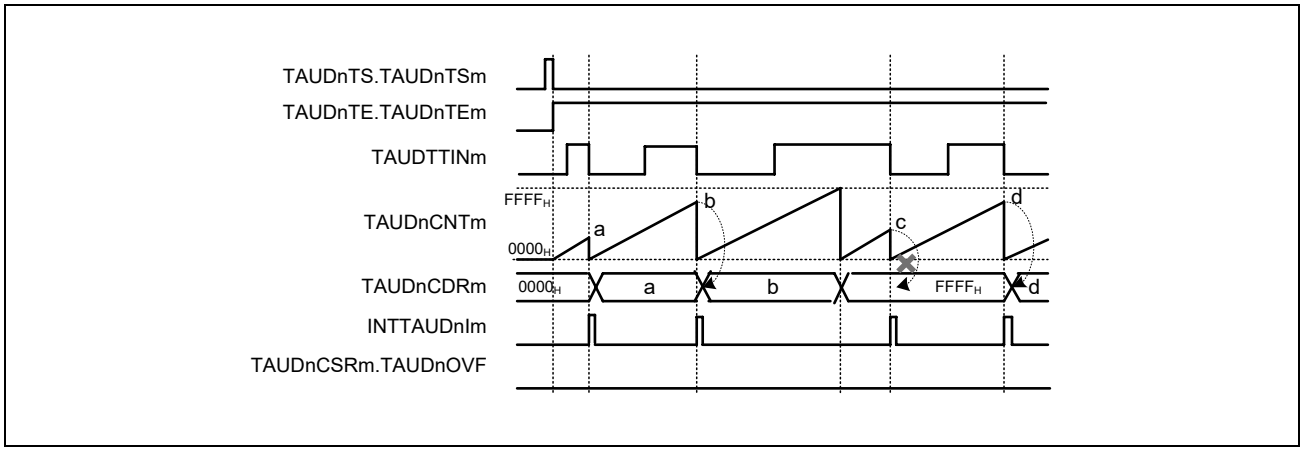


Figure 18.49 TAUDnCMORm.TAUDnCOS[1:0] = 10_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B

- When an overflow occurs, TAUDnCDRm is set to $FFFF_H$ and TAUDnCSRm.TAUDnOVF remains = 0.
- Upon detection of the next effective TAUDTTINm input edge, TAUDnCNTm is reset to 0, but TAUDnCDRm and TAUDnCSRm.TAUDnOVF remain unchanged.
- Thus, the next effective TAUDTTINm input edge after the overflow is ignored.

(d) TAUDnCMORm.TAUDnCOS[1:0] = 11_B

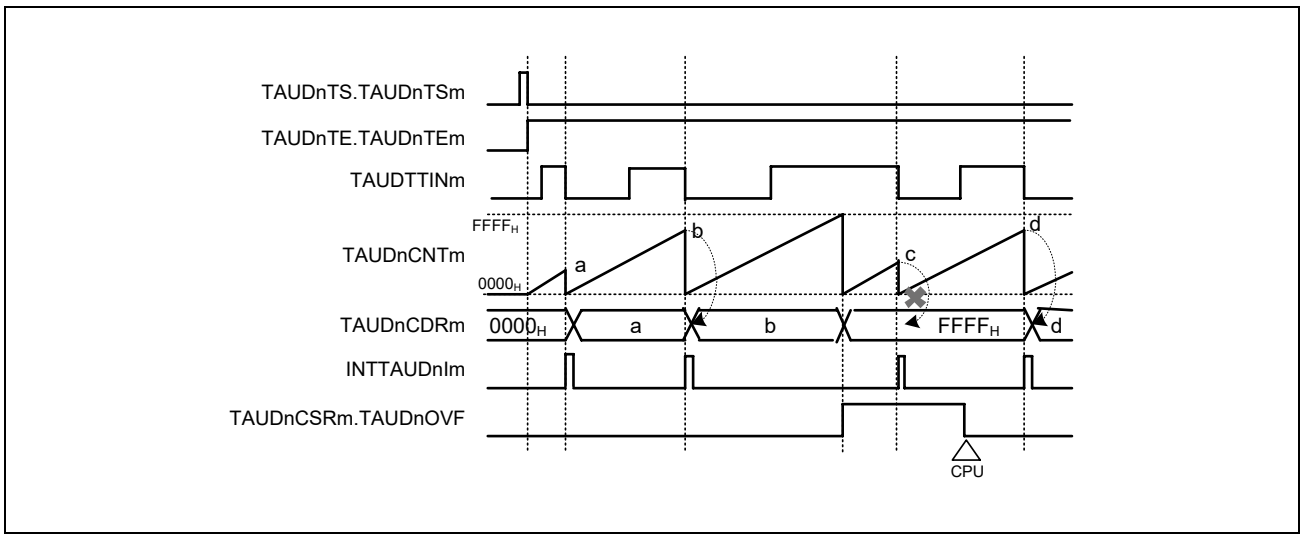


Figure 18.50 TAUDnCMORm.TAUDnCOS[1:0] = 11_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B

- When an overflow occurs, TAUDnCDRm is set to $FFFF_H$ and TAUDnCSRm.TAUDnOVF is set to 1.
- Upon detection of the next effective TAUDTTINm input edge, TAUDnCNTm is reset to 0, but TAUDnCDRm and TAUDnCSRm.TAUDnOVF remain unchanged.
- Thus, the next effective TAUDTTINm input edge after the overflow is ignored.
- TAUDnCSRm.TAUDnOVF is cleared by setting TAUDnCSCm.TAUDnCLOV to 1.

18.4.9.8 TAUDTTINm Input Signal Width Measurement Function

(1) Overview

Summary

This function measures the width of a TAUDTTINm signal, by starting the count at one edge of TAUDTTINm and capturing the count value at the other edge.

Prerequisites

- The operating mode should be set to capture and one-count mode. See **Table 18.79, Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Measurement Function**
- TAUDTTOUTm is not used with this function.
- TAUDnCMORm.TAUDnMD0 should be set to 0.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. When an effective TAUDTTINm start edge is detected, the counter TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm stop edge is detected, the value of TAUDnCNTm is captured, transferred to TAUDnCDRm, and an interrupt INTTAUDnIm is generated. The counter retains its value (TAUDnCDRm +1) and awaits the next effective TAUDTTINm input start edge.

If the counter reaches $FFFF_H$ before an effective TAUDTTINm stop edge is detected, it overflows. The counter is reset to 0000_H and subsequently continues to operate. The values transferred to TAUDnCDRm and TAUDnCSRm.TAUDnOVF respectively depend on the values of bits TAUDnCMORm.TAUDnCOS[1:0].

Table 18.78 Effects or Overflow

	When Overflow Occurs		When an Effective TAUDTTINm Input Stop Edge is Detected	
TAUDnCMORm. TAUDnCOS[1:0]	TAUDnCDRm	TAUDnCSRm. TAUDnOVF	TAUDnCDRm, TAUDnCNTm	TAUDnCSRm. TAUDnOVF
00	Unchanged	0	TAUDnCNTm loaded into TAUDnCDRm	1
01		1		
10	Set to $FFFF_H$	0	TAUDnCNTm stops counting TAUDnCDRm unchanged	Unchanged
11		1		

When TAUDnCMORm.TAUDnCOS[0] = 1, overflow bit TAUDnCSRm.TAUDnOVF can be cleared only by setting TAUDnCSCm.TAUDnCLOV to 1.

The combination of the value of TAUDnCDRm and TAUDnCSRm.TAUDnOVF can be used to deduce the width of the TAUDTTINm signal. However, if an overflow occurs multiple times before an effective TAUDTTINm input is detected, overflow bit TAUDnCSRm.TAUDnOVF cannot indicate the occurrence of multiple overflows.

This function cannot be forcibly restarted.

NOTE

When TAUDnCMORm.COS[1:0] = 10_B or 11_B, the value of TAUDnCNTm is not loaded to TAUDnCDRm when the first effective TAUDTTINm input edge occurs after an overflow. However, an interrupt is generated.

(2) Equations

TAUDTTINm input signal width
= count clock cycle × [(TAUDnCSRm.TAUDnOVF × ($FFFF_H$ + 1)) + TAUDnCDRm capture value + 1]

(3) Block Diagram and General Timing Diagram

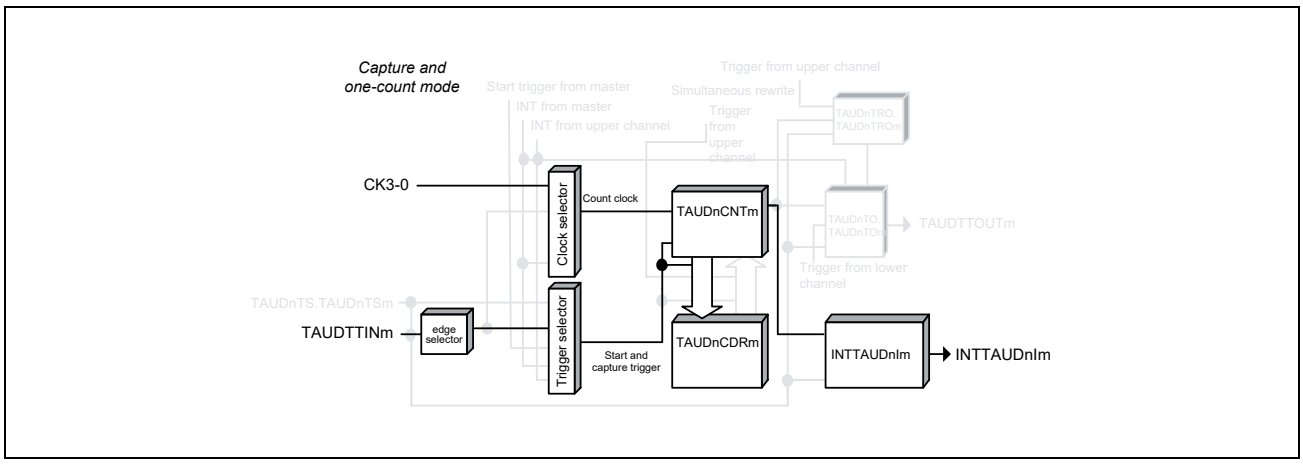


Figure 18.51 Block Diagram of TAUDTTINm Input Signal Width Measurement Function

The following settings apply to the general timing diagram.

- Detection of rising and falling edges = high width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)
- When an effective TAUDTTINm input is detected after an overflow, TAUDnCDRm is changed and TAUDnCSRm.TAUDnOVF is set to 1. (TAUDnCMORm.TAUDnCOS[1:0] = 00_B)

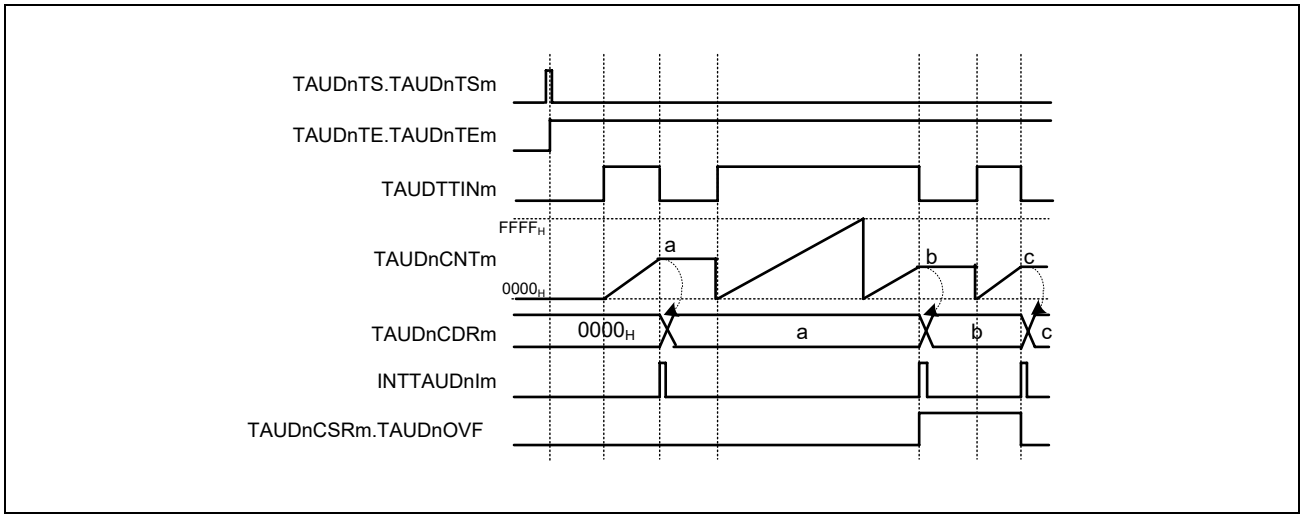


Figure 18.52 General Timing Diagram of TAUDTTINm Input Signal Width Measurement Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.79 Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Measurement Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	010: Effective edge of the TAUDTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUDnCOS[1:0]	See **Table 18.78, Effects or Overflow.**
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0110: Capture and one-count mode
0	TAUDnMD0	0: Disables the start trigger during operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.80 Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Measurement Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	10: Detection of rising and falling edges (low width measurement) 11: Detection of rising and falling edges (high width measurement)

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the TAUDTTINm input signal width measurement function. Therefore, these registers should be set to 0.

Table 18.81 Simultaneous Rewrite Settings for TAUDTTINm Input Signal Width Measurement Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Signal Width Measurement Function

Table 18.82 Operating Procedure for TAUDTTINm Input Signal Width Measurement Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.79, Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Measurement Function**, and **Table 18.80, Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Measurement Function**. The TAUDnCDRm register functions as a capture register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When a TAUDTTINm start edge is detected, TAUDnCNTm starts to count up.
During Operation	Detection of TAUDTTINm edge TAUDnCDRm, TAUDnCNTm, and TAUDnCSRm registers can be read at any time. The TAUDnCSCm.TAUDnCLOV bit can be set to 1.	TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm edge is detected: • TAUDnCNTm transfers (captures) its value to TAUDnCDRm, and retains its value. INTTAUDnIm is then generated. Counting stops at the “value that transferred to TAUDnCDRm + 1” and TUAnCNTm waits for detection of the TAUDTTINm start edge. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and both it and TAUDnCSRm.TAUDnOVF retain their current values.

Restart → Start Operation

(6) Specific Timing Diagrams: Overflow Operation

(a) TAUDnCMORm.TAUDnCOS[1:0] = 00_B

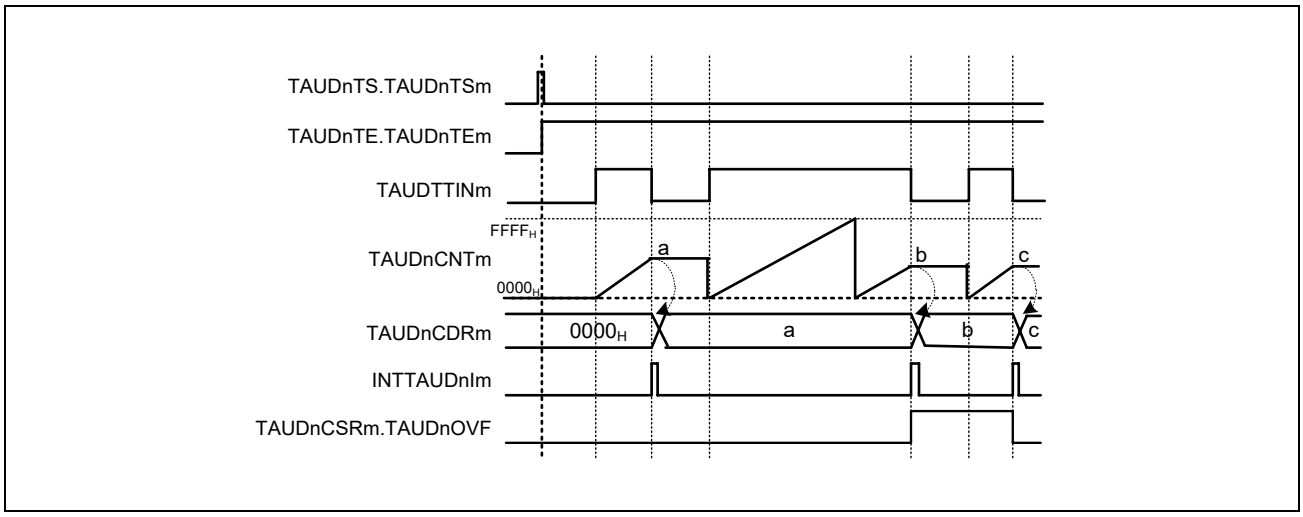


Figure 18.53 TAUDnCMORm.TAUDnCOS[1:0] = 00_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B

- When an overflow occurs, the value of TAUDnCDRm remains unchanged and TAUDnCSRm.TAUDnOVF remains = 0.
- Upon detection of the next effective TAUDTTINm input edge, the value of TAUDnCNTm is loaded into TAUDnCDRm and TAUDnCSRm.TAUDnOVF is set to 1.
- Upon detection of the next effective TAUDTTINm input edge with no overflow occurring, TAUDnCSRm.TAUDnOVF is cleared to 0.

(b) TAUDnCMORm.TAUDnCOS[1:0] = 01_B

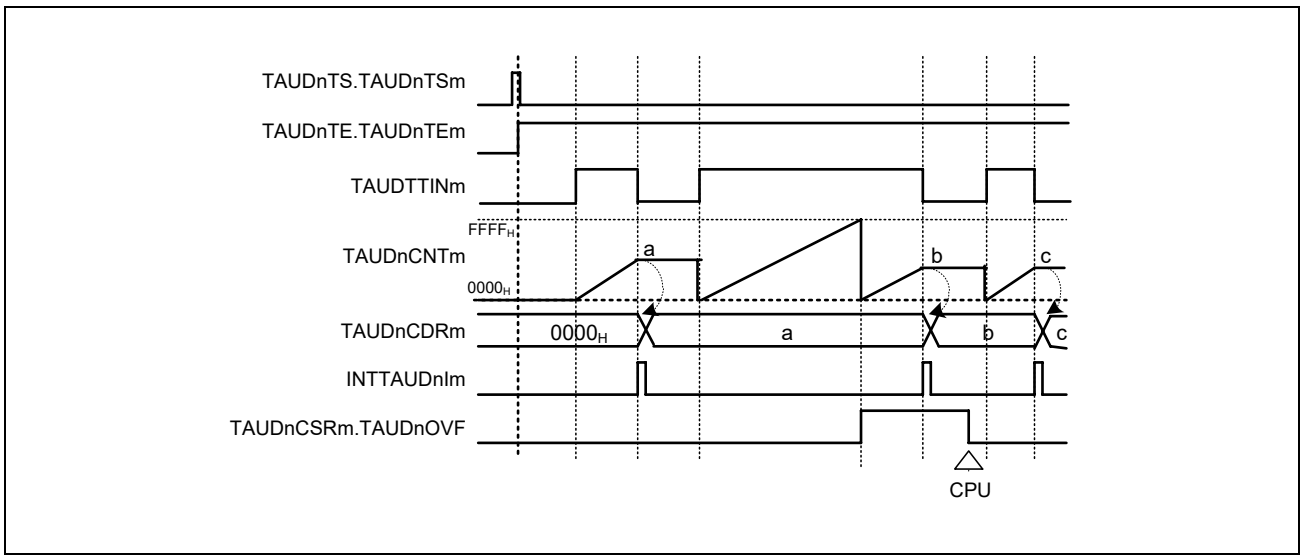


Figure 18.54 TAUDnCMORm.TAUDnCOS[1:0] = 01_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B

- When an overflow occurs, the value of TAUDnCDRm remains unchanged and TAUDnCSRm.TAUDnOVF is set to 1.
- Upon detection of the next effective TAUDTTINm input edge, the value of TAUDnCNTm is loaded into TAUDnCDRm.
- TAUDnCSRm.TAUDnOVF is only cleared by a CPU command (by setting TAUDnCSCm.TAUDnCLOV bit to 1).

(c) TAUDnCMORm.TAUDnCOS[1:0] = 10_B

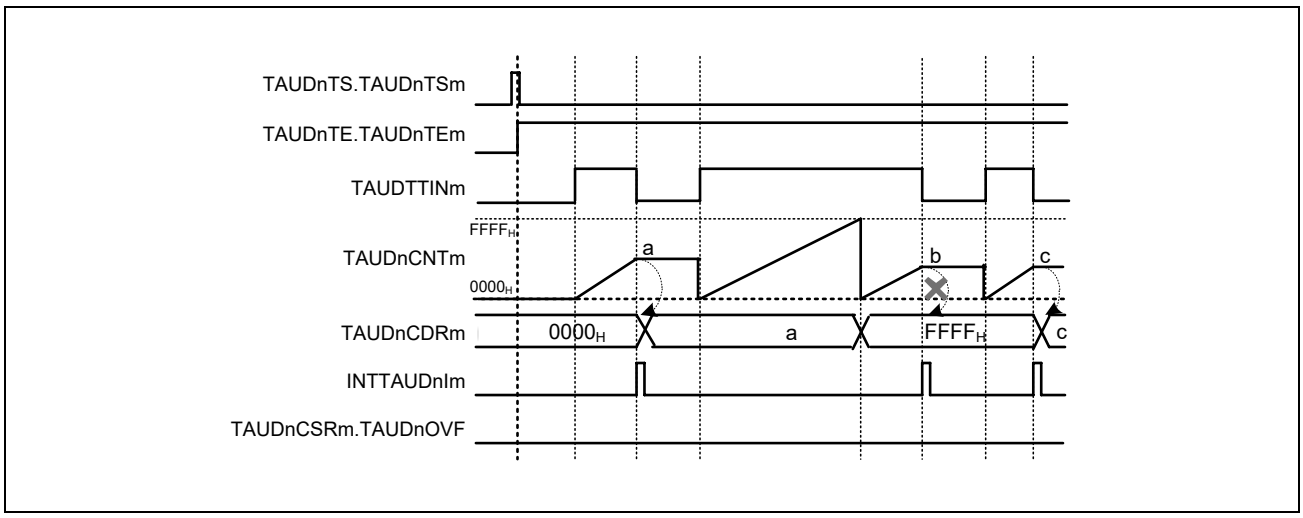


Figure 18.55 TAUDnCMORm.TAUDnCOS[1:0] = 10_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B

- When an overflow occurs, TAUDnCDRm is set to FFFF$_H$ and TAUDnCSRm.TAUDnOVF remains = 0.
- Upon detection of the next effective TAUDTTINm input edge, the counter of TAUDnCNTm is stopped, and TAUDnCDRm and TAUDnCSRm.TAUDnOVF remain unchanged.
- Thus, the next effective TAUDTTINm input edge after the overflow is ignored.

(d) TAUDnCMORm.TAUDnCOS[1:0] = 11_B

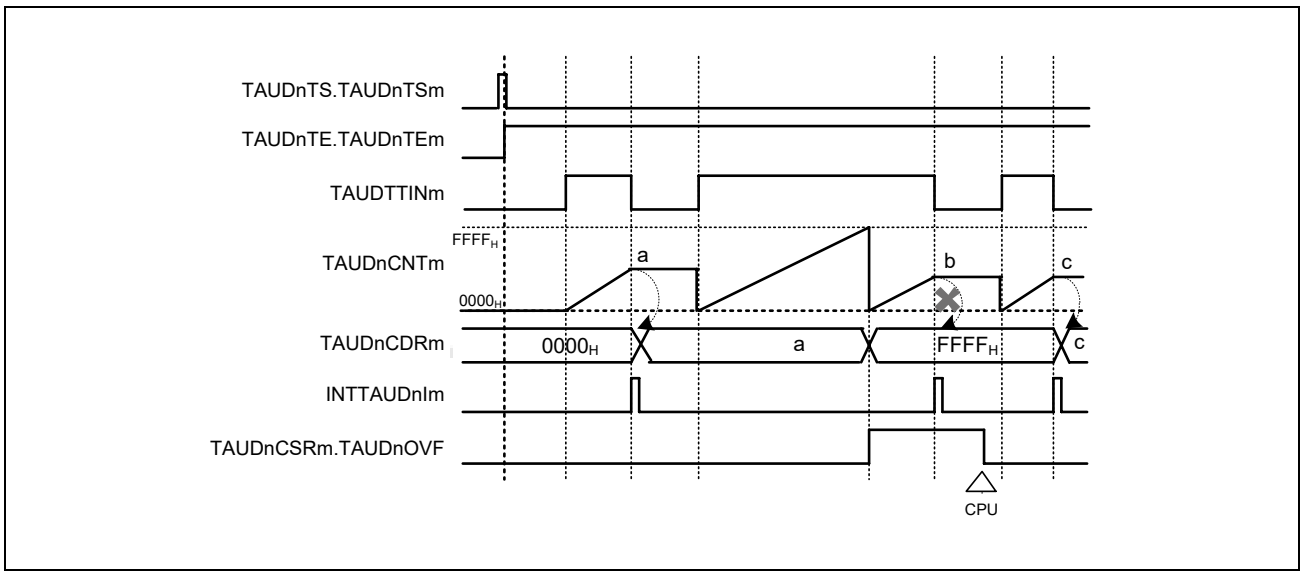


Figure 18.56 TAUDnCMORm.TAUDnCOS[1:0] = 11_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B

- When an overflow occurs, TAUDnCDRm is set to $FFFF_H$ and TAUDnCSRm.TAUDnOVF is set to 1.
- Upon detection of the next effective TAUDTTINm input edge, the counter of TAUDnCNTm is stopped, and TAUDnCDRm and TAUDnCSRm.TAUDnOVF remain unchanged.
- Thus, the next effective TAUDTTINm input edge after the overflow is ignored.
- TAUDnCSRm.TAUDnOVF is cleared by setting TAUDnCSCm.TAUDnCLOV to 1.

18.4.9.9 TAUDTTINm Input Position Detection Function

(1) Overview

Summary

This function measures the input signal duration by capturing the count value at the effective edge of TAUDTTINm.

Prerequisites

- The operating mode should be set to count capture mode (see **Table 18.83, Contents of TAUDnCMORm Register for TAUDTTINm Input Position Detection Function**).
- TAUDTTOUTm is not used with this function.

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The counter starts counting from 0000_H. When an effective TAUDTTINm input edge is detected, the current value of TAUDnCNTm is loaded into TAUDnCDRm and an interrupt (INTTAUDnlm) is generated. The count operation continues.

When the counter reaches $FFFF_H$, the counter restarts to count from 0000_H.

NOTE

The input TAUDTTINm is sampled at the frequency of the operation clock, specified by the TAUDnCMORm.TAUDnCKS[1:0] bits. As a result, the output cycle of TAUDTTOUTm has an error of ± 1 operation clock cycle.

Conditions

If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt does not occur at the beginning of operation or after restart. For details, see **Section 18.4.6, TAUDTTOUTm Output and INTTAUDnlm Generation when Counter Starts or Restarts.**

(2) Equations

Functional duration at a TAUDTTINm input pulse = count clock cycle × (TAUDnCDRm capture value + 1)

(3) Block Diagram and General Timing Diagram

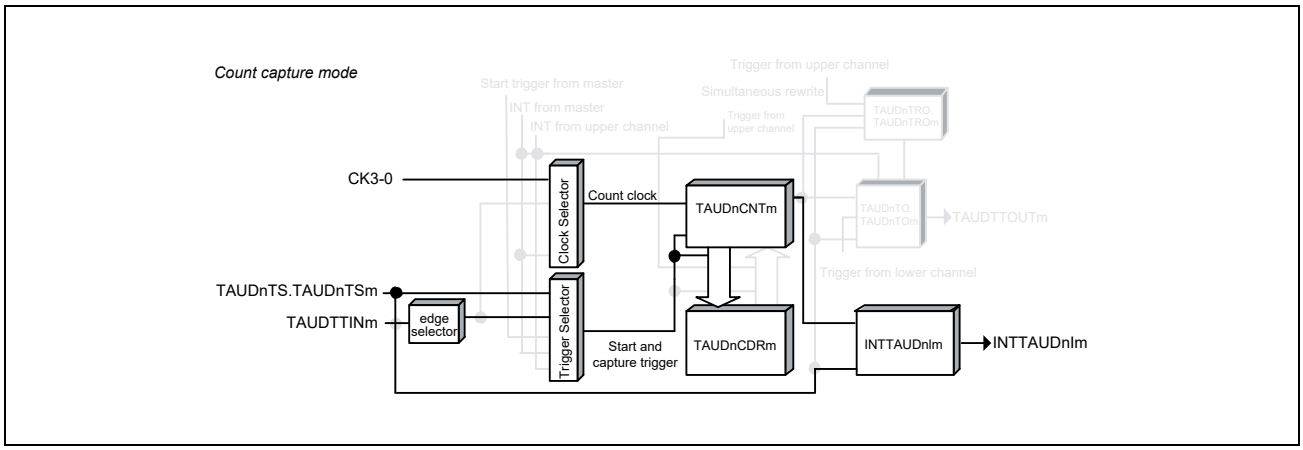


Figure 18.57 Block Diagram of TAUDTTINm Input Position Detection Function

The following settings apply to the general timing diagram.

- INTTAUDnIm not generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

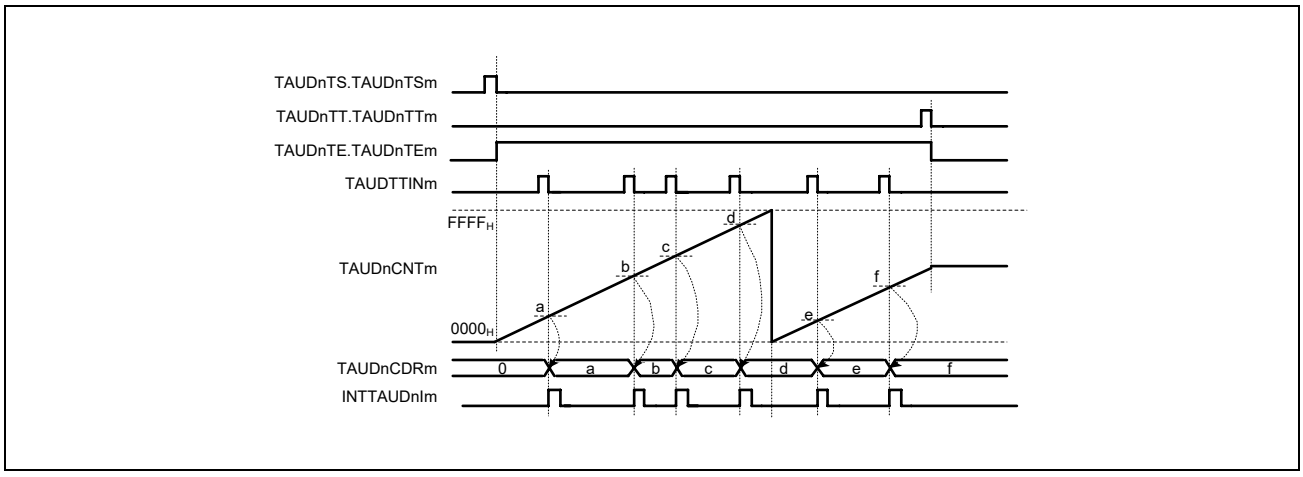


Figure 18.58 General Timing Diagram of TAUDTTINm Input Position Detection Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.83 Contents of TAUDnCMORm Register for TAUDTTINm Input Position Detection Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as an external capture trigger.
7, 6	TAUDnCOS[1:0]	01: This value must be set.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1011: Count capture mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.84 Contents of TAUDnCMURm Register for TAUDTTINm Input Position Detection Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

The channel output mode is not used by this function.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.85 Simultaneous Rewrite Settings for TAUDTTINm Input Position Detection Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Position Detection Function

Table 18.86 Operating Procedure for TAUDTTINm Input Position Detection Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.83, Contents of TAUDnCMORm Register for TAUDTTINm Input Position Detection Function** and **Table 18.84, Contents of TAUDnCMURm Register for TAUDTTINm Input Position Detection Function**. The TAUDnCDRm register functions as a capture register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. If TAUDnCMORm.TAUDnMD0 is 1, INTTAUDnIm occurs.
During Operation	The TAUDnCMURm.TAUDnTIS[1:0] bits can be changed at any time. The TAUDnCDRm and TAUDnCSRm registers can be read at any time.	TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm edge is detected: • TAUDnCNTm transfers (captures) its own value to TAUDnCDRm. • INTTAUDnIm occurs. • The counter is not cleared to 0000_H and TAUDnCNTm continues counting. Afterwards, this procedure is repeated. If TAUDnCNTm reaches $FFFF_H$, the counter restarts from 0000_H.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm retains their current values.

Restart

(6) Specific Timing Diagrams

(a) Operation stop and restart

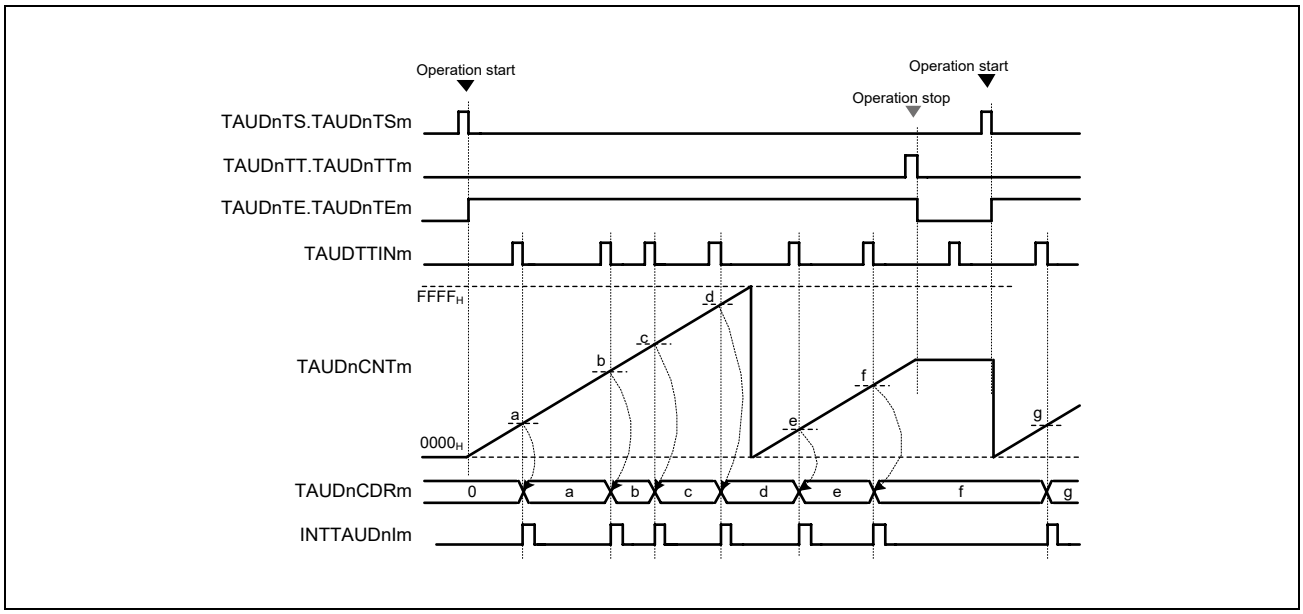


Figure 18.59 Operation Stop and Restart (TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

- The counter can stop operating by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm stops and retains its current value.
- If the counter stops operating, effective TAUDTTINm input edges are ignored.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. TAUDnCNTm restarts to count from 0000_H.

18.4.9.10 TAUDTTINm Input Period Count Detection Function

(1) Overview

Summary

This function measures the cumulative width of a TAUDTTINm input signal.

Prerequisites

- The operating mode should be set to capture and gate count mode (see **Table 18.87, Contents of TAUDnCMORm Register for TAUDTTINm Input Period Count Detection Function**).
- TAUDTTOUTm is not used with this function.

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The counter awaits an effective TAUDTTINm input edge.

When an effective TAUDTTINm input start edge is detected, the counter starts to count from 0000_H.

When an effective TAUDTTINm input stop edge is detected, the current TAUDnCNTm value is loaded into TAUDnCDRm and an interrupt (INTTAUDnIm) is generated. The counter stops and retains its value (TAUDnCDRm + 1) until the next effective TAUDTTINm input start edge is detected.

When the next effective TAUDTTINm input start edge is detected, the counter restarts to count from the value retained when stopped.

If the counter reaches $FFFF_H$, the counter restarts to count from 0000_H.

NOTE

TAUDTTINm input signal is sampled at the frequency of an operation clock set by the TAUDnCMORm.TAUDnCKS[1:0] bits.

As this function is to measure the TAUDTTINm input signal width, setting TAUDnTS.TAUDnTSm to 1 is disabled while TAUDnTE.TAUDnTEm = 1.

Conditions

The effective start and stop edges are specified by the TAUDnCMURm.TAUDnTIS[1:0] bits:

- If TAUDnCMURm.TAUDnTIS[1:0] = 10_B, the TAUDTTINm input low period is measured. The start trigger is a falling edge and the stop trigger is a rising edge.
- If TAUDnCMURm.TAUDnTIS[1:0] = 11_B, the TAUDTTINm input high period is measured. The start trigger is a rising edge and the stop trigger is a falling edge.

(2) Equations

Cumulative TAUDTTINm input width = count clock cycle × (TAUDnCDRm capture value + 1)

(3) Block Diagram and General Timing Diagram

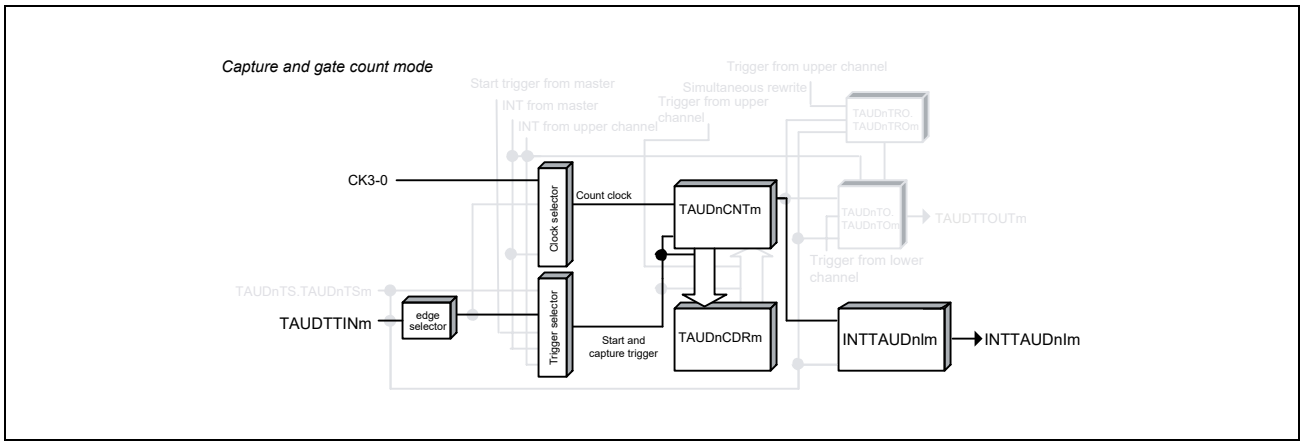


Figure 18.60 Block Diagram of TAUDTTINm Input Period Count Detection Function

The following settings apply to the general timing diagram.

- Detection of rising and falling edges = high width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)

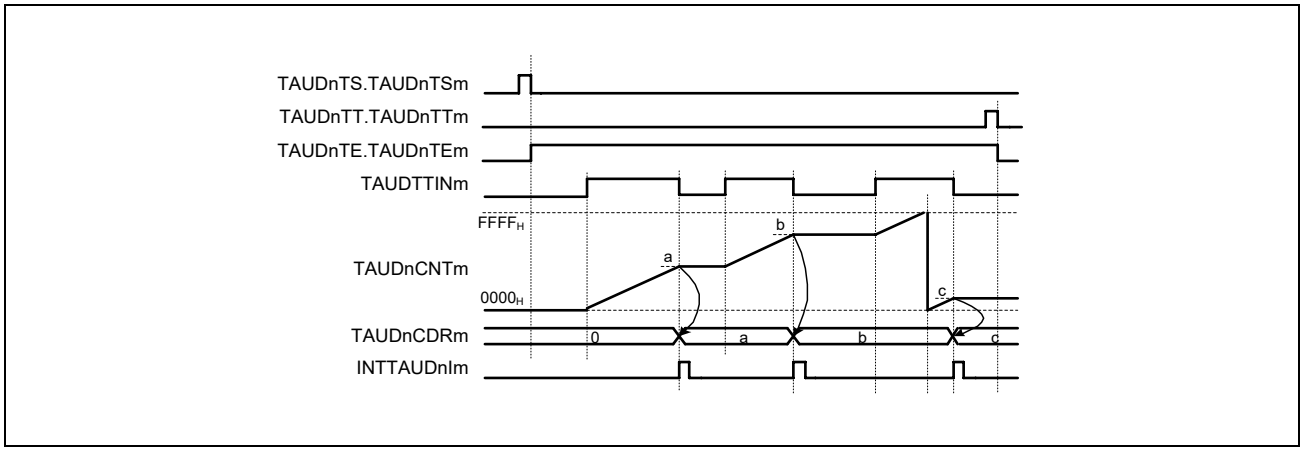


Figure 18.61 General Timing Diagram of TAUDTTINm Input Period Count Detection Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.87 Contents of TAUDnCMORm Register for TAUDTTINm Input Period Count Detection Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	010: Effective edge of the TAUDTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUDnCOS[1:0]	01: This value must be set.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1101: Capture and gate count mode
0	TAUDnMD0	0: Disables the start trigger during operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.88 Contents of TAUDnCMURm Register for TAUDTTINm Input Period Count Detection Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	10: Detection of rising and falling edges (low width measurement) 11: Detection of rising and falling edges (high width measurement)

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.89 Simultaneous Rewrite Settings for TAUDTTINm Input Period Count Detection Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Period Count Detection Function

Table 18.90 Operating Procedure for TAUDTTINm Input Period Count Detection Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.87, Contents of TAUDnCMORm Register for TAUDTTINm Input Period Count Detection Function**, and **Table 18.88, Contents of TAUDnCMURm Register for TAUDTTINm Input Period Count Detection Function**. The TAUDnCDRm register functions as a capture register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0. Detection of TAUDTTINm start edge	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When a start edge is detected, TAUDnCNTm is cleared to 0000_H and starts counting up.
During Operation	Detection of TAUDTTINm edge The TAUDnCDRm, TAUDnCNTm, and TAUDnCSRm registers can be read at any time. The TAUDnCSCm.TAUDnCLOV bit can be set to 1.	When a TAUDTTINm start edge (rising edge for high width measurement, falling edge for low width measurement) is detected, TAUDnCNTm starts counting up from the stop value. When TAUDnCNTm detects a stop edge (falling edge for high width measurement, rising edge for low width measurement), it transfers the value to TAUDnCDRm and INTTAUDnIm is generated. Counting stops at the “value transferred to TAUDnCDRm + 1” and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When TAUDnCNTm reaches $FFFF_H$, the counter restarts count operation from 0000_H. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm retains its current values.

Restart

(6) Specific Timing Diagrams

(a) Operation stop and restart

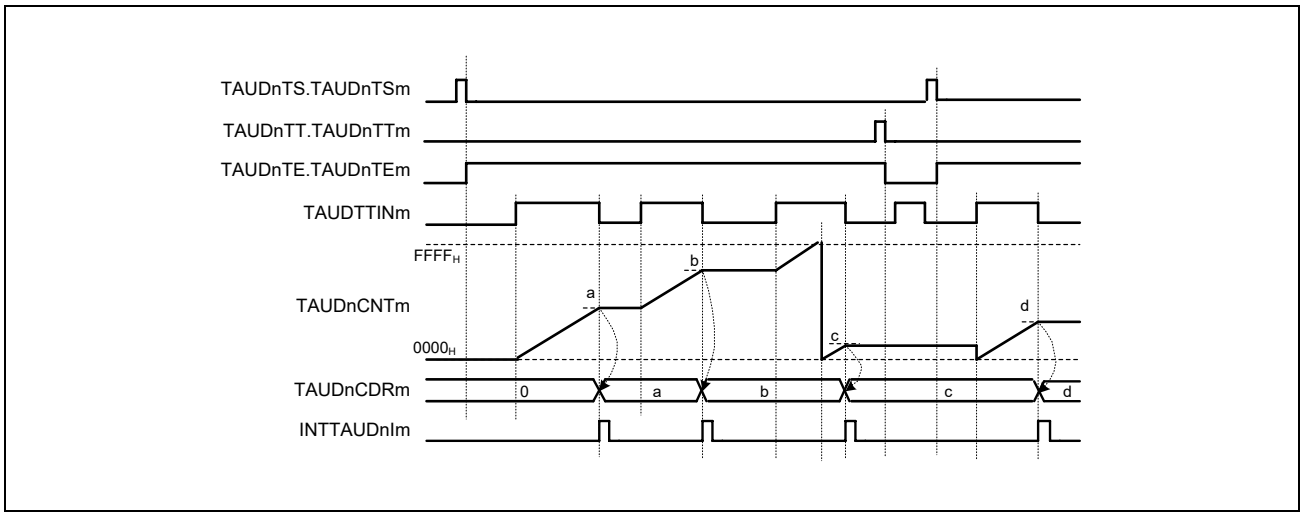


Figure 18.62 Operation Stop and Restart (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)

- The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm stops and retains its current value.
- If the counter is stopped, effective TAUDTTINm input edges are ignored.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. TAUDnCNTm restarts to count from 0000_H.

18.4.9.11 TAUDTTINm Input Pulse Interval Judgment Function

(1) Overview

Summary

This function outputs the result of a comparison between the count value (TAUDnCNTm) and the value in the channel data register (TAUDnCDRm) when a TAUDTTINm input pulse occurs. An interrupt request signal INTTAUDnIm is generated if the result of the comparison is true.

Prerequisites

- The operating mode should be set to judge mode. See **Table 18.91, Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Judgment Function**.
- TAUDTTOUTm is not used with this function.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from this value.

When an effective TAUDTTINm edge is detected or TAUDnTS.TAUDnTSm is set to 1, the function compares the current values of TAUDnCNTm and TAUDnCDRm. An interrupt request signal INTTAUDnIm is generated if the result of the comparison is true. TAUDnCNTm reloads the value of TAUDnCDRm and subsequently continues to operate, regardless of the result of the comparison.

If the counter reaches 0000_H before an effective TAUDTTINm edge is detected, TAUDnCNTm overflows and is set to $FFFF_H$. It then continues to count down.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

The TAUDnCMORm.TAUDnMD0 bit specifies the type of comparison:

- If TAUDnCMORm.TAUDnMD0 = 0, INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm.
- If TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm is generated when TAUDnCNTm > TAUDnCDRm.

(2) Block Diagram and General Timing Diagram

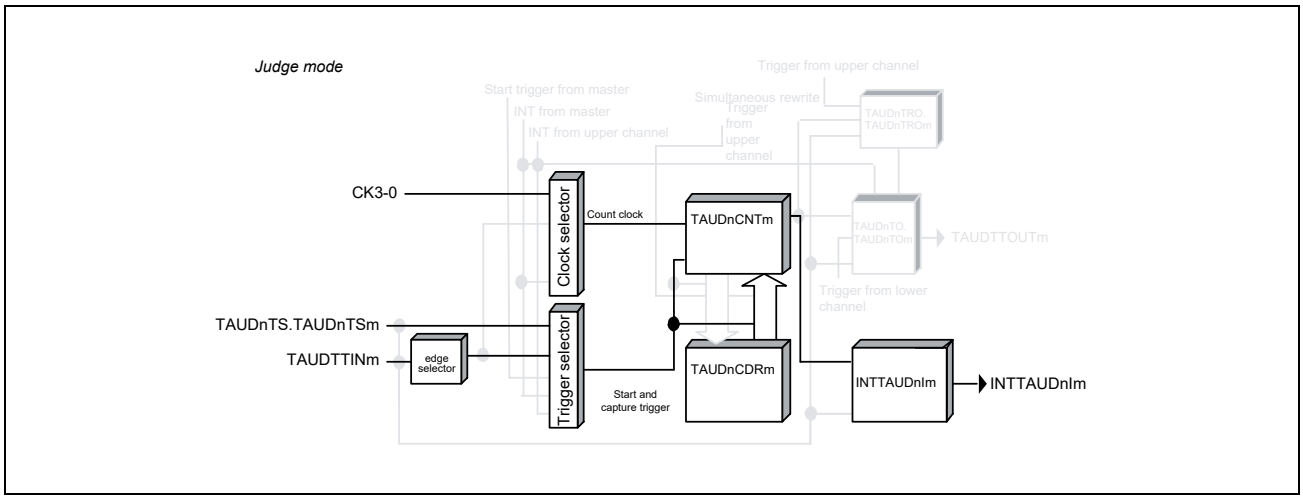


Figure 18.63 Block Diagram of TAUDTTINm Input Pulse Interval Judgment Function

The following settings apply to the general timing diagram.

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

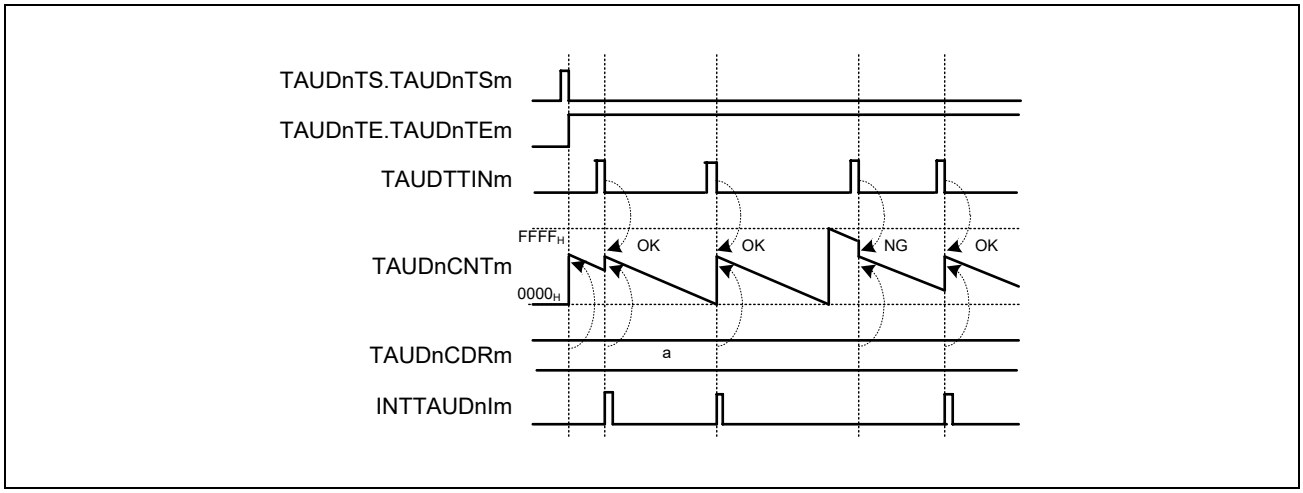


Figure 18.64 General Timing Diagram of TAUDTTINm Input Pulse Interval Judgment Function

(3) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.91 Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Judgment Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: Effective edge of the TAUDTTINm input signal is used as an external start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0001: Judge mode
0	TAUDnMD0	0: INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm 1: INTTAUDnIm is generated when TAUDnCNTm > TAUDnCDRm

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.92 Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Judgment Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.93 Simultaneous Rewrite Settings for TAUDTTINm Input Pulse Interval Judgment Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(4) Operating Procedure for TAUDTTINm Input Pulse Interval Judgment Function

Table 18.94 Operating Procedure for TAUDTTINm Input Pulse Interval Judgment Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.91 Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Judgment Function**, and **Table 18.92 Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Judgment Function**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCDRm value is loaded into TAUDnCNTm.
During Operation	Detection of TAUDTTINm edge The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at any time.	When TAUDnCMORm.TAUDnMD0 = 0: If TAUDnCNTm ≤ TAUDnCDRm when a TAUDTTINm input edge is detected, INTTAUDnIm is generated. When TAUDnCMORm.TAUDnMD0 = 1: If TAUDnCNTm > TAUDnCDRm when a TAUDTTINm input edge is detected, INTTAUDnIm is generated. If a TAUDTTINm input edge is detected, then TAUDnCNTm starts to count down from the value of TAUDnCDRm. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its current value.

Restart

18.4.9.12 TAUDTTINm Input Signal Width Judgment Function

(1) Overview

Summary

This function compares the count value (TAUDnCNTm) for the high or low level width of a TAUDTTINm input signal and the TAUDnCDRm value, and outputs the judgment result from the interrupt request signal INTTAUDnIm.

Prerequisites

- The operating mode should be set to judge and one-count mode (see **Table 18.95, Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Judgment Function**).
- TAUDTTOUTm is not used with this function.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. When an effective TAUDTTINm input start edge is detected, the current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from this value.

When an effective TAUDTTINm stop edge is detected, the function compares the current values of TAUDnCNTm and TAUDnCDRm. An interrupt request signal INTTAUDnIm is generated if the result of the comparison is true. The counter TAUDnCNTm retains its value until the next effective TAUDTTINm start edge is detected, regardless of the result of the comparison.

If the counter reaches 0000_H before an effective TAUDTTINm stop edge is detected, TAUDnCNTm overflows and is set to $FFFF_H$. The counter then continues to count down.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

- The TAUDnCMORm.TAUDnMD0 bit specifies the type of comparison:
  - If TAUDnCMORm.TAUDnMD0 = 0, INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm.
  - If TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm is generated when TAUDnCNTm > TAUDnCDRm.
- The TAUDnCMURm.TAUDnTIS[1:0] bits specify a type of width measurement:
  - For high width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 11_B), TAUDTTINm rising edge is used as a start edge and TAUDTTINm falling edge as a stop edge.
  - For low width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 10_B), TAUDTTINm falling edge is used as a start edge and TAUDTTINm rising edge as a stop edge.
- This function cannot make a forced restart.

(2) Block Diagram and General Timing Diagram

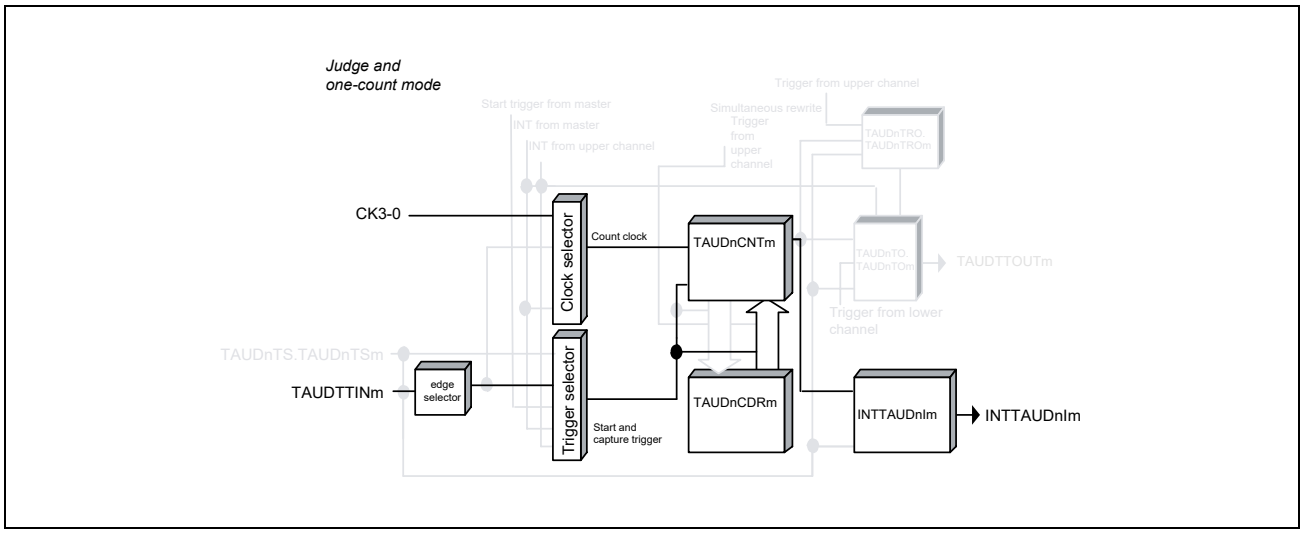


Figure 18.65 Block Diagram of TAUDTTINm Input Signal Width Judgment Function

The following settings apply to the general timing diagram.

- INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm (TAUDnCMORm.TAUDnMD0 = 0)
- Effective TAUDTTINm start edge = rising edge, effective TAUDTTINm stop edge = falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)

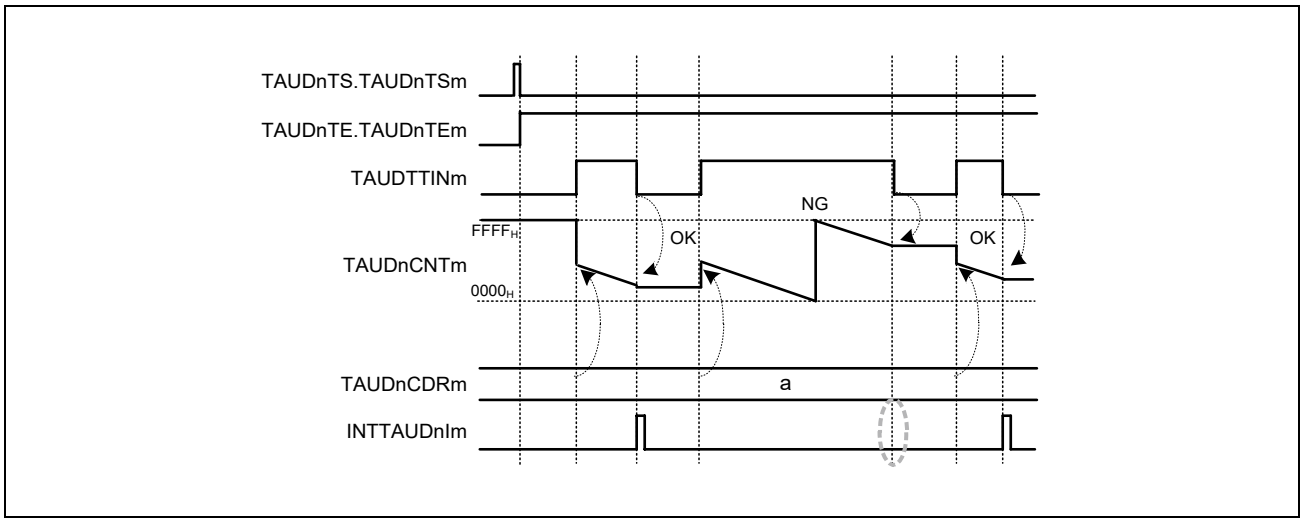


Figure 18.66 General Timing Diagram of TAUDTTINm Input Signal Width Judgment Function

(3) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.95 Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Judgment Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	010: Effective edge of the TAUDTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0111: Judge and one-count mode
0	TAUDnMD0	0: INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm 1: INTTAUDnIm is generated when TAUDnCNTm > TAUDnCDRm

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.96 Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Judgment Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	10: Detection of rising and falling edges (low width measurement) 11: Detection of rising and falling edges (high width measurement)

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.97 Simultaneous Rewrite Settings for TAUDTTINm Input Signal Width Judgment Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(4) Operating Procedure for TAUDTTINm Input Signal Width Judgment Function

Table 18.98 Operating Procedure for TAUDTTINm Input Signal Width Judgment Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 18.95, Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Judgment Function**, and **Table 18.96, Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Judgment Function.** Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge.
During Operation	Detection of TAUDTTINm edge The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at any time.	Upon detection of a TAUDTTINm start edge, TAUDnCNTm starts count down from the value of TAUDnCDRm. When TAUDnCMORm.TAUDnMD0 = 0: If TAUDnCNTm ≤ TAUDnCDRm when a TAUDTTINm input stop edge is detected, INTTAUDnIm is generated. When TAUDnCMORm.TAUDnMD0 = 1: If TAUDnCNTm > TAUDnCDRm when a TAUDTTINm input stop edge is detected, INTTAUDnIm is generated. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its current value.

Restart

18.4.10 Independent Channel Real-Time Functions

- This section describes functions that output the value of the TAUDnTRO.TROm bit in real time:
- **Section 18.4.10.1, Real-Time Output Function Type 1**
- **Section 18.4.10.2, Real-Time Output Function Type 2**

18.4.10.1 Real-Time Output Function Type 1

(1) Overview

Summary

This function outputs a value of the TAUDnTRO.TAUDnTROm bit from TAUDTTOUTm when a specified channel generates an interrupt (INTTAUDnIm). In this function, the interrupt is generated at certain specified intervals.

The upper channel is a channel which generates a real-time output trigger (TAUDnTRC.TAUDnTRCm = 1), and the lower channel is a channel which makes a real-time output in response to the upper channel trigger (TAUDnTRC.TAUDnTRCm = 0).

Prerequisites

- Channels should use the TAUDTTOUTm control of other channels.
- The operating mode for the upper channel should be set to interval timer mode (see **Table 18.99, Contents of TAUDnCMORm Register for Real-Time Output Function Type 1**.
- Any operating mode can be set for lower channels.
- The channel output mode for all the channels should be set to independent channel output mode 1 with real-time output (see **18.4.4, Channel Output Modes**).
- Real-time output should be enabled for the upper channel (TAUDnTRE.TAUDnTREm = 1).

Functional description

The counter of the upper channel is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm, enabling count operation. The current value of the data register of the upper channel (TAUDnCDRm) is loaded into the counter (TAUDnCNTm) and the counter starts to count down from this value.

When the counter of the upper channel reaches 0000_H, INTTAUDnIm is generated and TAUDTTOUTm outputs the current value of the real-time output bit (TAUDnTRO.TAUDnTROm) of every channel (only channels with TAUDnTRE.TAUDnTREm = 1). TAUDnCNTm then reloads the TAUDnCDRm value to continue operation subsequently.

The TAUDTTOUTm signal changes only when an interrupt is generated, and when its value is different to current value of TAUDnTRO.TAUDnTROm at the moment that the interrupt is generated.

Conditions

- The channel which is monitored for INTTAUDnIm occurrence is specified by setting TAUDnTRC.TAUDnTRCm to 1 for the corresponding channel. The TAUDnTRC.TAUDnTRCm bit should be 0 for all other channels.
- If real-time output of a lower channel is disabled (TAUDnTRE.TAUDnTREm = 0) or if the channel itself is used as a rewrite trigger (TAUDnTRC.TAUDnTRCm = 1), the value of that channel's TAUDnTRO.TAUDnTROm bit is output when INTTAUDnIm is generated in that channel.
- If real-time output of a lower channel is enabled (TAUDnTRE.TAUDnTREm = 1) and TAUDnTRC.TAUDnTRCm = 0, the value of that channel's TAUDnTRO.TAUDnTROm bit is output when INTTAUDnIm is generated in the upper channel.
- If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not output. For details, see **Section 18.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

(2) Equations

INTTAUDnIm generation cycle = count clock cycle × (TAUDnCDRm value + 1)

(3) Block Diagram and General Timing Diagram

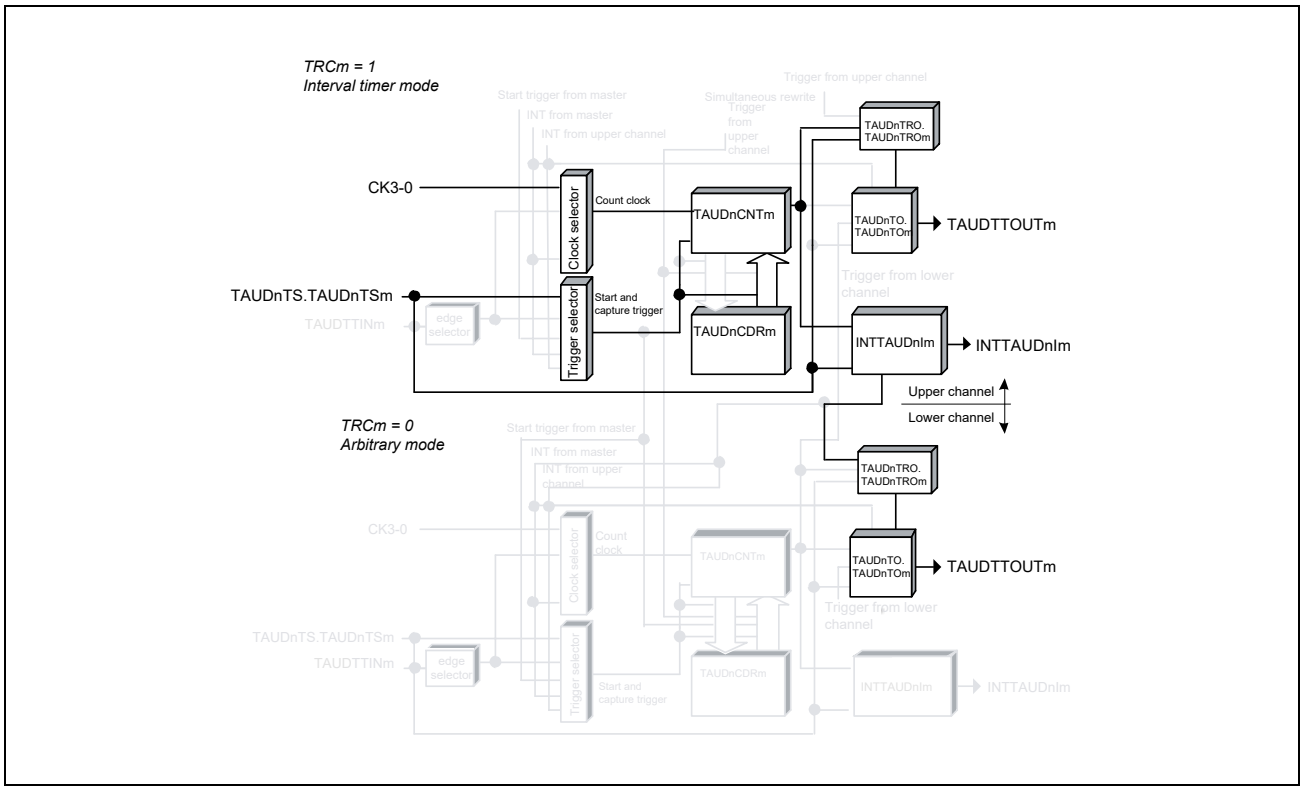


Figure 18.67 Block Diagram of Real-Time Output Function Type 1

The following settings apply to the general timing diagram.

- INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

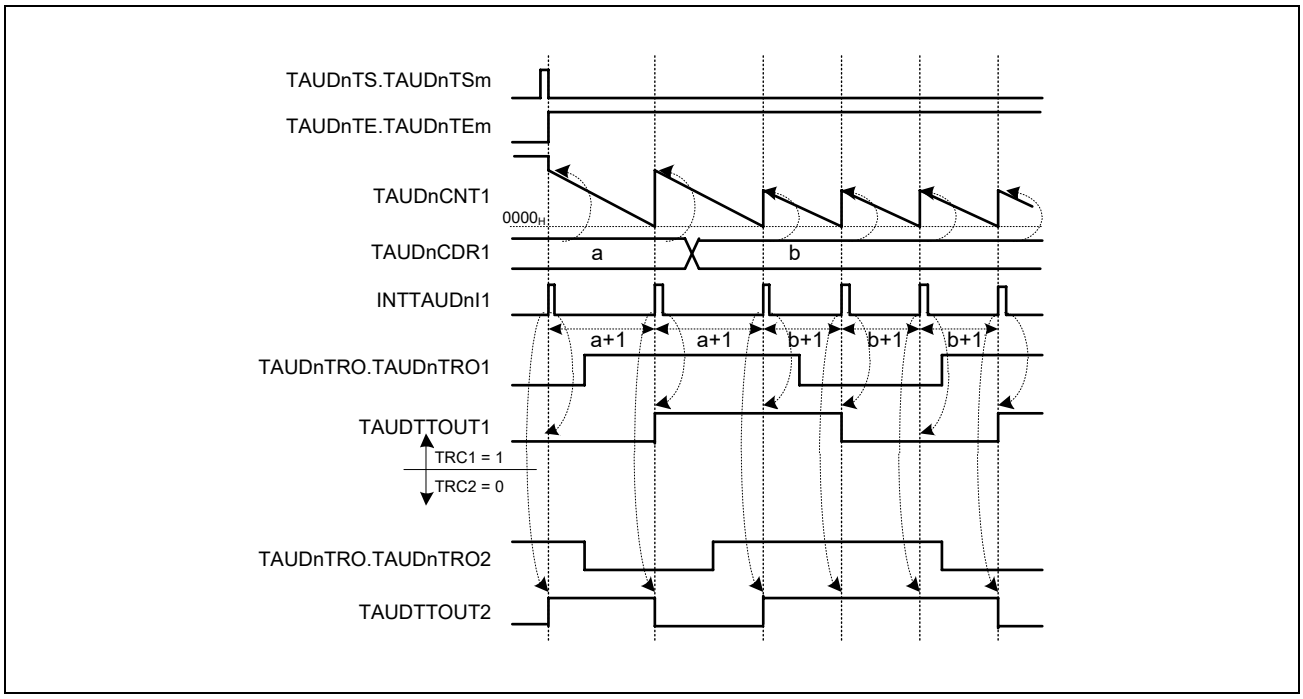


Figure 18.68 General Timing Diagram of Real-Time Output Function Type 1

(4) Register Settings for Upper Channels

(a) TAUDnCMORm for upper channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.99 Contents of TAUDnCMORm Register for Real-Time Output Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for upper channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.100 Contents of TAUDnCMURm Register for Real-Time Output Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for upper channels

Table 18.101 Control Bit Settings in Independent Channel Output Mode 1 with Real-Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	1: Channel m generates a unique real-time output trigger.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for upper channels

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the real-time output function type 1. Therefore, these registers should be set to 0.

Table 18.102 Simultaneous Rewrite Settings for Real-Time Output Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Register Settings for Lower Channels

(a) TAUDnCMORm for lower channels

The TAUDnCMORm register for lower channels is available for any setting.

(b) TAUDnCMURm for lower channels

The TAUDnCMURm register for lower channels is available for any setting.

(c) Channel output mode for lower channels

Table 18.103 Control Bit Settings for the Lower Channels in Independent Channel Output Mode 1 with Real-Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for lower channels

Simultaneous rewrite registers for lower channels can be set arbitrarily.

(6) Operating Procedure for Real-Time Output Function Type 1

Table 18.104 Operating Procedure for Real-Time Output Function Type 1

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers for upper channels as described in **Table 18.99, Contents of TAUDnCMORm Register for Real-Time Output Function Type 1**, and **Table 18.100, Contents of TAUDnCMURm Register for Real-Time Output Function Type 1**. Set TAUDnCMORm and TAUDnCMURm registers for lower channels as described in **Section (5), Register Settings for Lower Channels**. Set the value of TAUDnCDRm register (only channels with TAUDnTRC.TAUDnTRCm = 1) Set channel output mode by setting the control bits as described in **Table 18.101, Control Bit Settings in Independent Channel Output Mode 1 with Real-Time Output**. Set channel output mode by setting the control bits as described in **Table 18.103, Control Bit Settings for the Lower Channels in Independent Channel Output Mode 1 with Real-Time Output**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm = 1 on the channel with TAUDnTRC.TAUDnTRCm set to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	[Channel with TAUDnTRC.TAUDnTRCm set to 1] TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCDRm value is loaded into TAUDnCNTm.TAUDnCMORm. If TAUDnCMORm.TAUDnMD0 is 1, INTTAUDnIm occurs.
During Operation	TAUDnCDRm and TAUDnTRO.TAUDnTROm can be changed at any time. The TAUDnCNTm register can be read at any time.	TAUDnCNTm counts down. When the counter reaches 0000_H: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated. • TAUDTTOUTm outputs the current value of the real-time output bit TAUDnTRO.TAUDnTROm. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. Both TAUDnCNTm and TAUDTTOUTm retain their current values.

Restart

(7) Specific Timing Diagrams

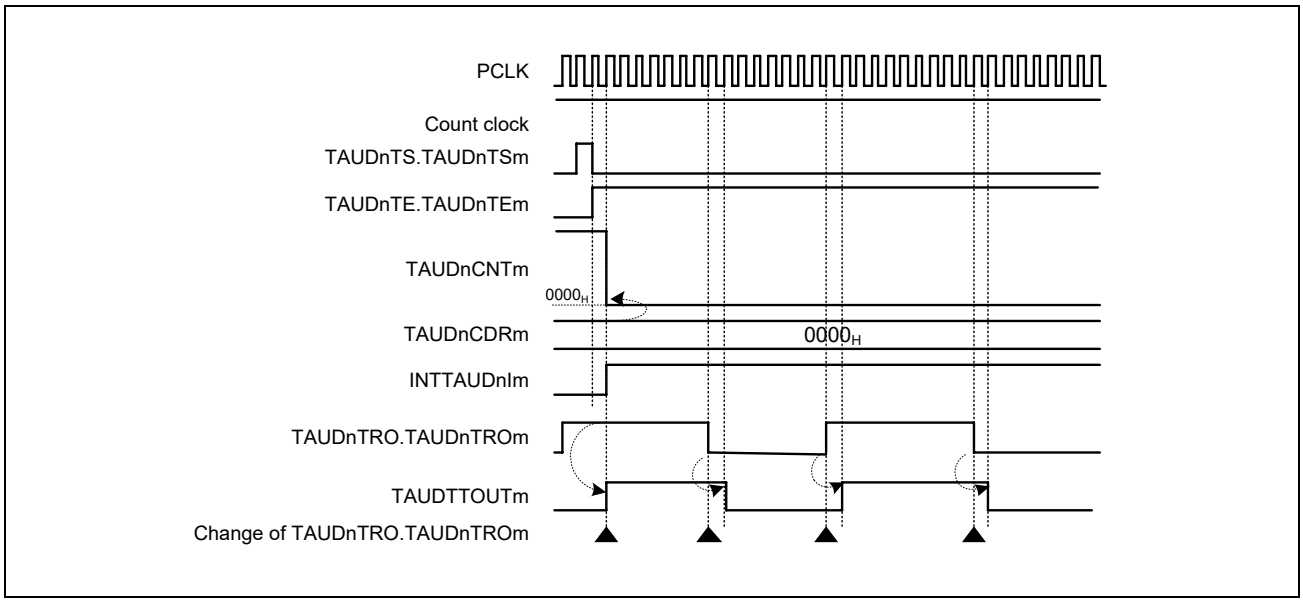


Figure 18.69 TAUDnCDRm = 0000_H, TAUDnCMORm.TAUDnMD0 = 1

- The value of TAUDTTOUTm changes according to the setting of TAUDnTRO.TAUDnTROm with a delay of one PCLK cycle.

18.4.10.2 Real-Time Output Function Type 2

(1) Overview

Summary

This function outputs the value of TAUDnTRO.TAUDnTROm bit from TAUDTTOUTm when a specified channel generates an interrupt (INTTAUDnIm). In this function, the interrupt is generated when an effective TAUDTTINm input edge is detected or the function starts.

The upper channel is a channel which generates a real-time output trigger (TAUDnTRC.TAUDnTRCm = 1), and the lower channel is a channel which makes a real-time output in response to the upper channel trigger (TAUDnTRC.TAUDnTRCm = 0).

Prerequisites

- Channels should use the TAUDTTOUTm control of the other channels.
- The operating mode for the upper channel should be set to interval timer mode (**see Table 18.105, Contents of TAUDnCMORm Register for Real-Time Output Function Type 2**).
- Any operating mode can be set for lower channels.
- The channel output mode for all the channels should be set to independent channel output mode 1 with real-time output (see **Section 18.4.4, Channel Output Modes**).
- Real-time output should be enabled for the upper channel (TAUDnTRE.TREm = 1).

Functional description

The counter for upper channels is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1.This sets TAUDnTE.TAUDnTEm to 1, enabling count operation. The counter starts to count up.

When an effective TAUDTTINm input edge is generated on one of upper channels, an interrupt occurs and TAUDTTOUTm outputs the current value of the real-time output bit (TAUDnTRO.TAUDnTROm) of every channel (only channels with TAUDnTRE.TAUDnTREm = 1).

The TAUDTTOUTm signal changes only when an interrupt is generated, and when TAUDTTOUTm value is different to the current value of TAUDnTRO.TAUDnTROm during the occurrence of the interrupt.

Conditions

- The channel which is monitored for INTTAUDnIm occurrence is specified by setting TAUDnTRC.TRCm to 1 for the corresponding channel. The TAUDnTRC.TRCm bit should be 0 for all other channels.
- If real-time output of a lower channel is disabled (TAUDnTRE.TAUDnTREm = 0) or if the channel itself is used as a rewrite trigger (TAUDnTRC.TAUDnTRCm = 1), the value of that channel's TAUDnTRO.TAUDnTROm bit is output when INTTAUDnIm is generated in that channel.
- If real-time output of a lower channel is enabled (TAUDnTRE.TAUDnTREm = 1) and TAUDnTRC.TAUDnTRCm = 0, the value of that channel's TAUDnTRO.TAUDnTROm bit is output when INTTAUDnIm is generated in the upper channel.
- If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not output. For details, see **Section 18.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

(2) Block Diagram and General Timing Diagram

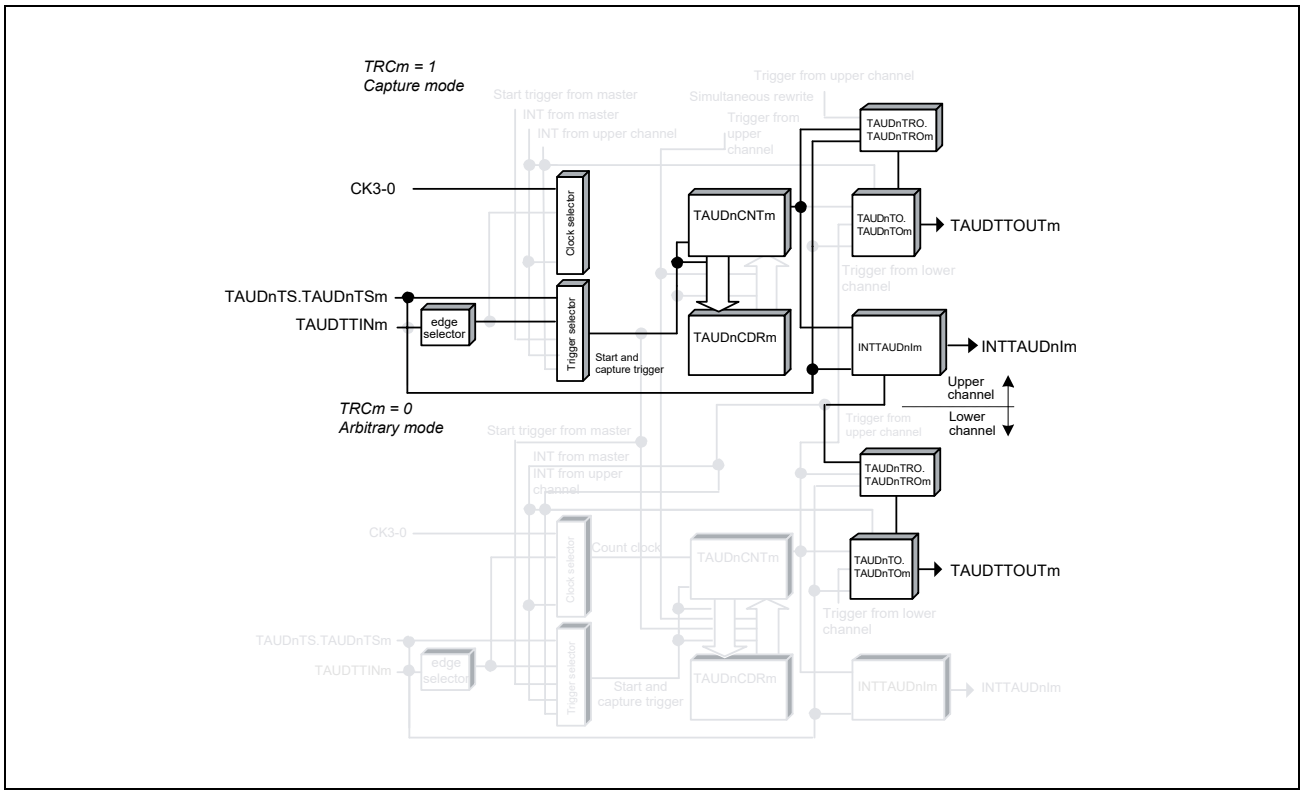


Figure 18.70 Block Diagram of Real-Time Output Function Type 2

The following settings apply to the general timing diagram.

- INTTAUDnIm not generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 0)

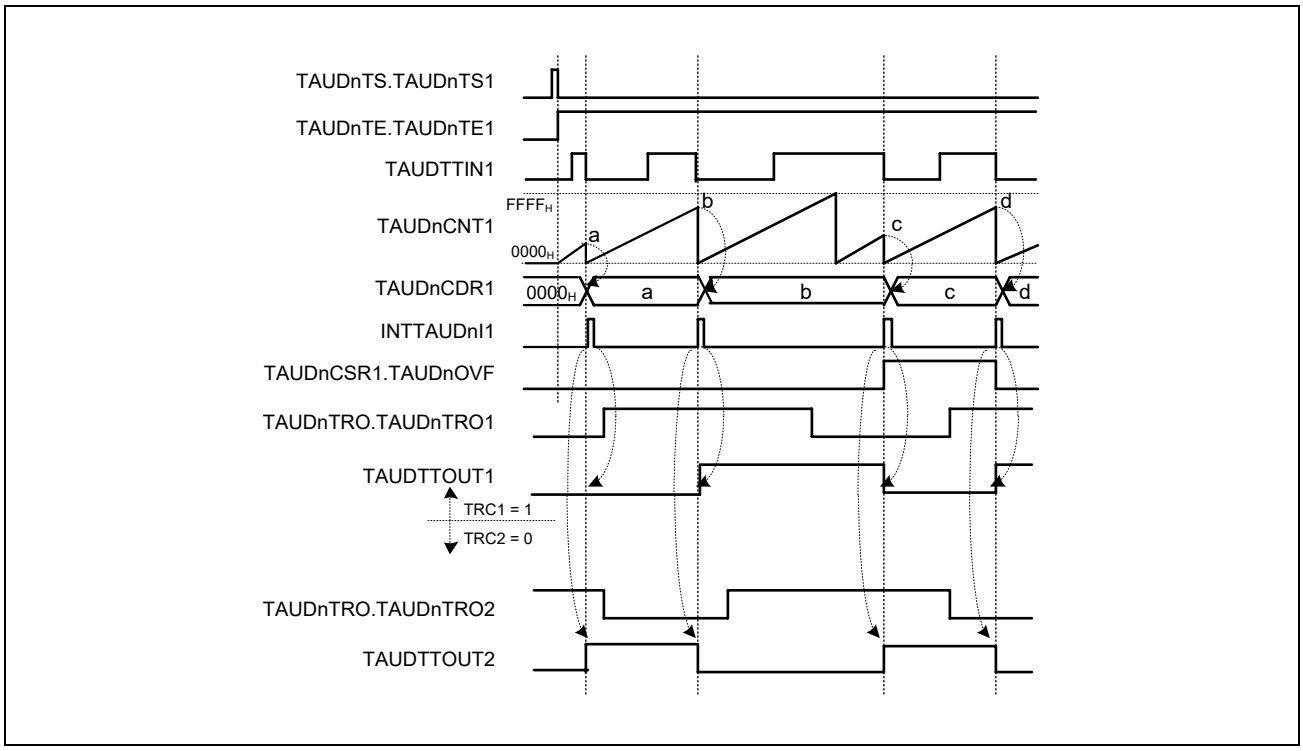


Figure 18.71 General Timing Diagram of Real-Time Output Function Type 2

(3) Register Settings for Upper Channels

(a) TAUDnCMORm for upper channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.105 Contents of TAUDnCMORm Register for Real-Time Output Function Type 2

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: Effective edge of the TAUDTTINm input signal is used as an external start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0010: Capture mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for upper channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.106 Contents of TAUDnCMURm Register for Real-Time Output Function Type 2

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode for upper channels

Table 18.107 Control Bit Settings in Independent Channel Output Mode 2 with Real-Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	1: Channel m generates a unique real-time output trigger.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for upper channels

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the real-time output function type 2. Therefore, these registers should be set to 0.

Table 18.108 Simultaneous Rewrite Settings for Real-Time Output Function Type 2

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(4) Register Settings for Lower Channels

(a) TAUDnCMORm for lower channels

The TAUDnCMORm register for lower channels is available for any setting.

(b) TAUDnCMURm for lower channels

The TAUDnCMURm register for lower channels is available for any setting.

(c) Channel output mode for lower channels

Table 18.109 Control Bit Settings for Lower Channels in Independent Channel Output Mode 2 with Real-Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode.
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation.
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output 1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for lower channels

Simultaneous rewrite registers for lower channels can be set arbitrarily.

(5) Operating Procedure for Real-Time Output Function Type 2

Table 18.110 Operating Procedure for Real-Time Output Function Type 2

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers for upper channels as described in **Table 18.105, Contents of TAUDnCMORm Register for Real-Time Output Function Type 2**, and **Table 18.106, Contents of TAUDnCMURm Register for Real-Time Output Function Type 2**. Set TAUDnCMORm and TAUDnCMURm registers for the lower channel as described in **Section (4), Register Settings for Lower Channels**. The TAUDnCDRm register functions as a capture register. (channels with TAUDnTRC.TAUDnTRCm = 1) Set channel output mode by setting the control bits as described in **Table 18.107, Control Bit Settings in Independent Channel Output Mode 2 with Real-Time Output**. Set channel output mode by setting the control bits as described in **Table 18.109, Control Bit Settings for Lower Channels in Independent Channel Output Mode 2 with Real-Time Output.**	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm = 1 on the channel with TAUDnTRC.TAUDnTRCm set to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	[Channel with TAUDnTRC.TAUDnTRCm set to 1] TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCNTm is cleared to 0000_H. If TAUDnCMORm.TAUDnMD0 is 1, INTTAUDnIm occurs.
During Operation	TAUDnTRO.TAUDnTROm can be changed at any time.	TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm input edge is detected: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated. • TAUDTTOUTm outputs the current value of the real-time output bit TAUDnTRO.TAUDnTROm. TAUDTTOUTm outputs the current value of real-time output bit TAUDnTRO.TAUDnTROm. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm, TAUDnCSRm.TAUDnOVF, and TAUDTTOUTm retain their current values.

Restart

(6) Specific Timing Diagrams

(a) Operation start and stop

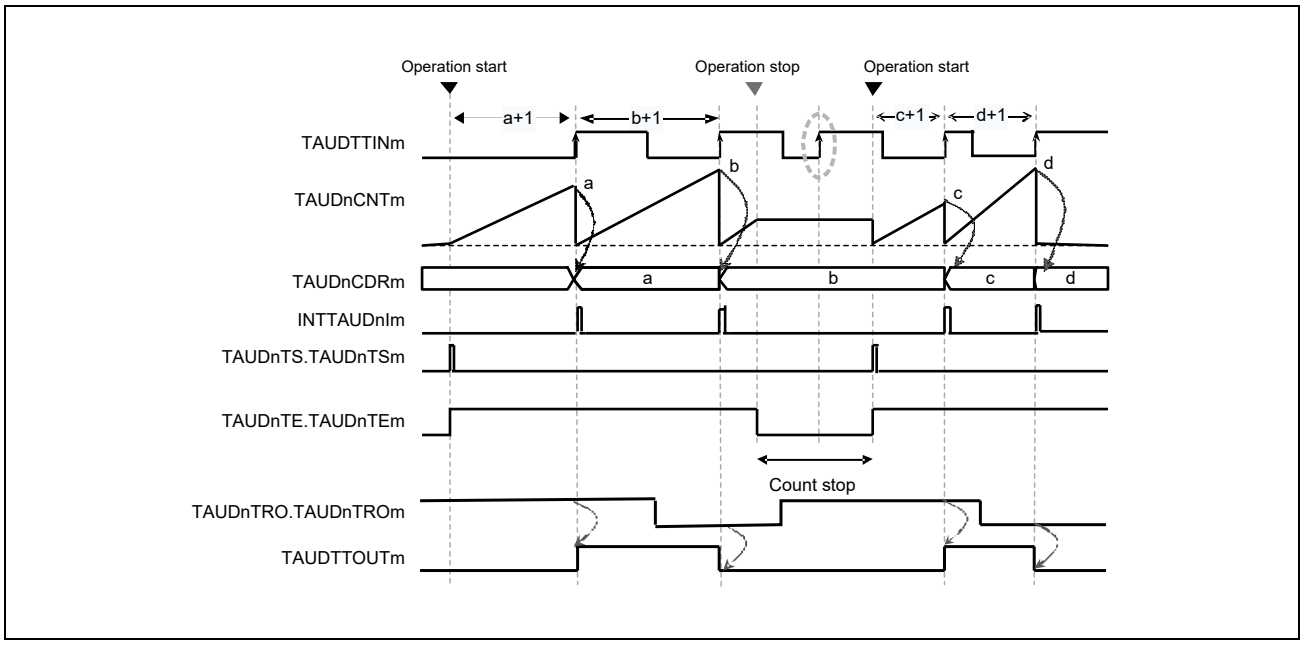


Figure 18.72 Operation Start and Stop (TAUDnCMORm.TAUDnMD0 = 0)

- When TAUDnTS.TAUDnTSm is set to 1, the counter starts counting up.
- When an effective input edge is detected, the current value of the counter is written to the data register (TAUDnCDRm) and an interrupt is generated.
- TAUDTTOUTm outputs the current value of the real-time output bit (TAUDnTRO.TAUDnTROm) and the counter resets and starts to count up again.
- The TAUDTTOUTm signal only changes when an interrupt is generated, and then only when its value is different to current value of TAUDnTRO.TAUDnTROm at the moment that the interrupt is generated.
- If the counter is stopped (TAUDnTE.TAUDnTEm = 0), effective input edges are ignored and no interrupt is generated.

18.4.11 Independent Channel Simultaneous Rewrite Functions

This section describes functions that carry out simultaneous rewrite:

- **Section 18.4.11.1, Simultaneous Rewrite Trigger Generation Function Type 1**

18.4.11.1 Simultaneous Rewrite Trigger Generation Function Type 1

(1) Overview

Summary

This function generates an interrupt on a specific channel that can be used by lower channels as a simultaneous rewrite trigger. The interrupt is generated at regular intervals.

The upper channel is a channel which generates a simultaneous rewrite trigger (TAUDnRDC.TAUDnRDCm = 1), and the lower channel is a channel which makes a simultaneous rewrite in response to the upper channel trigger (TAUDnRDC.TAUDnRDCm = 0).

Prerequisites

- Two or more channels lower than the channel used as upper channel are enabled for simultaneous rewrite (TAUDnRDE.RDEm = 1).
- The operating mode for the upper channel should be set to interval timer mode (see **Table 18.111, Contents of TAUDnCMORm Register for Simultaneous Rewrite Trigger Generation Function Type 1**).
- For the operating mode that can be set for lower channels, see **Table 18.42, Channel Operation Functions and Methods They Use**.
- TAUDTTOUTm is not used for any channel in this function.

Functional description

The counter operation is enabled by setting the channel trigger bits (TAUDnTS.TAUDnTSm) for upper and lower channels to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of the data register buffer for upper channels (TAUDnCDRm buf) is loaded into the counter (TAUDnCNTm) and the counter starts to count down from this value. The counter for lower channels start to count according to the selected operating mode.

Once the counter reaches 0000_H, an interrupt occurs on the channel. The current value of the corresponding TAUDnCDRm buffer is loaded into TAUDnCNTm to continue operation subsequently.

If the channel where an interrupt occurs is specified as a trigger channel for simultaneous rewrite (TAUDnRDC.RDCm = 1) and is an upper channel, simultaneous rewrite takes place on all lower channels in which simultaneous rewrite is currently possible (TAUDnRSF.RSFm = 1).

The values of the data registers are copied to the corresponding data register buffers. Each time a counter starts to count down, it reads the value in the data register buffer and counts down from this value.

The value of a data register can be changed at any time, but it is only transferred to the corresponding data register buffer when simultaneous rewrite occurs.

Conditions

- The channel which is monitored for INTTAUDnIm occurrence is specified by setting TAUDnRDC.TAUDnRDCm = 1 for the corresponding channel. The TAUDnRDC.TAUDnRDCm bit should be 0 for all other channels in which simultaneous rewrite should take place.
- If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not generated. For details, see **Section 18.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

(2) Equations

Simultaneous rewrite trigger generation cycle = count clock cycle × (TAUDnCDRm + 1)

To control simultaneous rewrite, the following condition should be satisfied:

[PWM]

TAUDnCDRm
= [(value of TAUDnCDRm of master channel subject to simultaneous rewrite + 1) × number of interrupts] – 1

[Triangle PWM]

TAUDnCDRm
= [(value of TAUDnCDRm of master channel subject to simultaneous rewrite + 1) × 2 × number of interrupts] – 1

That is, the ratio of TAUDnCDRm + 1 and TAUDnCDRm_master + 1 of master channel subject to simultaneous rewrite should be an integer. This integer corresponds to the number of interrupts.

For triangle PWM, remember that the cycle doubles.

(3) Block Diagram and General Timing Diagram

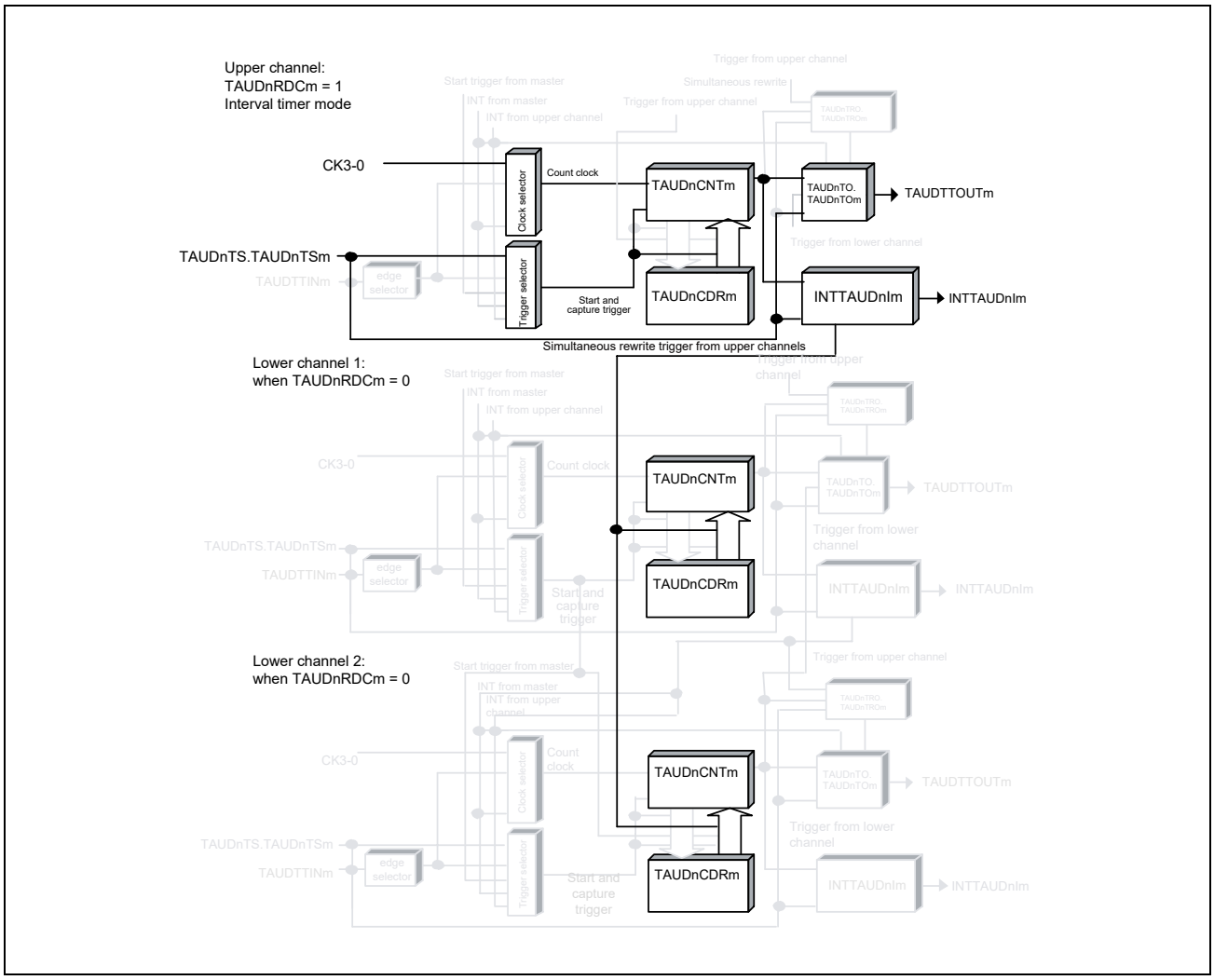


Figure 18.73 Block Diagram of Simultaneous Rewrite Trigger Generation Function Type 1

The following settings apply to the general timing diagram.

- INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

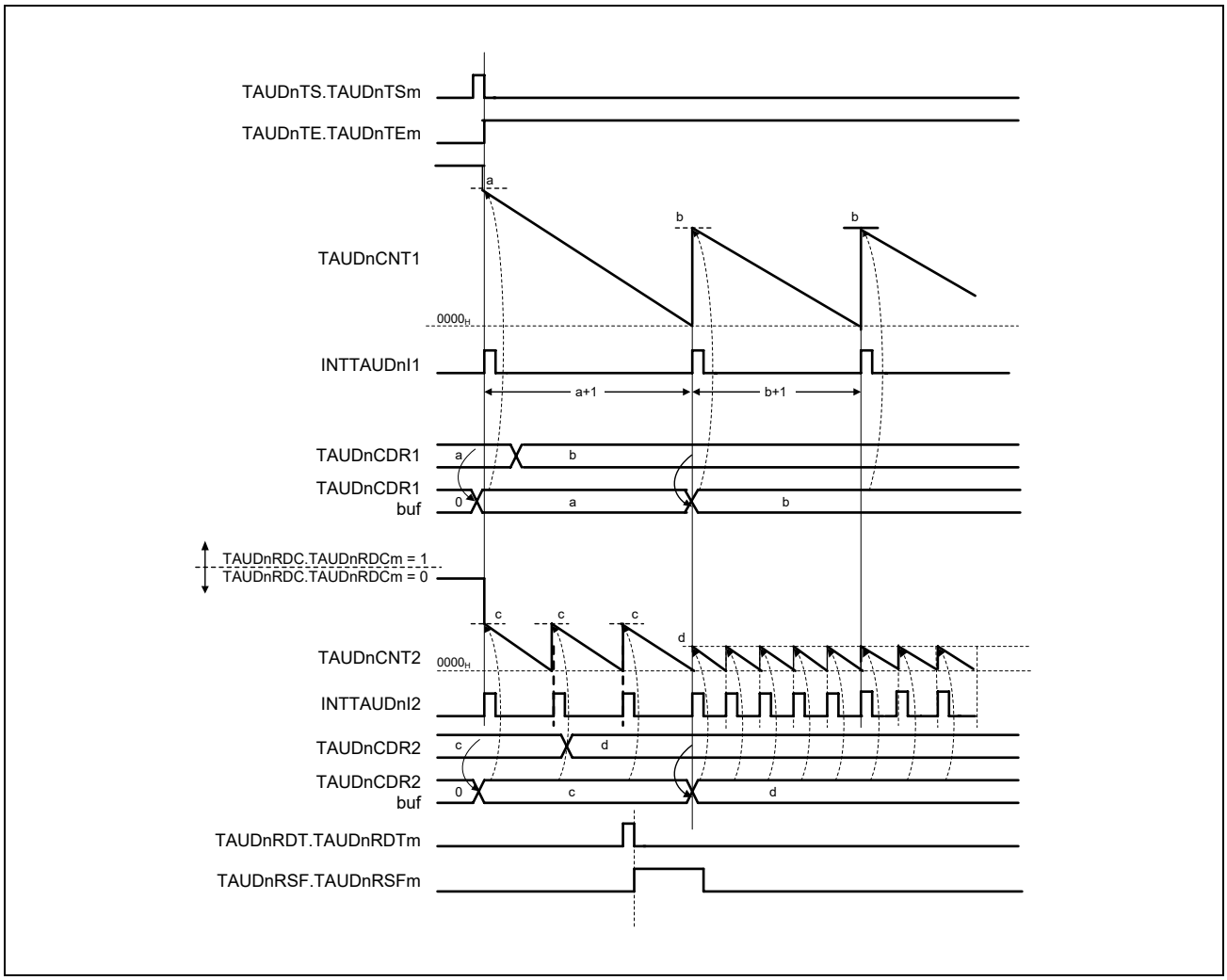


Figure 18.74 General Timing Diagram of Simultaneous Rewrite Trigger Generation Function Type 1

(4) Register Settings for Upper Channels

(a) TAUDnCMORm for upper channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.111 Contents of TAUDnCMORm Register for Simultaneous Rewrite Trigger Generation Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for upper channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.112 Contents of TAUDnCMURm Register for Simultaneous Rewrite Trigger Generation Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for upper channels

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite for upper channels

Table 18.113 Simultaneous Rewrite Settings for Simultaneous Rewrite Trigger Generation Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	1: Selects one of upper channels as simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Loads a simultaneous rewrite control signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	1: Monitors INTTAUDnIm signal which triggers a simultaneous rewrite on the channel.

(5) Register Settings for Lower Channels

(a) TAUDnCMORm for lower channels

TAUDnCMORm register for lower channels must follow the TAUDnCMORm register settings in the operating mode which can be set (see **Table 18.42, Channel Operation Functions and Methods They Use**).

(b) TAUDnCMURm for lower channels

TAUDnCMURm register for lower channels must follow the TAUDnCMURm register settings in the operating mode which can be set (see **Table 18.42, Channel Operation Functions and Methods They Use**).

(c) Channel output mode for lower channels

Output can be made according to the operating mode setting (master/slave) for lower channels. As for the available function for simultaneous rewrite trigger generation function type 1, see **Table 18.42, Channel Operation Functions and Methods They Use**.

(d) Simultaneous rewrite for lower channels

Table 18.114 Simultaneous Rewrite Settings for Lower Channels in Simultaneous Rewrite Trigger Generation Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	1: Selects one of upper channels as simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Loads a simultaneous rewrite control signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for Simultaneous Rewrite Trigger Generation Function Type 1

Table 18.115 Operating Procedure for Simultaneous Rewrite Trigger Generation Function Type 1

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers for the upper channel as described in **Table 18.111, Contents of TAUDnCMORm Register for Simultaneous Rewrite Trigger Generation Function Type 1**, and **Table 18.112, Contents of TAUDnCMURm Register for Simultaneous Rewrite Trigger Generation Function Type 1**. Set TAUDnCMORm and TAUDnCMURm registers for lower channels as described in **Section (5), Register Settings for Lower Channels**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCDRm value is loaded into TAUDnCNTm. If TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm occurs.
During Operation	TAUDnRDT.TAUDnRDTm and TAUDnCDR.CDRm is changeable. TAUDnRSF.TAUDnRSFm can be always read.	TAUDnCNTm counts down. When the counter reaches 0000_H: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated. If INTAUDnIm occurs on the channel where TAUDnRDC.TAUDnRDCm is set to 1, simultaneous rewrite is controlled. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its current value.

Restart

18.4.12 Synchronous Channel Operation Functions

This section lists all the synchronous channel operation functions provided by the timer array unit D. For a general overview of synchronous channel operation, see **Section 18.2, Overview**.

This section describes functions that generate PWM signals at regular intervals.

- **Section 18.4.12.1, PWM Output Function**
- **Section 18.4.12.2, One-Shot Pulse Output Function**
- **Section 18.4.12.3, Trigger Start PWM Output Function**
- **Section 18.4.12.4, Delay Pulse Output Function**
- **Section 18.4.12.5, Offset Trigger Output Function**
- **Section 18.4.12.6, A/D Conversion Trigger Output Function Type 1**
- **Section 18.4.12.7, Triangle PWM Output Function**
- **Section 18.4.12.8, Triangle PWM Output Function with Dead Time**
- **Section 18.4.12.9, A/D Conversion Trigger Output Function Type 2**
- **Section 18.4.12.10, Interrupt Request Signals Culling Function**
- **Section 18.4.12.11, One-Phase PWM Output Function**

18.4.12.1 PWM Output Function

(1) Overview

Summary

This function generates multiple PWM outputs by using a master and multiple slave channels. It enables the pulse cycle (frequency) and the duty cycle of the TAUDTTOUTm to be set. The pulse cycle is set in the master channel. The duty cycle is set in the slave channel.

Prerequisites

- It requires two channels.
- The operating mode for the master channel should be set to interval timer mode (see **Table 18.116, Contents of TAUDnCMORm Register for Master Channels of PWM Output Function**).
- The operating mode for the slave channels should be set to one count mode (see **Table 18.119, Contents of TAUDnCMORm Register for Slave Channels of PWM Output Function**).
- TAUDTTOUTm is not used with the master channel of this function.
- The channel output mode for the slave channels should be set to Synchronous Channel Output Mode 1 (see **Section 18.4.4, Channel Output Modes**).

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is loaded into TAUDmCNT, and the counter starts counting down from the TAUDnCDRm value. If an INTTAUDnlm occurs on the master channel and TAUDTTOUTm (slave) is set/reset, PWM output is made.

- Master channel:
  When the master channel counter reaches 0000_H and the pulse cycle time has passed, INTTAUDnIm occurs. The counter loads TAUDnCDRm value into TAUDnCNTm and counts down.
- Slave channels:
  When INTTAUDnIm occurs on the master channel, the counter operation of the slave channel is triggered. The current value of TAUDnCDRm (slave) is loaded into TAUDnCNTm (slave) and the counter starts counting down from the TAUDnCDRm value. TAUDTTOUTm signal is set to the active level.
  When the counter reaches to 0000_H (duty time has elapsed), INTTAUDnIm occurs and a TAUDTTOUTm signal is set to an inactive level. The counter is reset to $FFFF_H$ and waits for the next INTTAUDnlm (start of the next pulse cycle) of the master channel.

The counter can stop operating by setting the TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but their values are retained. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

Simultaneous rewrite can be used with this function. See **Section 18.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

Duty cycle [%] = (TAUDnCDRm (slave) / (TAUDnCDRm (master) + 1) × 100

- Duty cycle = 0%
  TAUDnCDRm (slave) = 0000_H
- Duty cycle = 100%
  TAUDnCDRm (slave) ≥ TAUDnCDRm (master) + 1

(3) Block Diagram and General Timing Diagram

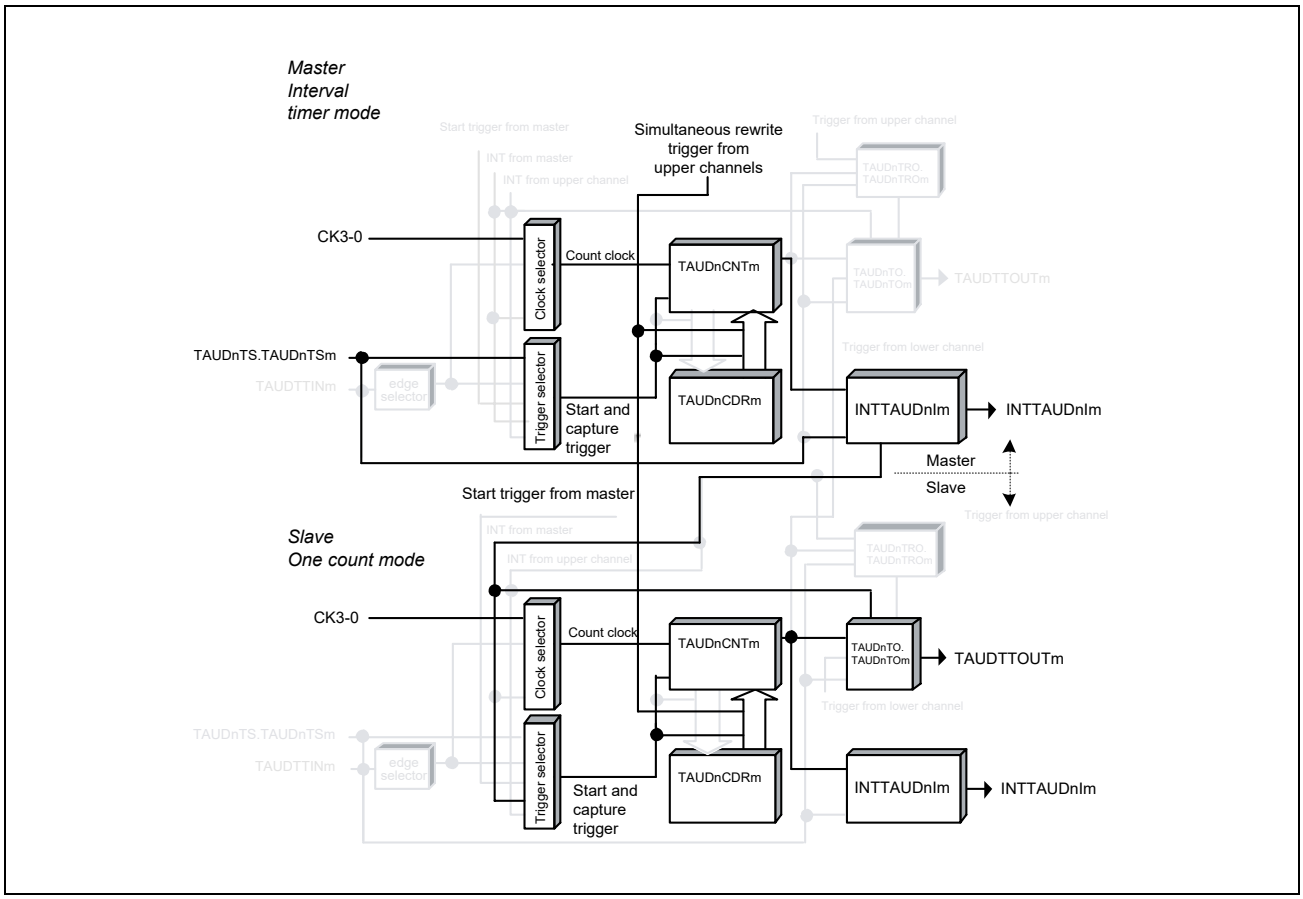


Figure 18.75 Block Diagram of PWM Output Function

The following settings apply to the general timing diagram.

- Slave channels: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

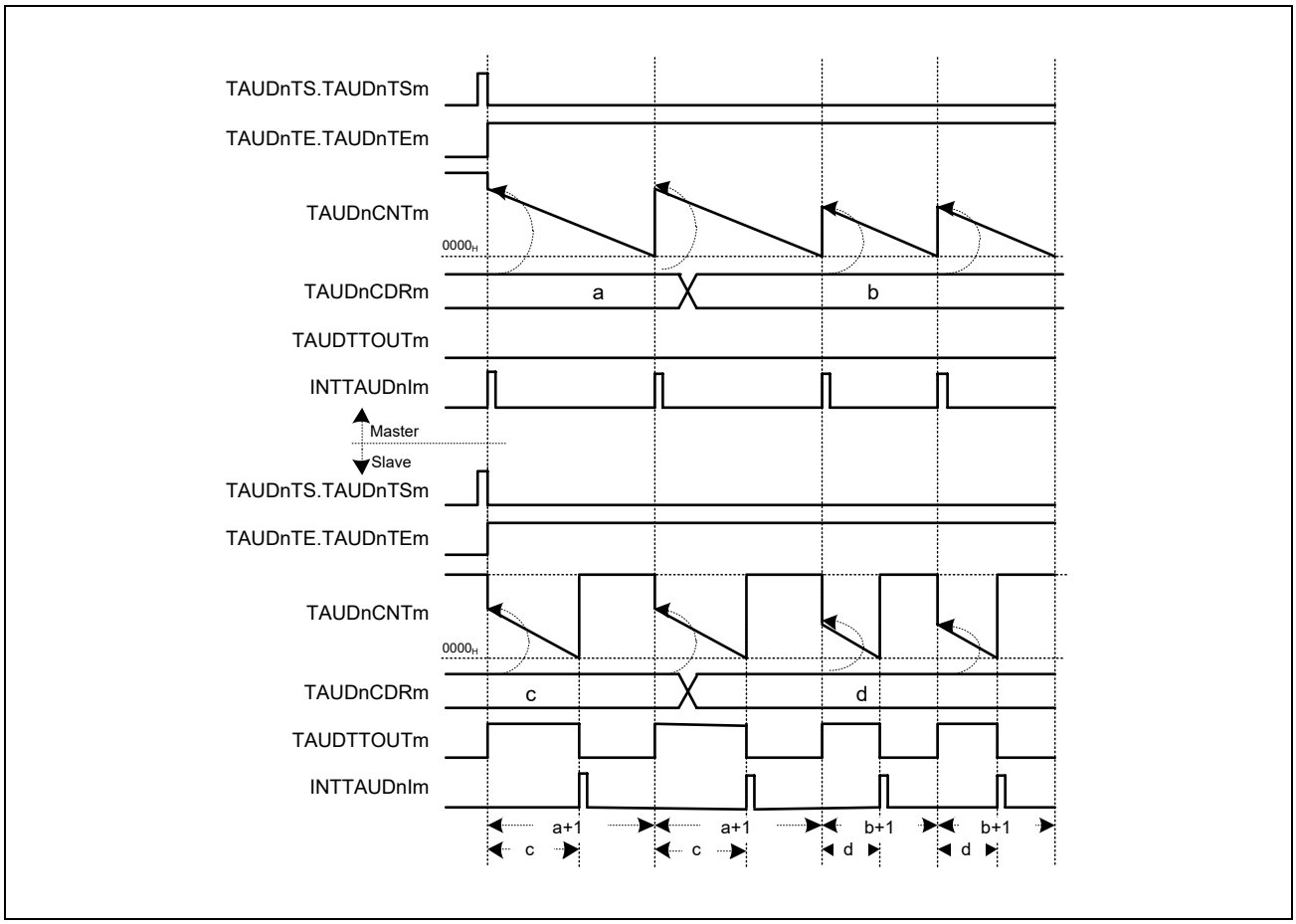


Figure 18.76 General Timing Diagram of PWM Output Function

NOTE

- The interval between the slave channel starting to count and an interrupt being generated is the value of corresponding TAUDnCDRm, whereas for the master channel the interval is the value of the corresponding TAUDnCDRm + 1.
- TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.116 Contents of TAUDnCMORm Register for Master Channels of PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.117 Contents of TAUDnCMURm Register for Master Channels of PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

The channel output mode is not used with this function.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.118 Simultaneous Rewrite Settings for Master Channels of the PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

NOTE

Use with TAUDnRDS.TAUDnRDSm bit = 1 requires an upper channel higher than the master channel that operates with the **Section 18.4.11.1, Simultaneous Rewrite Trigger Generation Function Type 1**.

Conduct operation settings under the following conditions:

- Simultaneous rewrite trigger output function type 1 setting channel: TAUDnRDCm = 1, TAUDnRDSm = 1
  TAUDnCDRm settings for this channel are as follows:
  = ((TAUDnCDRm setting for the master channel targeted for simultaneous rewrite + 1) × interrupt count) – 1
- Master channels: TAUDnRDCm = 0, TAUDnRDSm = 1
- Slave channels: TAUDnRDCm = 0, TAUDnRDSm = 1

If TAUDnCDRm (slave) setting > TAUDnCDRm (master) setting + 1, the duty value (which exceeds 100%) is aggregated to be 100% output.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.119 Contents of TAUDnCMORm Register for Slave Channels of PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Start trigger during operation is valid.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.120 Contents of TAUDnCMURm Register for Slave Channels of PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels

Table 18.121 Control Bit Settings in Synchronous Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.122 Simultaneous Rewrite Settings for Slave Channels of PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for PWM Output Function

Table 18.123 Operating Procedure for PWM Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm is generated on the master channel and TAUDTTOUTm (slave) is set.
During operation	TAUDnCDRm can be changed at any time. TAUDnTOL.TAUDnTOLm can be changed. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCNTm of master channel loads TAUDnCDRm value and counts down. When the counter reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCDRm value is loaded into TAUDnCNTm (master) to continue count operation. • TAUDnCDRm value is loaded into TAUDnCNTm (slave) to perform counting down. • TAUDTTOUTm (slave) is set to the active level. If TAUDnCNTm (slave) reaches 0000_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is set to an inactive level. In addition, the counter of slave channel stops.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Duty cycle = 0%

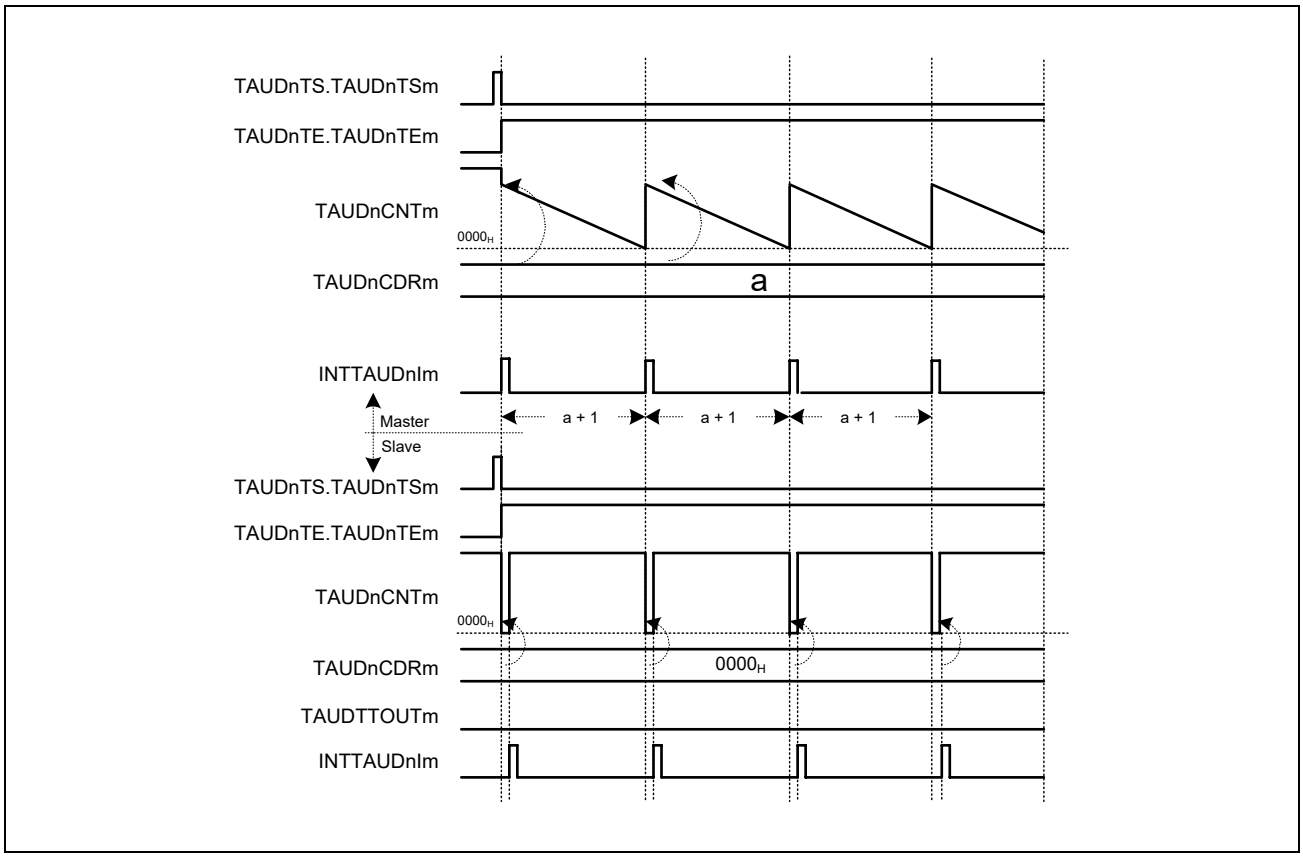


Figure 18.77 TAUDnCDRm (Slave) = 0000_H, Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0)

- Every time the master channel generates an interrupt (INTTAUDnIm), 0000_H is loaded in to TAUDnCNTm (slave). Therefore, TAUDnCNTm (slave) cannot start to count and TAUDTTOUTm remains inactive.
- TAUDnCDRm value is loaded into TAUDnCNTm (slave) to generate an interrupt.

(b) Duty cycle = 100%

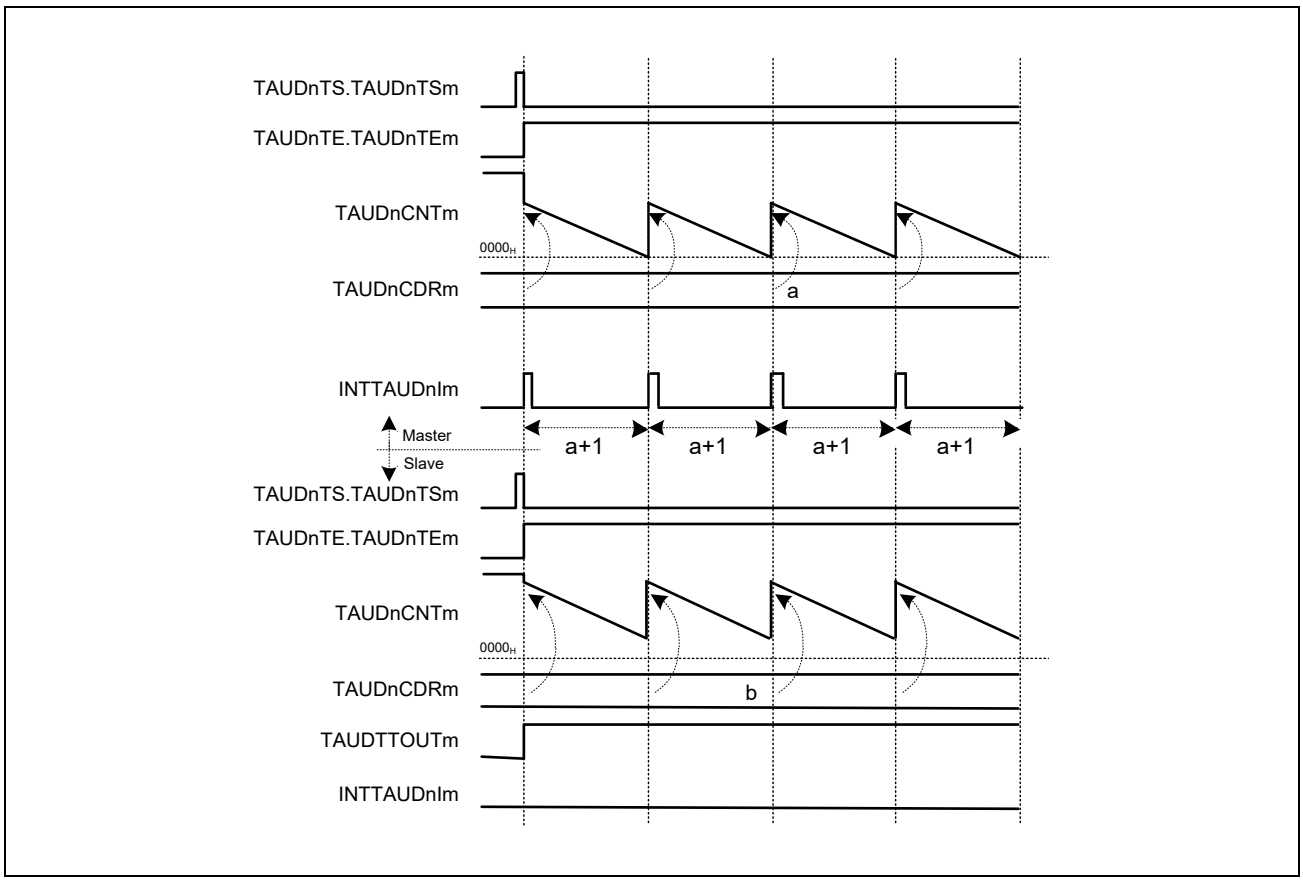


Figure 18.78 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1,
Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0)

- If TAUDnCDRm (slave) value is greater than TAUDnCDRm (master) value, the slave channel counter does not reach 0000_H and consequently, no interrupt occurs. TAUDTTOUTm remains active.

(c) Operation stop and restart

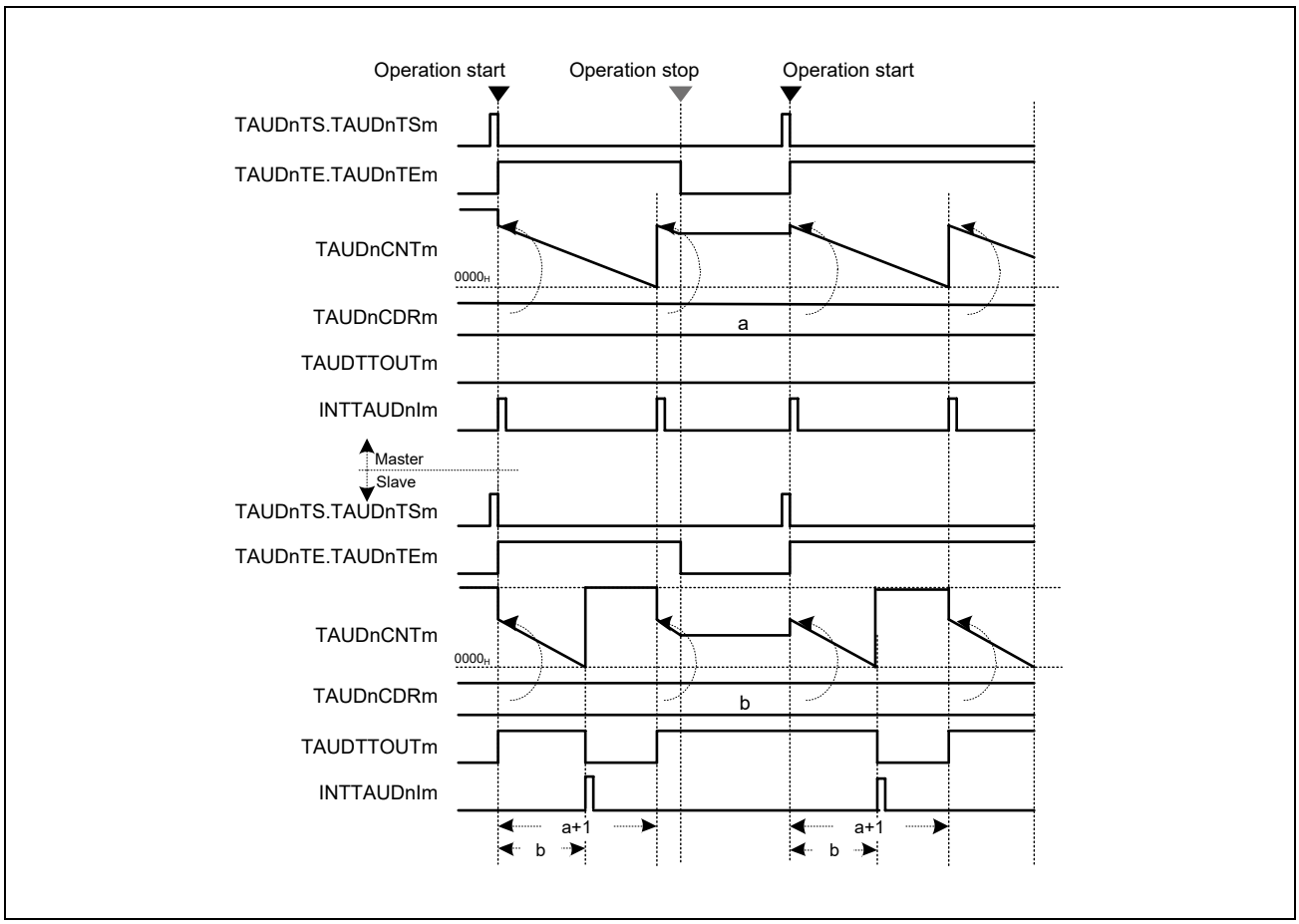


Figure 18.79 Operation Stop and Restart
Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0)

- The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm and TAUDTTOUTm of all channels stop and the current values are retained. No interrupts are generated.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm of master and slave channels to 1. TAUDnCNTm of master and slave channels reload the current values of TAUDnCDRm and start to count down from these values.

18.4.12.2 One-Shot Pulse Output Function

(1) Overview

Summary

This function outputs a signal pulse with a specific pulse width and delay time (both defined relative to an external input signal pulse or software trigger) by using a master and a slave channel. The delay time is specified using the master channel. The pulse width is specified using the slave channel.

Prerequisites

- It requires two channels.
- The operating mode for the master channel should be set to one-count mode (see **Table 18.124, Contents of TAUDnCMORm Register for Master Channels of One-Shot Pulse Output Function**).
- The operating mode for slave channels should be set to pulse one-count mode (see **Table 18.127, Contents of TAUDnCMORm Register for Slave Channels of One-Shot Pulse Output Function**).
- TAUDTTOUTm is not used with the master channel of this function.
- The channel output mode for the slave channel should be set to synchronous channel output mode 2 (see **Section 18.4.4, Channel Output Modes**).
- TAUDTTINm (master) has to be detected while TAUDnCNTm (master) and TAUDnCNTm (slave) await a trigger. Furthermore, the slave is only triggered by an interrupt from the master channel and not by TAUDTTINm (slave).
- If only a software trigger is to be used, do not select the alternative pin function TAUDTTINm.

Functional description

The counters are enabled by setting the channel trigger bits (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm, enabling count operation.

- Master channel:
  When the next effective TAUDTTINm input edge or a software trigger (TAUDnTS.TAUDnTSm = 1 (m: master channel number) when TAUDnTE.TAUDnTEm = 1) is detected, the current value of TAUDnCDRm is loaded into TAUDnCNTm. The counter starts to count down from this value. If TAUDnCMORm.TAUDnMD0 = 0, a trigger (TAUDTTINm) which is detected within the delay time is ignored.
  When the counter of master channel reaches 0000_H, INTTAUDnIm is generated. The counter is reset to $FFFF_H$ and waits for the next effective TAUDTTINm input edge or a software trigger (TAUDnTS.TAUDnTSm = 1 (m: master channel number) when TAUDnTE.TAUDnTEm = 1).
- Slave channels:
  INTTAUDnIm generated on master channel triggers the counter operation of slave channel. The current value of TAUDnCDRm (slave) is loaded into TAUDnCNTm (slave). The counter starts counting down from this value. An interrupt occurs and the TAUDTTOUTm signal is set. When the counter reaches 0001_H, INTTAUDnIm is generated and TAUDTTOUTm signal is reset. The counter stops at 0000_H and waits for the next INTTAUDnIm of master channel.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but their values are retained. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Setting TAUDnTS.TAUDnTSm to 1 while counting allows the counter to restart counting of master channel without making a stop (forced restart).

Conditions

- If TAUDnCMORn.TAUDnMD0 of master channel is set to 0, TAUDTTINm input edges detected during counting are ignored.
- Simultaneous rewrite can be used with this function. See **Section 18.4.3, Simultaneous Rewrite**.

(2) Equations

Delay from trigger input to pulse output = (TAUDnCDRm (master) + 1) × count clock cycle

Pulse width = (TAUDnCDRm (slave)) × count clock cycle

(3) Block Diagram and General Timing Diagram

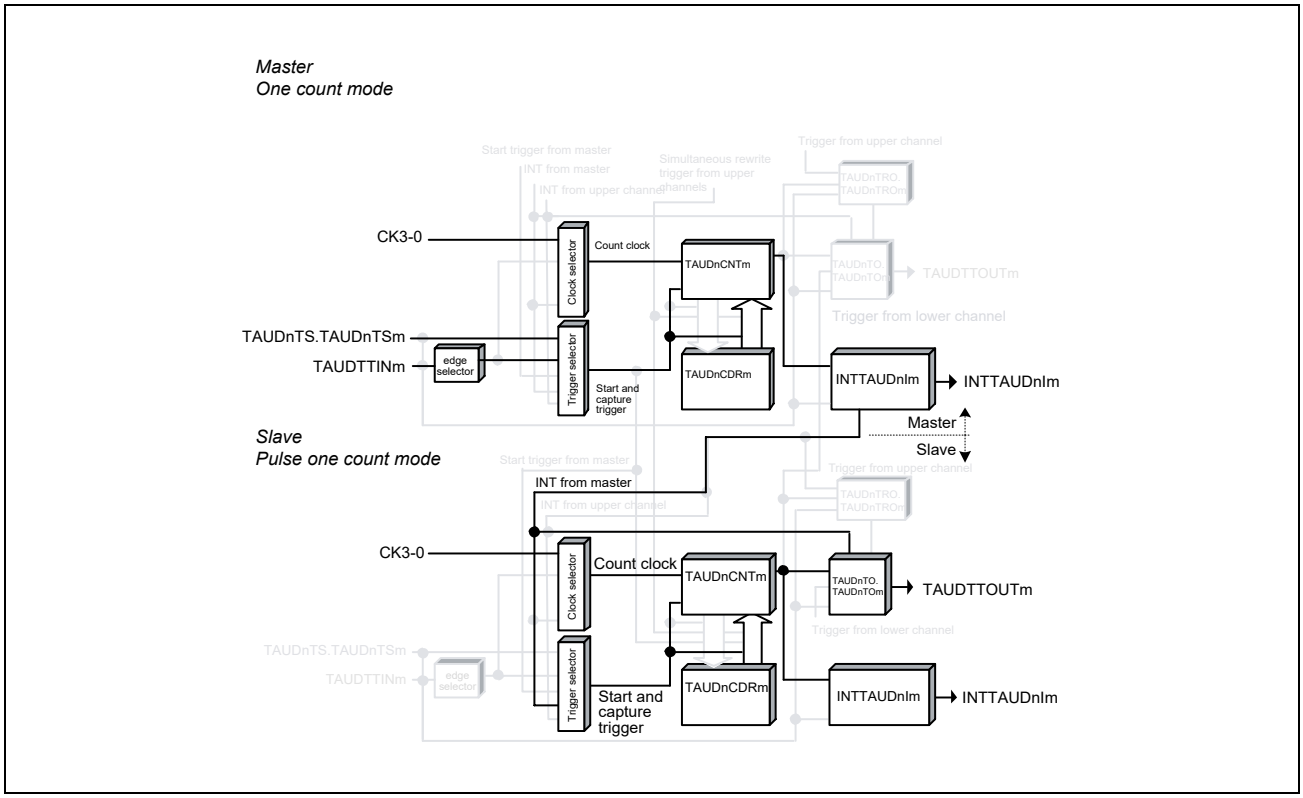


Figure 18.80 Block Diagram of One-Shot Pulse Output Function

The settings in **Figure 18.81, General Timing Diagram of One-Shot Pulse Output Function (in the Case of an External Input Signal)** and **Figure 18.82, General Timing Diagram of One-Shot Pulse Output Function (in the Case of a Software Trigger)** are as follows.

- Start trigger detection is disabled during counting (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

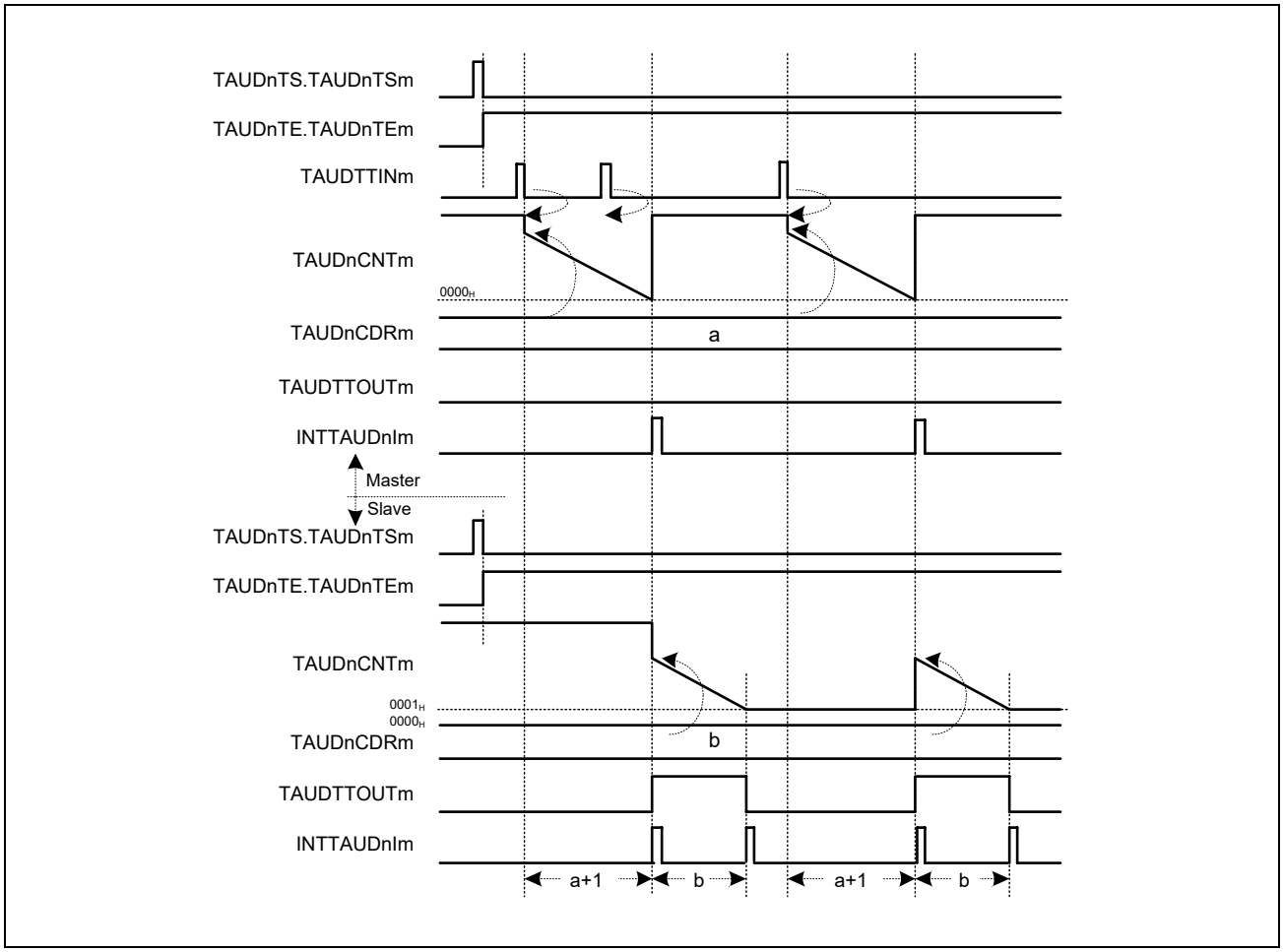


Figure 18.81 General Timing Diagram of One-Shot Pulse Output Function (in the Case of an External Input Signal)

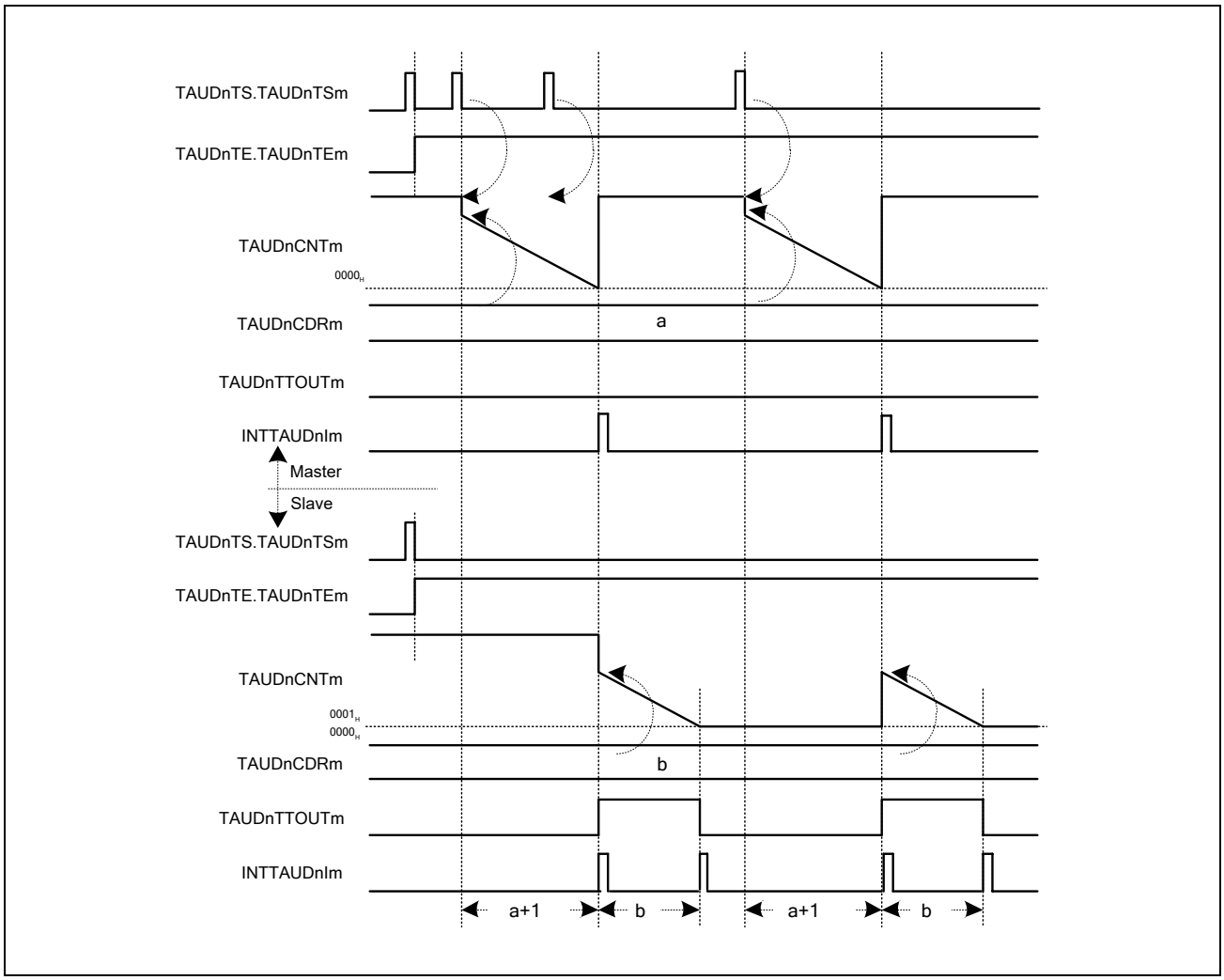


Figure 18.82 General Timing Diagram of One-Shot Pulse Output Function (in the Case of a Software Trigger)

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.124 Contents of TAUDnCMORm Register for Master Channels of One-Shot Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as the start trigger
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	0: Disables detection of start trigger during count operation. 1: Enables start trigger detection while counting. TAUDnMD0 bit of master and slave channels should have the same value.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.125 Contents of TAUDnCMURm Register for Master Channels of One-Shot Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used with this function.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.126 Simultaneous Rewrite Settings for Master Channels of One-Shot Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.127 Contents of TAUDnCMORm Register for Slave Channels of One-Shot Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1010: Pulse one-count mode
0	TAUDnMD0	0: Disables detection of start trigger during count operation. 1: Enables start trigger detection while counting. TAUDnMD0 bit of master and slave channels should have the same value.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.128 Contents of TAUDnCMURm Register for Slave Channels of One-Shot Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for the slave channel

Table 18.129 Control Bit Settings in Independent Channel Output Mode 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.130 Simultaneous Rewrite Settings for Slave Channels of One-Shot Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for One-Shot Pulse Output Function

Table 18.131 Operating Procedure for One-Shot Pulse Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the master channel awaits a TAUDTTINm input.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	When an effective TAUDTTINm input edge is detected, TAUDnCDRm value of master channel is loaded into TAUDnCNTm to start countdown. When the counter reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCNTm (master) returns to $FFFF_H$ and awaits the next effective TAUDTTINm input edge. • The TAUDnCDRm value is reloaded into TAUDnCNTm (slave) to start counting down. • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is set. When TAUDnCNTm (slave) reaches 0001_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is reset. In addition, the counter of slave channel stops. If TAUDTTINm input is detected on the master channel during count operation and TAUDnCMORm.TAUDnMD0 = 0, the input is ignored.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) TAUDnCDRm (master) = 0000_H

The following settings apply to this diagram:

- Disables detection of start trigger during count operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

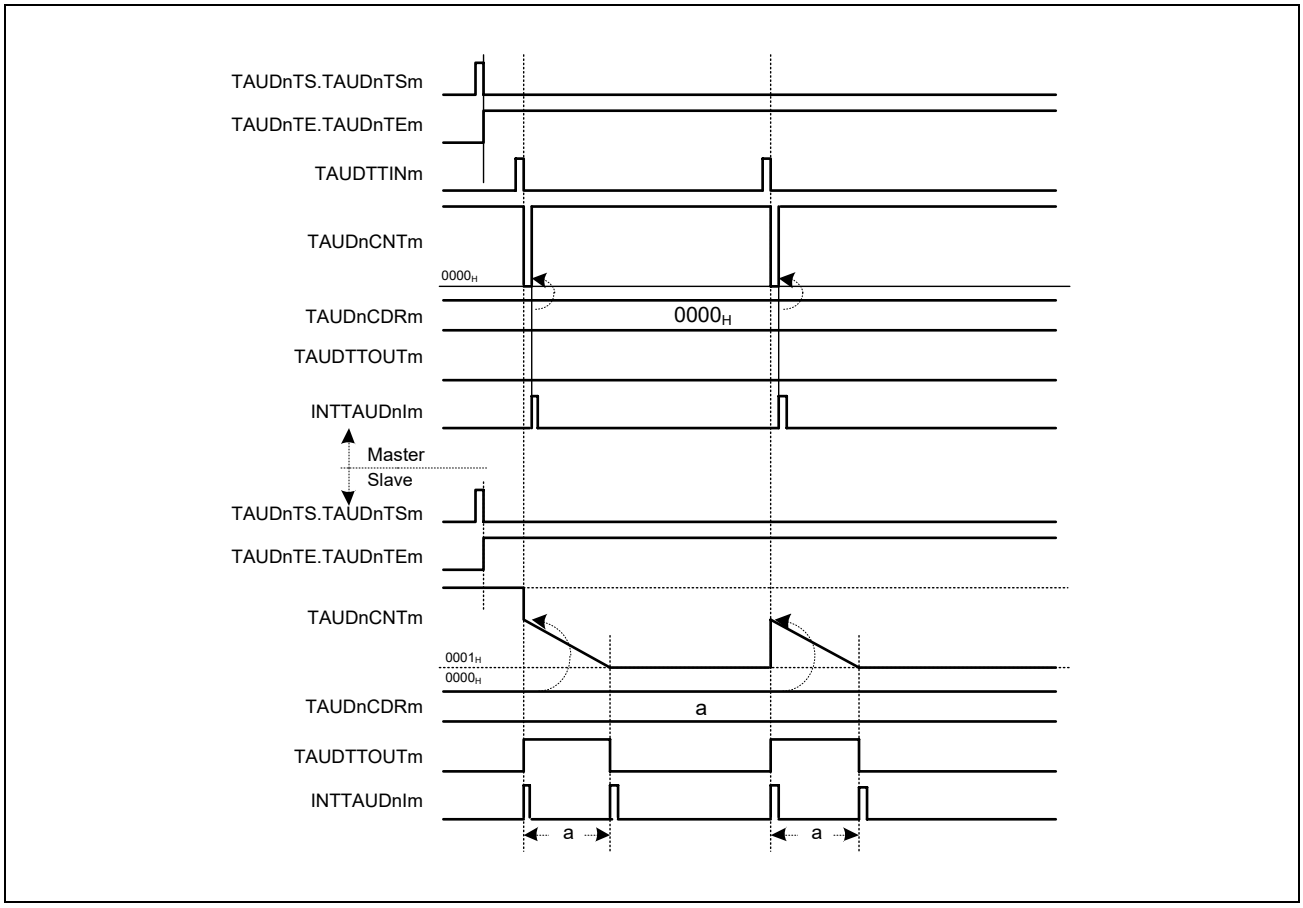


Figure 18.83 TAUDnCDRm (Master) = 0000_H

- When an effective TAUDTTINm input edge is detected, the value 0000_H is written to TAUDnCNTm (master). The counter is set to 0000_H for one count and returns to $FFFF_H$. Thus the slave channel starts to count down one count clock later to TAUDTTINm (master).

(b) TAUDnCDRm (slave) = 0000_H

The following settings apply to this diagram:

- Disables detection of start trigger during count operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

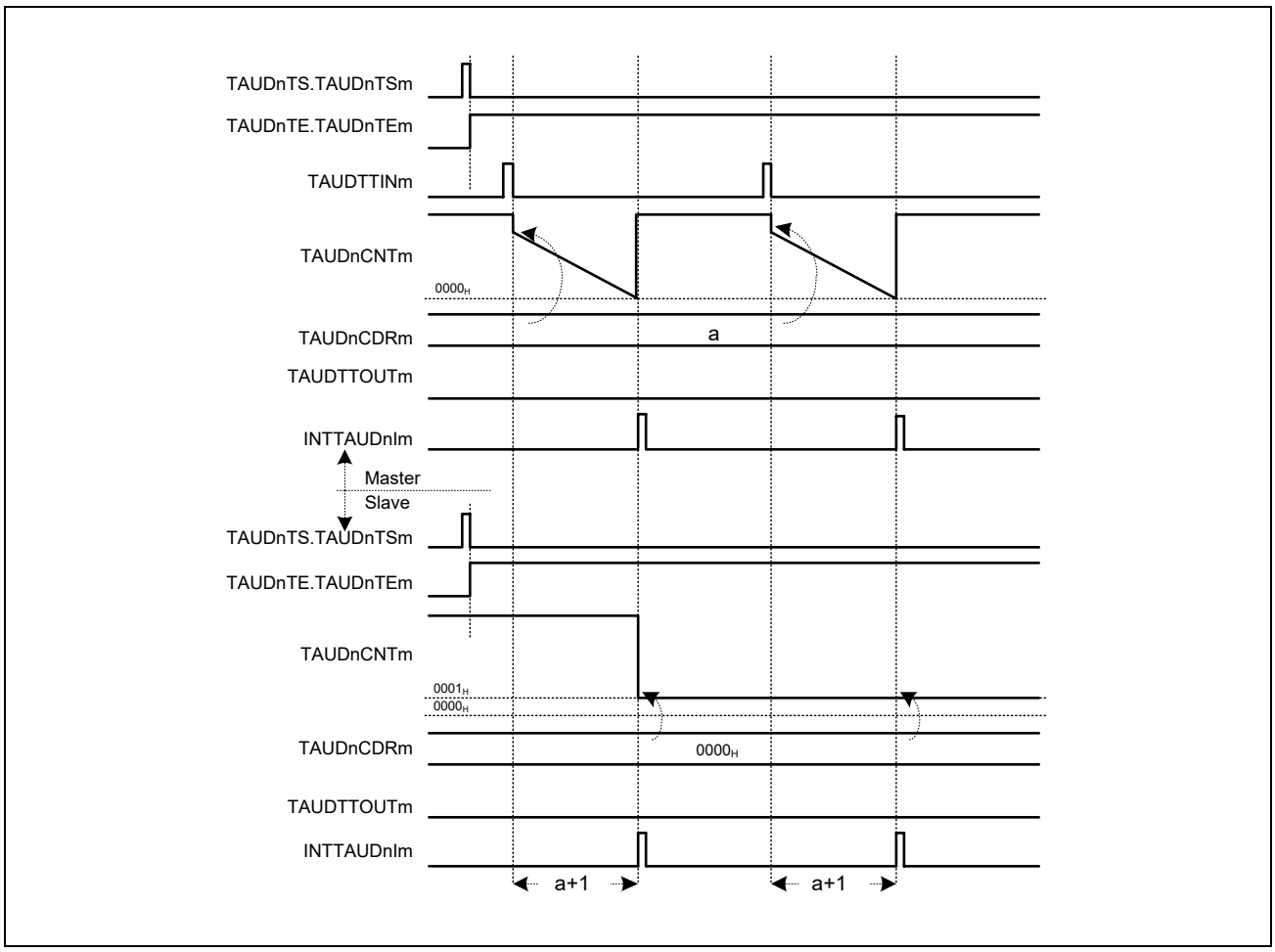


Figure 18.84 TAUDnCDRm (Slave) = 0000_H

- TAUDTTOUTm remains at not active state, because the pulse width is zero.

(c) TAUDnCMORm.TAUDnMD0 = 1

The following settings apply to this diagram:

- Enables start trigger detection while counting. (TAUDnCMORm.TAUDnMD0 = 1)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

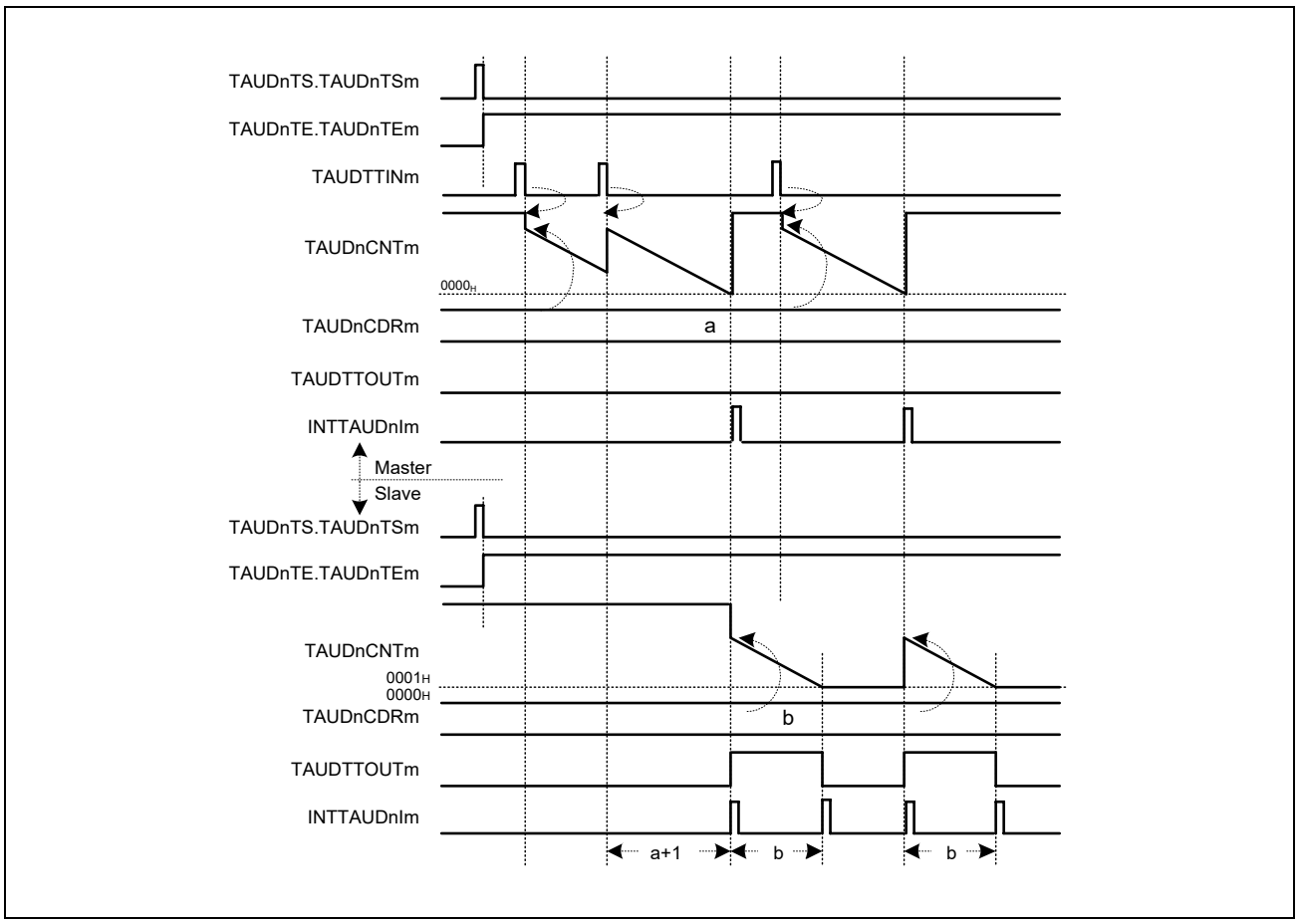


Figure 18.85 TAUDnCMORm.TAUDnMD0 = 1

- If an effective TAUDTTINm input edge is detected while the counter of the master channel counts down, TAUDnCNTm reloads the value of TAUDnCDRm. The counter restarts to count down. This means the delay is extended by the value of TAUDnCNTm at the time an effective TAUDTTINm input edge is detected.

(d) Restarting the master channel while the slave channel is counting

The following settings apply to this diagram:

- Disables detection of start trigger during count operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

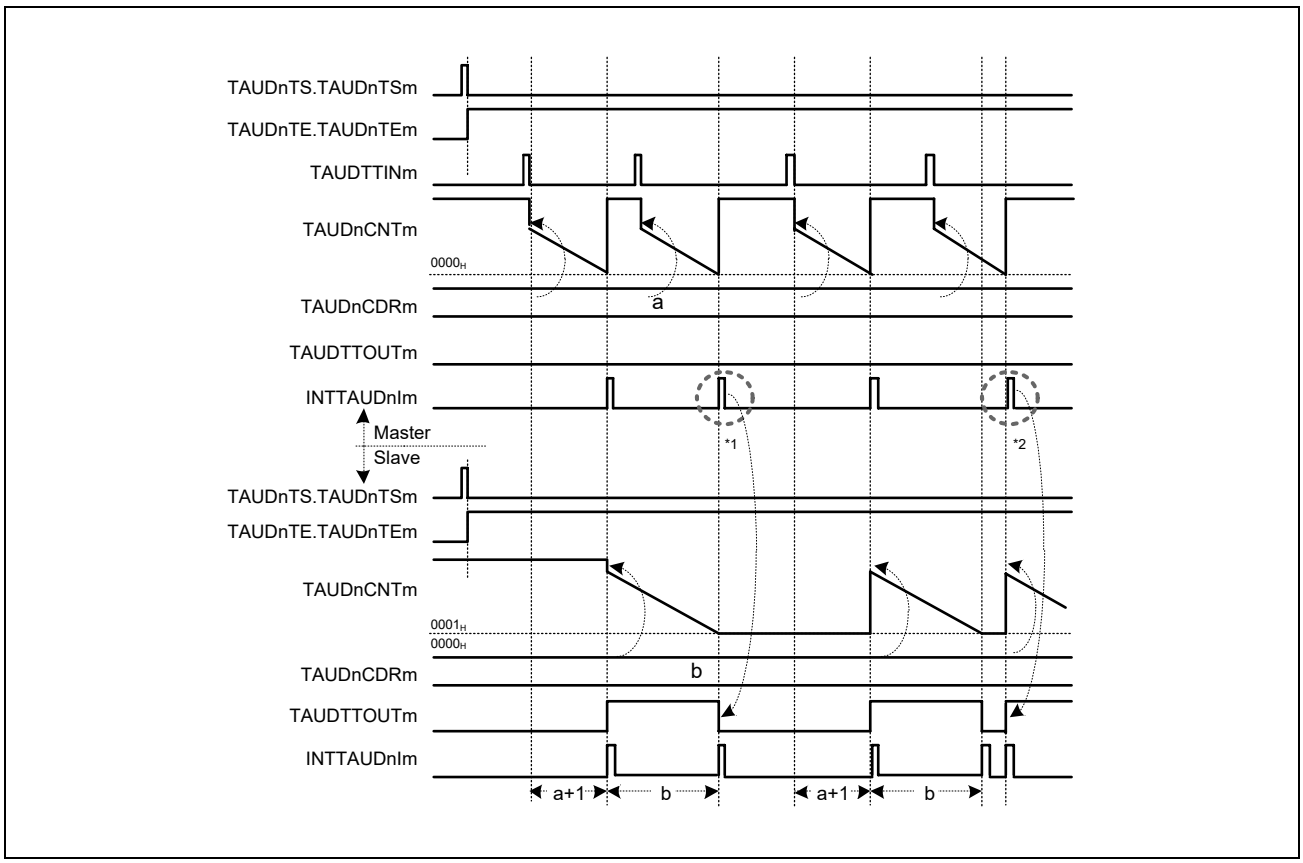


Figure 18.86 Interval of TAUDTTINm ≤ delay time + pulse width + 1

- If the master channel generates an interrupt before the counter of the slave channel has reached 0001_H or exactly when 0001_H is reached (*1), the interrupt (master) is ignored.
- If an interrupt of the master channel occurs when the counter of the slave channel awaits the next trigger, the value of TAUDnCDRm (slave) is reloaded. An interrupt is generated and TAUDTTOUTm toggles. If TAUDnCNTm (master) has started to count down while the TAUDnCNTm (slave) is still counting (*2), TAUDTTOUTm is not output with the expected delay time.
- To generate the correct one-shot pulse, the start trigger for the master channel must be detected while the master and slave channels are waiting for the start trigger, and not while they are counting.

18.4.12.3 Trigger Start PWM Output Function

(1) Overview

Summary

This function generates a PWM output using a master and a slave channel. It enables the pulse cycle (frequency) and the duty of the TAUDTTOUTm to be set. The pulse cycle is specified using the master channel. The duty is specified using the slave channel. The Trigger Start PWM Output Function is identical to the PWM Output Function except that the master channel of this function can be reset by an effective TAUDTTINm input edge.

Prerequisites

- It requires two channels.
- The operation mode of the master channel must be set to Interval Timer Mode (see **Table 18.132, Contents of the TAUDnCMORm Register for the Master Channel of the Trigger Start PWM Output Function**).
- The operation mode of the slave channel must be set to One-Count Mode (see **Table 18.135, Contents of the TAUDnCMORm Register for Slave Channels of the Trigger Start PWM Output Function**).
- The channel output mode of the slave channel must be set to Synchronous Channel Output Mode 1 (see **Section 18.4.4, Channel Output Modes**).
- TAUDTTOUTm is not used with the master channel of this function.

Functional description

The counters (master and slave) are enabled by setting the channel trigger bits (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm, enabling counting. The current value of TAUDnCDRm is loaded to TAUDnCNTm, and the counter starts to count down from this value. INTTAUDnIm is generated on the master channel, and a PWM output is realized by setting and resetting TAUDTTOUTm (slave).

- Master channel:
  The current value of TAUDnCDRm is loaded to the counter (TAUDnCNTm), INTTAUDnIm is generated and the counter starts to count down from this value.
  When the counter reaches 0000_H and the pulse cycle time has elapsed, INTTAUDnIm is generated and the counters (master and slave) reload the current TAUDnCDRm values.
  If an effective TAUDTTINm input edge is detected, the counter of the master channel reloads the current TAUDnCDRm value, restarts counting down and generates an interrupt.
- Slave channel:
  When the slave detects an interrupt from the master channel, it starts to count down from the current value of TAUDnCDRm. The TAUDTTOUTm signal is set to the active level.
  When the counter reaches 0000_H (duty time has elapsed), INTTAUDnIm is generated and the TAUDTTOUTm signal is reset. The counter returns to $FFFF_H$ and awaits the next INTTAUDnIm of the master channel.

The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1 for the master and slave channel, which in turn sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channel stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

Simultaneous rewrite can be used with this function. See **Section 18.4.3, Simultaneous Rewrite.**

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

Duty cycle [%] = [TAUDnCDRm (slave) / (TAUDnCDRm (master) + 1)] × 100

- Duty cycle = 0%
  TAUDnCDRm (slave) = 0000_H
- Duty cycle = 100%
  TAUDnCDRm (slave) ≥ TAUDnCDRm (master) + 1

(3) Block Diagram and General Timing Diagram

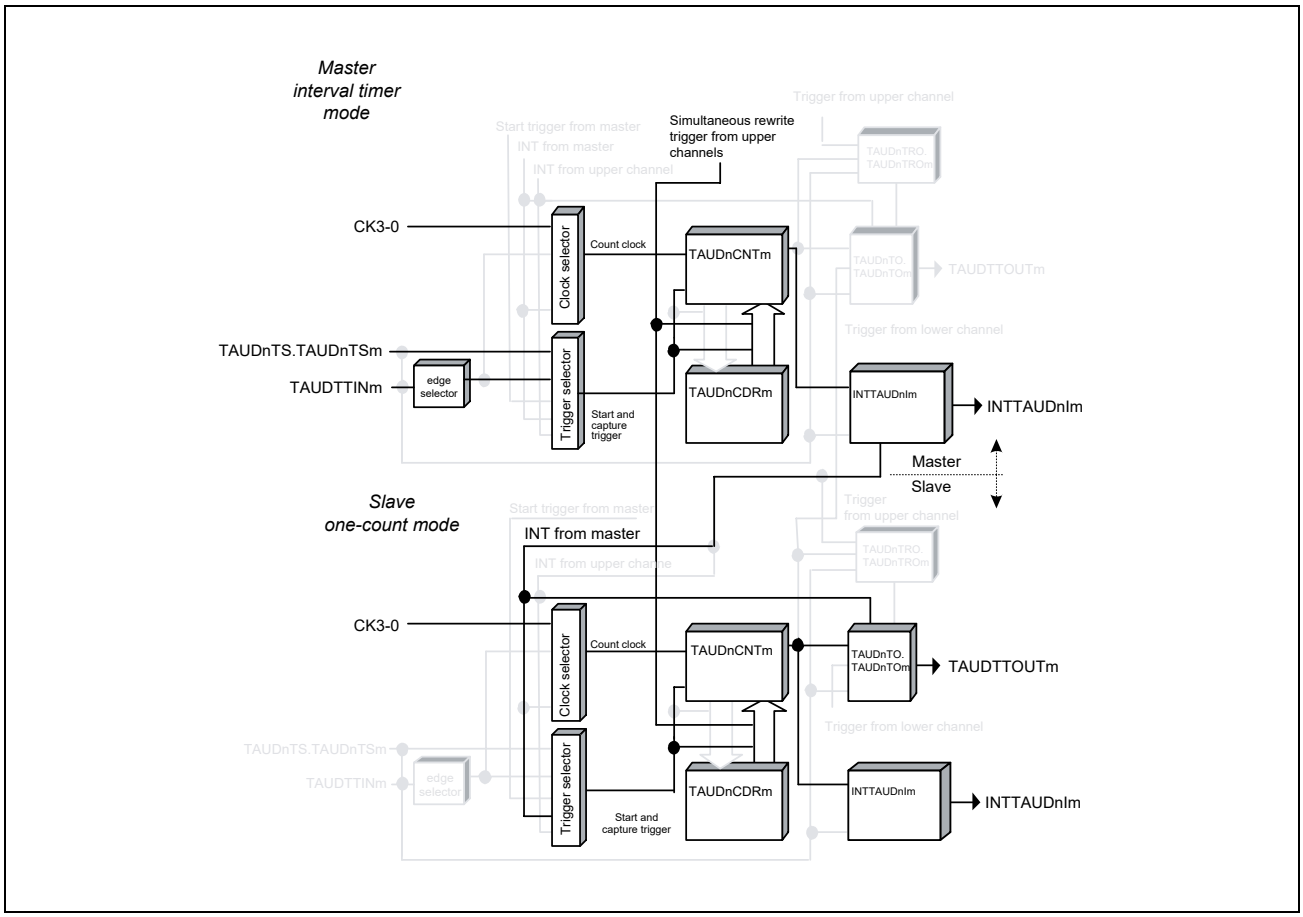


Figure 18.87 Block Diagram for Trigger Start PWM Output Function

The following settings apply to the general timing diagram.

- Detection of rising edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)
- Positive logic (TAUDnTOL.TAUDnTOLm (slave) = 0)

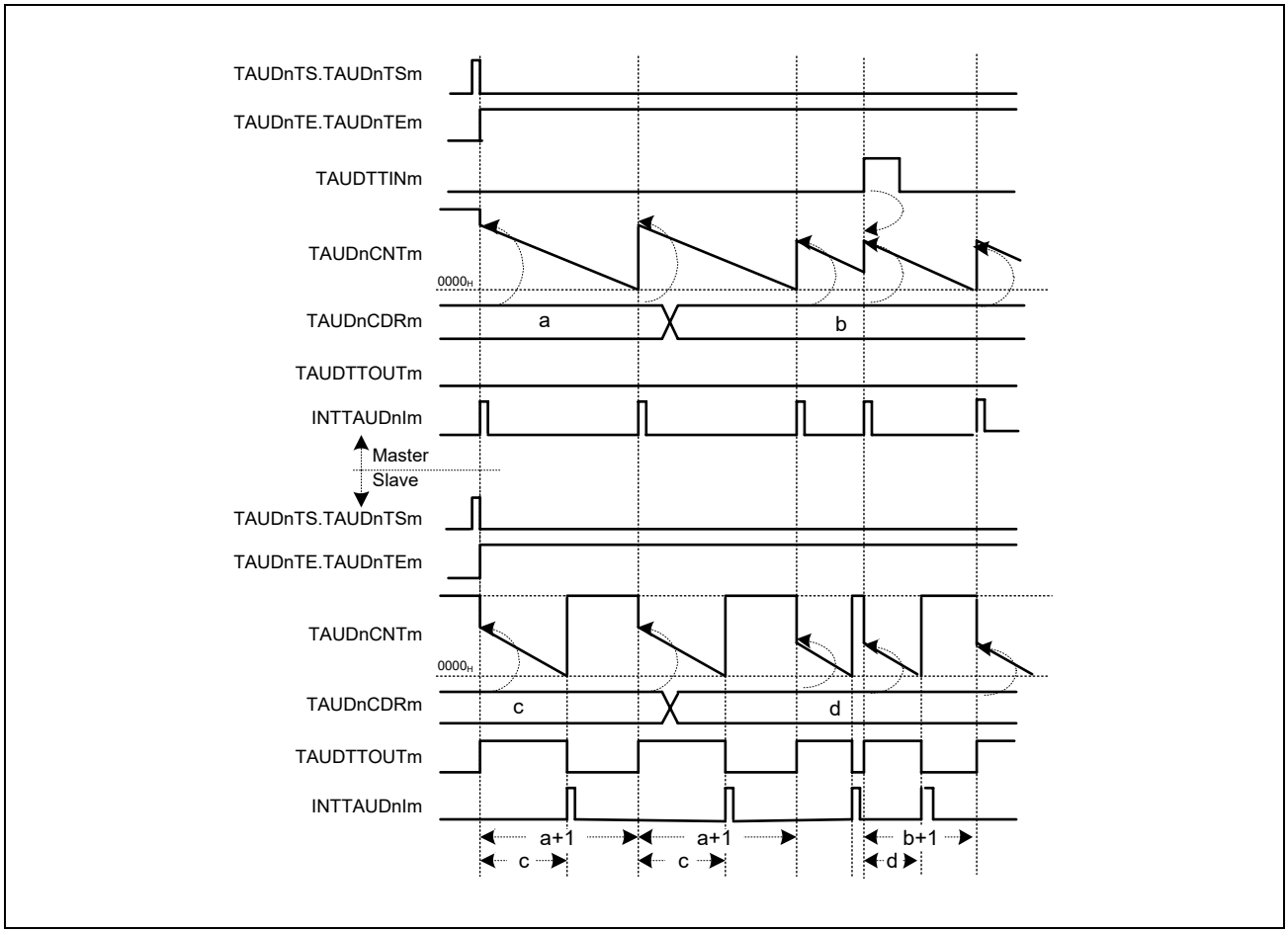


Figure 18.88 General Timing Diagram for Trigger Start PWM Output Function

NOTE

TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.132 Contents of the TAUDnCMORm Register for the Master Channel of the Trigger Start PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bits of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock.
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as the start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.133 Contents of the TAUDnCMURm Register for Master Channels of the Trigger Start PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges 11: Setting prohibited

(c) Channel output mode for master channels

This function does not use channel output mode.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.134 Simultaneous Rewrite Settings for Master Channels of the Trigger Start PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.135 Contents of the TAUDnCMORm Register for Slave Channels of the Trigger Start PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bits of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock.
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Start trigger during operation is valid. The value of the TAUDnMD[0] bit of the master and slave channel must be identical.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.136 Contents of the TAUDnCMURm Register for Slave Channels of the Trigger Start PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels

Table 18.137 Control Bit Settings in Synchronous Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode.
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation.
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation.
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TAUDnTDEm = 0), set these bits to 0.
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output.
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TAUDnTREm = 0), set these bits to 0.
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel.
TAUDnTME.TAUDnTMEm	0: Disables modulation.

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.138 Simultaneous Rewrite Settings for Slave Channels of the Trigger Start PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for the Trigger Start PWM Output Function

Table 18.139 Operating Procedure for the Trigger Start PWM Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set the TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm is generated on the master channel.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCNTm of master channels loads the TAUDnCDRm value and counts down. When the counter reaches 0000_H: • INTTAUDnIm (master) occurs. • The TAUDnCDRm value is reloaded into TAUDnCNTm (master) to continue counting. • The TAUDnCDRm value is reloaded into TAUDnCNTm (slave) to start counting down. • TAUDTTOUTm (slave) is set. When TAUDnCNTm of the slave = 0000_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is set to an inactive level. In addition, the counter of slave channel stops. If a TAUDTTINm input is detected on the master channel while the counter is counting down: • TAUDnCNTm (master and slave) reloads the TAUDnCDRm value and counts down. • INTTAUDnIm (master) occurs. • TAUDTTOUTm (slave) is set to the active level.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Duty cycle = 0%

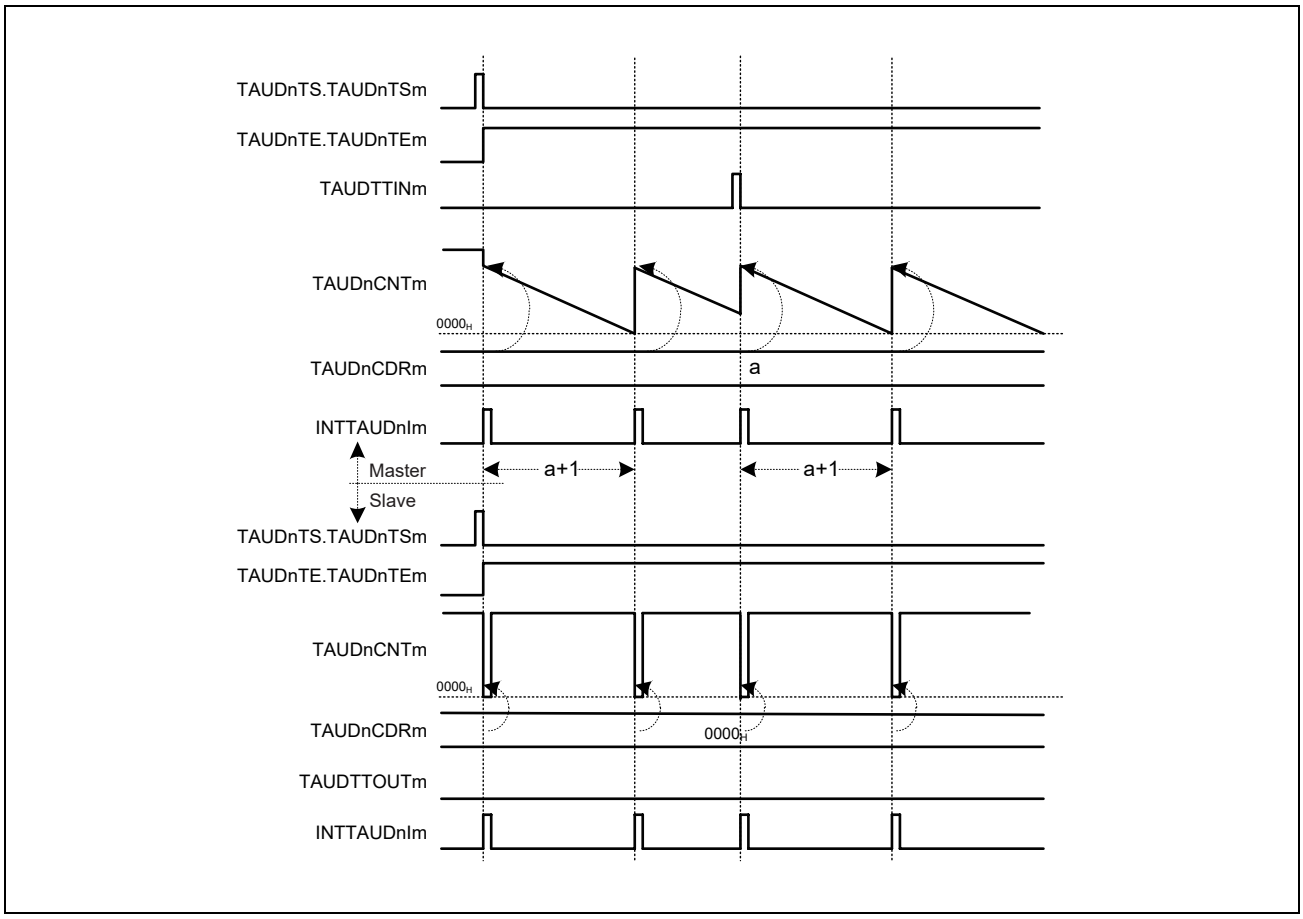


Figure 18.89 TAUDnCDRm (Slave) = 0000_H, Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0) Detection of Falling Edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

- Every time the master channel generates an interrupt (INTTAUDnIm), 0000_H is written to TAUDnCNTm (slave). Therefore, TAUDnCNTm (slave) cannot start to count and TAUDTTOUTm remains inactive.
- TAUDnCNTm (slave) generates an interrupt every time the value of TAUDnCDRm is reloaded. The detection of an effective TAUDTTINm input edge has no effect on TAUDTTOUTm (slave).

(b) Duty cycle = 100%

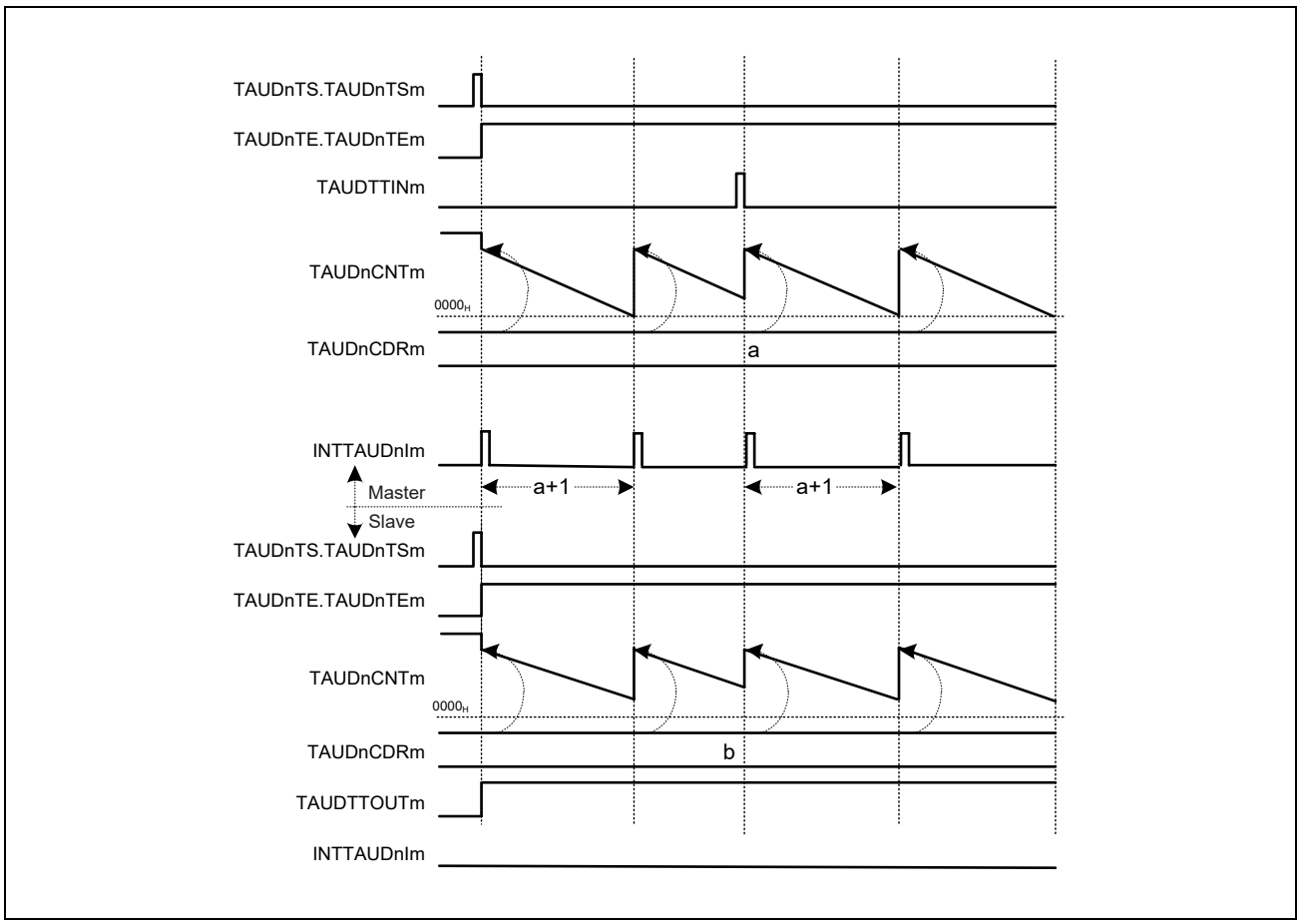


Figure 18.90 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1, Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0) Detection of Falling Edge (TAUDnCMURm.TIS[1:0] = 00_B)

- If the value of TAUDnCDRm (slave) is higher than the value of TAUDnCDRm (master), the counter of the slave channels cannot reach 0000_H and cannot generate interrupts.
  The TAUDTTOUTm remains in the active state.
  The detection of an effective TAUDTTINm input edge has no effect on TAUDTTOUTm (slave).

(c) TAUDTTINm detection and active slave counter

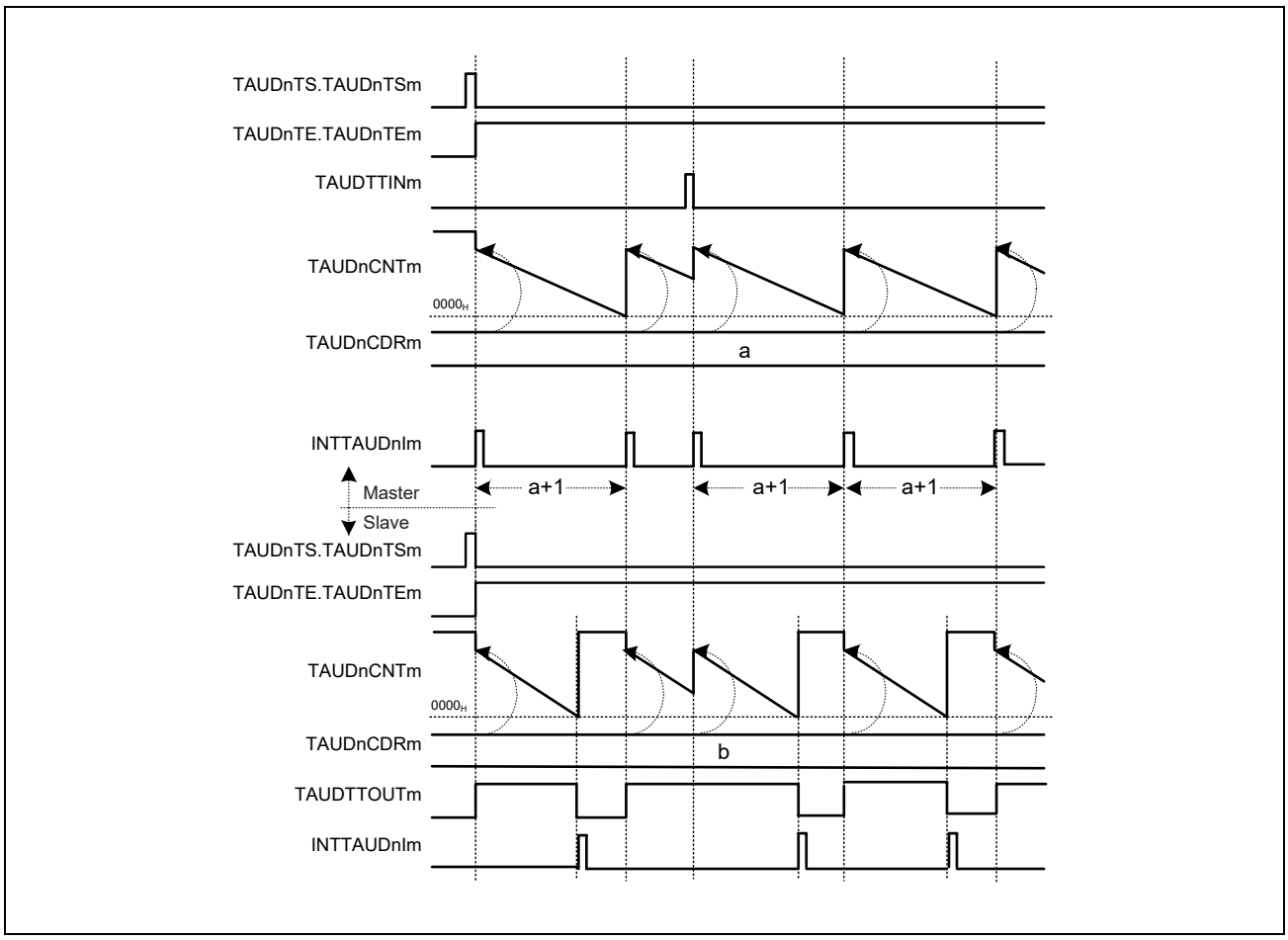


Figure 18.91 Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0), Detection of Falling Edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

- If TAUDnCNTm (slave) reloads the value of TAUDnCDRm (slave) while it is still counting down, TAUDTTOUTm cannot toggle and extends the duty.
  The duty does not correspond to the value of the slave's data register.

18.4.12.4 Delay Pulse Output Function

(1) Overview

Summary

This function outputs two signals. The pulse width and pulse cycle of the reference signal are defined using the master channel and slave channel 1. Slave channels 2 and 3 output the reference signal with a specified delay. The delay signal is identical to the reference signal, but delayed by the amount specified on slave channel 2.

The signal values are specified in the following way:

- The pulse cycle is specified using the master channel.
- The duty cycle of the reference signal is specified using slave channel 1. The duty cycle of the delay signal is specified using slave channel 3.
- The delay is specified on slave channel 2.

Prerequisites

- It requires four channels.
- The operating mode for the master channel should be set to interval timer mode (see **Table 18.140, Contents of TAUDnCMORm Register for Master Channels of Delay Pulse Output Function**).
- The operating mode for slave channels 1 and 2 should be set to one-count mode (see **Table 18.143, Contents of TAUDnCMORm Register for Slave Channel 1 of Delay Pulse Output Function**).
- The operating mode for slave channel 3 should be set to pulse one-count mode (see **Table 18.147, Contents of TAUDnCMORm Register for Slave Channel 2 of Delay Pulse Output Function**).
- TAUDTTOUTm is not used with the master channel and slave channel 2.
- The channel output mode for slave channel 1 should be set to synchronous channel output mode 1 (see **18.4.4, Channel Output Modes**).
- The channel output mode for slave channel 3 should be set to independent channel output mode 2 (see **18.4.4, Channel Output Modes**).

Functional description

The counters of the channel group are started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm to 1, enabling count operation.

- Master channel:
  The current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from this value. INTTAUDnIm is generated on the master channel.
  When the counter value of master channel reaches 0000_H and pulse cycle time has elapsed, INTTAUDnIm is generated. The TAUDnCDRm value is reloaded into the counter to perform count down.
- Slave channels 1 and 2:
  Slave channels 1 and 2 start to count down from the current TAUDnCDRm value when detecting an interrupt from the master channel. TAUDTTOUTm signal (slave 1) is set.
  - Slave channel 1:
    When the counter of slave channel 1 reaches 0000_H (duty time has elapsed), INTTAUDnlm is generated and TAUDTTOUTm signal is reset. The counter is reset to $FFFF_H$ and waits for the next INTTAUDnlm of master channel.
  - Slave channel 2:
    When the counter of slave channel 2 reaches 0000_H and delay time has elapsed, INTTAUDnIm is generated. The counter is reset to $FFFF_H$ and waits for the next INTTAUDnlm of master channel.
    Generating INTTAUDnIm (slave channel 2) triggers the counter of slave channel 3.
- Slave channel 3:
  When slave channel 3 detects an interrupt from slave channel 2, its counter starts counting down from the current value of TAUDnCDRm. INTTAUDnIm is generated and the TAUDTTOUTm signal (slave channel 3) is set.
  When the counter of slave channel 3 reaches 0001_H, INTTAUDnIn is generated and the TAUDTTOUTm signal is reset.
  The delayed PWM pulse is output from slave channel 3.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but their values are retained. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

Simultaneous rewrite can be used with this function. See **Section 18.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

Duty width 1 = (TAUDnCDRm (slave 1)) × count clock cycle

Delay width = (TAUDnCDRm (slave 2) + 1) × count clock cycle

Duty width 2 = (TAUDnCDRm (slave 3)) × count clock cycle

However, the delay width shall be set within the following range:

$0000_H \leq$ TAUDnCDRm (slave 2) < TAUDnCDRm (master)

NOTES

1. The waveform of TAUDTTOUTm (slave 3) becomes the waveform made by delaying the waveform of TAUDTTOUTm (slave 1) by the quantity generated by slave 2. It is impossible to make a delay longer than the pulse cycle.
2. If INTTAUDnIm of slave 2 is generated while slave 3 is counting, slave 3 restarts operation. Therefore, the waveform of TAUDTTOUTm (slave 3) is retained on the active level. In this case, TAUDTTOUTm (Slave-CH-3) cannot output the waveform generated by delaying the basic pulse of TAUDTTOUTm (Slave-CH-1).

(3) Block Diagram and General Timing Diagram

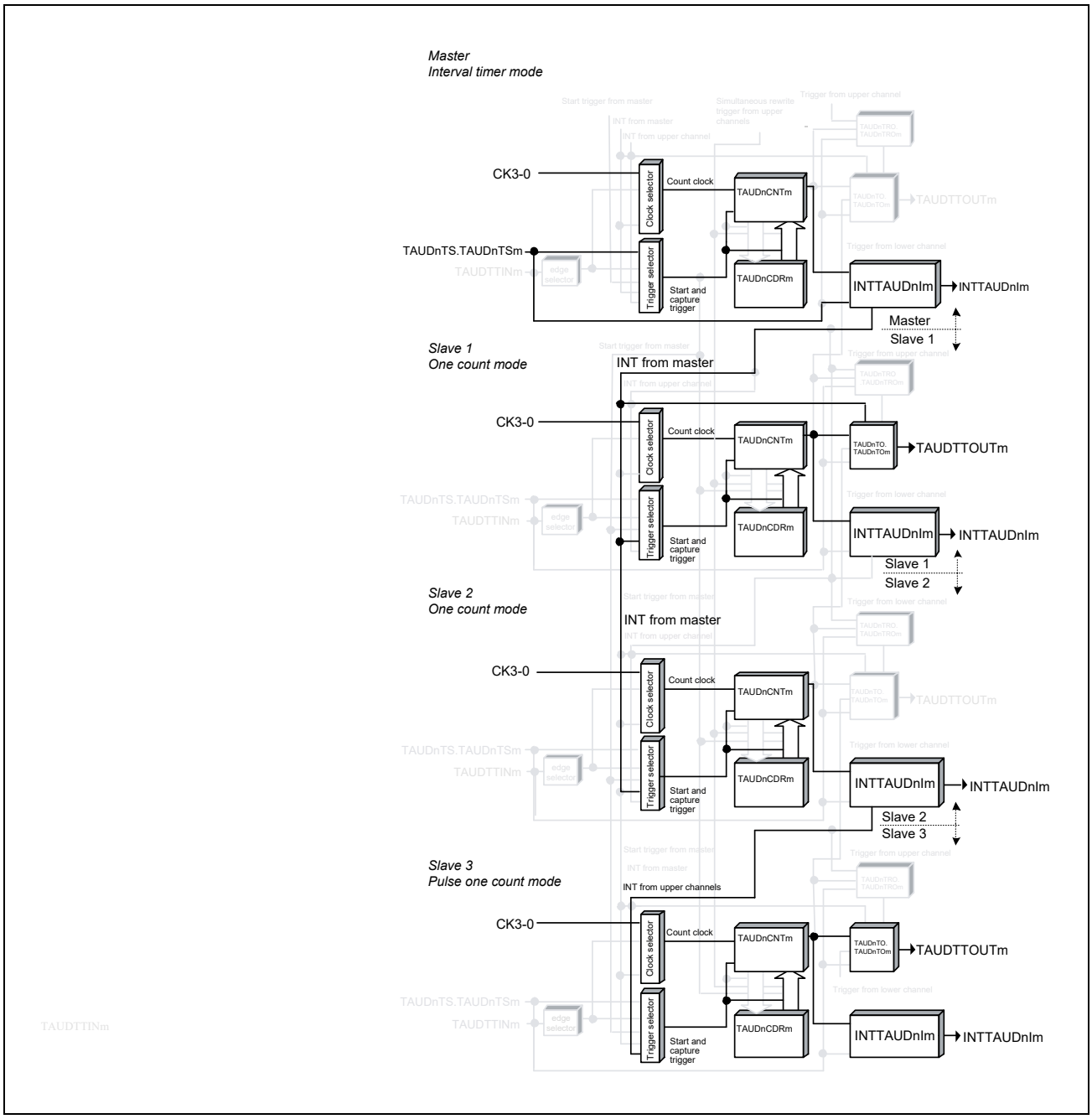


Figure 18.92 Block Diagram of Delay Pulse Output Function

The following settings apply to the general timing diagram.

- Slave channel 1: Positive logic (TAUDnTOL.TAUDnTOLm = 0)
- Slave channel 3: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

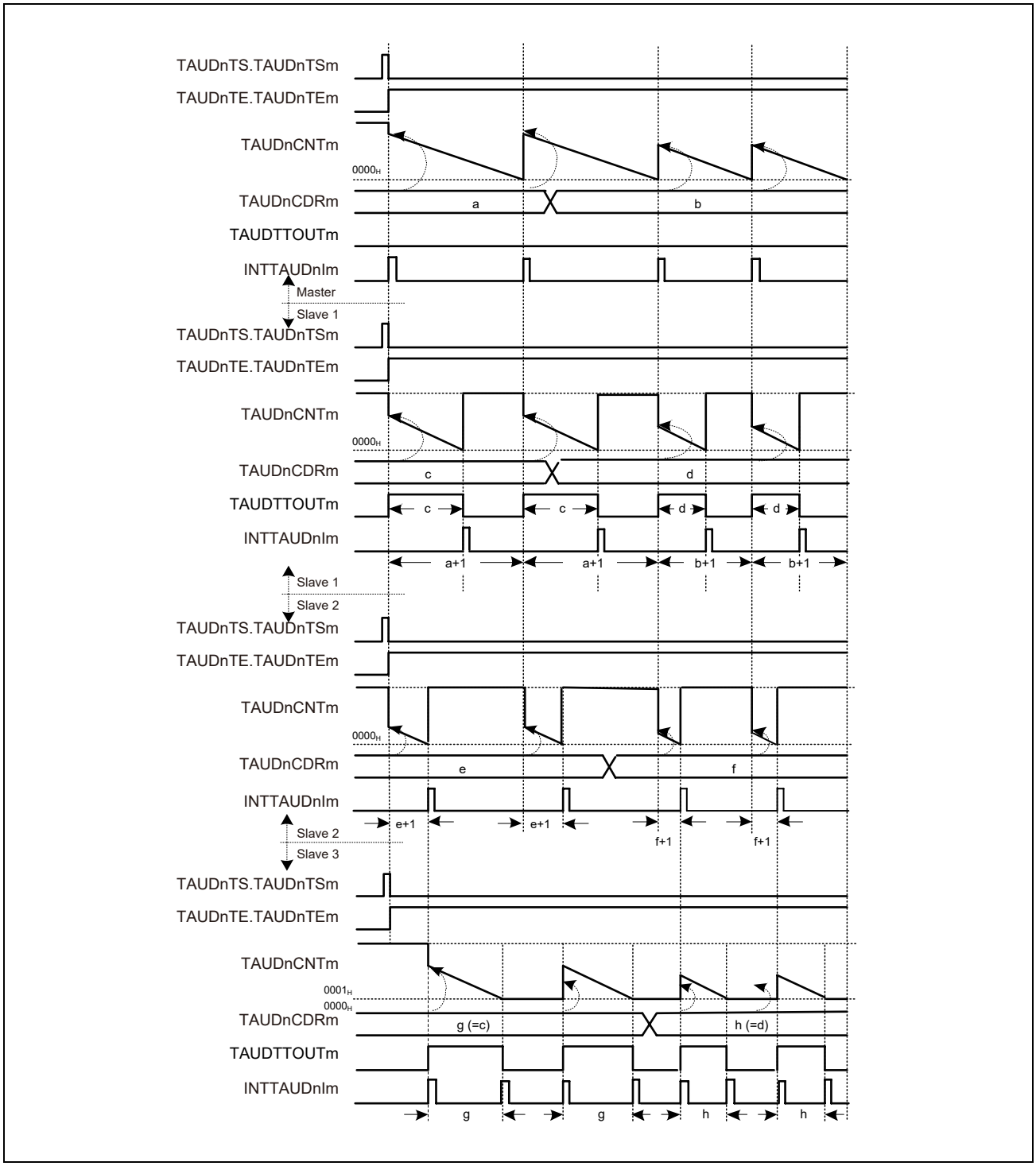


Figure 18.93 General Timing Diagram of Delay Pulse Output Function

NOTE

TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.140 Contents of TAUDnCMORm Register for Master Channels of Delay Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.141 Contents of TAUDnCMURm Register for Master Channels of Delay Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used for master channels with this function.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.142 Simultaneous Rewrite Settings for Master Channels of Delay Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(5) Register Settings for Slave Channel 1

(a) TAUDnCMORm for slave channel 1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.143 Contents of TAUDnCMORm Register for Slave Channel 1 of Delay Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Valid start trigger during operation

(b) TAUDnCMURm for slave channel 1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.144 Contents of TAUDnCMURm Register for Slave Channel 1 of Delay Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 1

Table 18.145 Control Bit Settings for Slave Channel 1 in Synchronous Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channel 1

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.146 Simultaneous Rewrite Settings for Slave Channel 1 of Delay Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Register Settings for Slave Channel 2

(a) TAUDnCMORm for slave channel 2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.147 Contents of TAUDnCMORm Register for Slave Channel 2 of Delay Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS [2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD [4:1]	0100: One-count mode
0	TAUDnMD0	1: Valid start trigger during operation

(b) TAUDnCMURm for slave channel 2

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.148 Contents of TAUDnCMURm Register for Slave Channel 2 of Delay Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 2

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used with this function.

(d) Simultaneous rewrite for slave channel 2

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.149 Simultaneous Rewrite Settings for Slave Channel 2 of Delay Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(7) Register Settings for Slave Channel 3

(a) TAUDnCMORm for slave channel 3

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.150 Contents of TAUDnCMORm Register for Slave Channel 3 of Delay Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	101: INTTAUDnIm of upper channel (m – 1) is a start trigger regardless of master setting.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1010: Pulse one-count mode
0	TAUDnMD0	1: Valid start trigger during operation

(b) TAUDnCMURm for slave channel 3

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.151 Contents of TAUDnCMURm Register for Slave Channel 3 of Delay Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 3

Table 18.152 Control Bit Settings in Independent Channel Output Mode 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channel 3

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.153 Simultaneous Rewrite Settings for Slave channel 3 of Delay Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(8) Operating Procedure for Delay Pulse Output Function

Table 18.154 Operating Procedure for Delay Pulse Output Function (1/2)

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 1: Set TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (5), Register Settings for Slave Channel 1**. Slave channel 2: Set TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (6), Register Settings for Slave Channel 2**. Slave channel 3: set the TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (7), Register Settings for Slave Channel 3**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.

Table 18.154 Operating Procedure for Delay Pulse Output Function (2/2)

	Operation	TAUDn Status
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master channel and slave channels 1 and 2 start. INTTAUDnIm is generated on the master channel and TAUDTTOUTm (slave channel 1) is set.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCNTm of master channel and slave channels 1 and 2 load TAUDnCDRm value and count down. When the counter of master channel reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCDRm value is reloaded into TAUDnCNTm (master) to continue count operation. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave 1/2) to start countdown. • TAUDTTOUTm (slave 1) is set. When TAUDnCNTm (slave 1) reaches 0000_H: • INTTAUDnIm (slave 1) occurs. • TAUDTTOUTm (slave 1) is reset. When TAUDnCNTm (slave 2) reaches 0000_H: • INTTAUDnIm (slave 2) occurs. • INTTAUDnIm (slave 3) occurs. • TAUDTTOUTm (slave 3) is set. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave 3) to start a countdown operation. When TAUDnCNTm (slave 3) reaches 0000_H: • INTTAUDnIm (slave 3) occurs. • TAUDTTOUTm (slave 3) is reset.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(9) Specific Timing Diagrams

(a) Duty cycle (slave 3) = 100%

The following values apply to **Figure 18.94:**

- TAUDnCDRm (master) = $000A_H$
- TAUDnCDRm (slave 1) = $000B_H$
- TAUDnCDRm (slave 2) = 0000_H
- TAUDnCDRm (slave 3) = $000B_H$

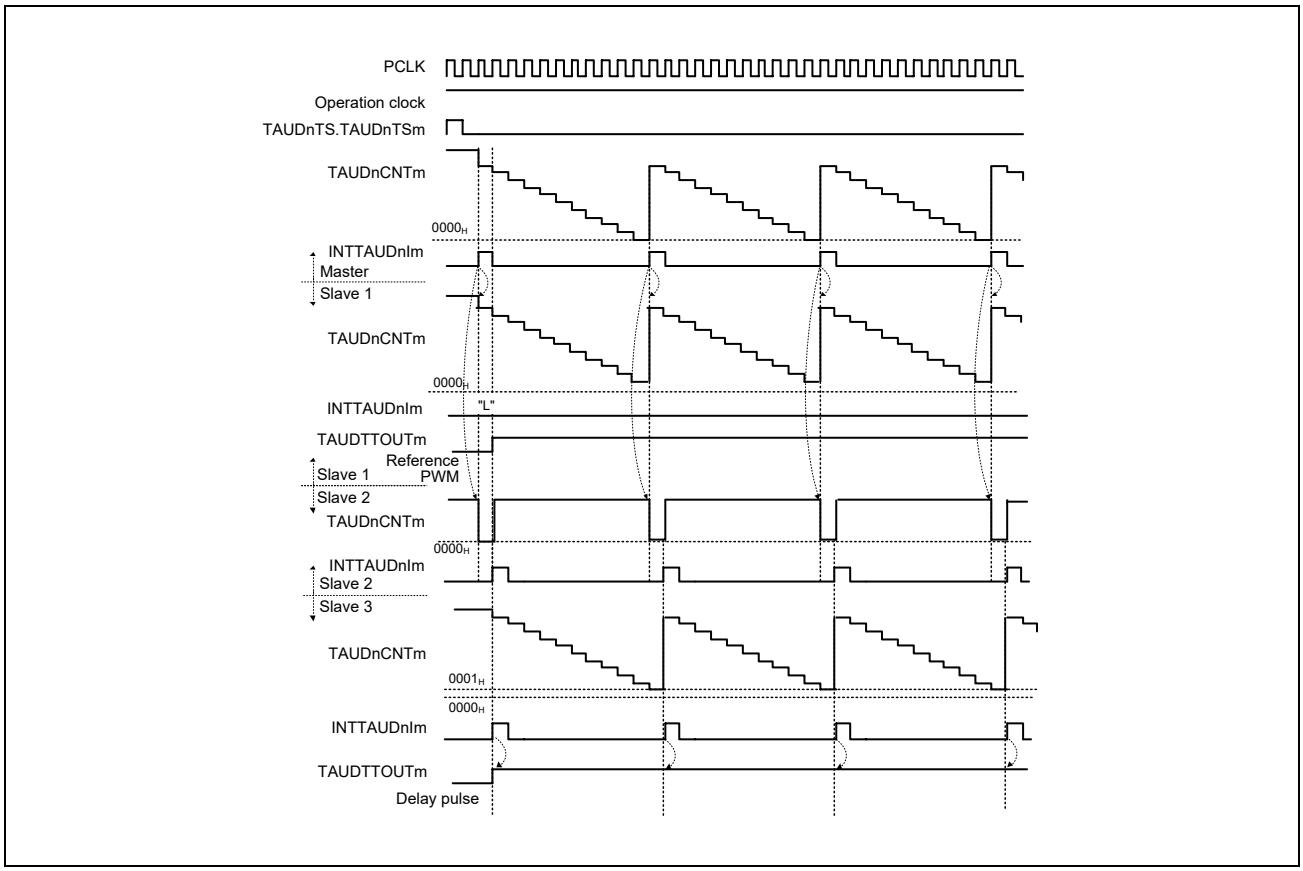


Figure 18.94 Duty Cycle (slave 3) = 100%

- If the value of TAUDnCDRm (slaves 1 and 3) is higher than the value of TAUDnCDRm (master), the counter of slave channel 1 cannot reach 0000_H and cannot generate interrupts. TAUDTTOUTm of channels 1 and 3 remain in the active state.

(b) TAUDTTOUTm (slave 1) = TAUDTTOUTm (slave 3)

The following values apply to **Figure 18.95**.

- TAUDnCDRm (master) = $000A_H$
- TAUDnCDRm (slave 1) = 0005_H
- TAUDnCDRm (slave 2) = 0000_H
- TAUDnCDRm (slave 3) = 0005_H

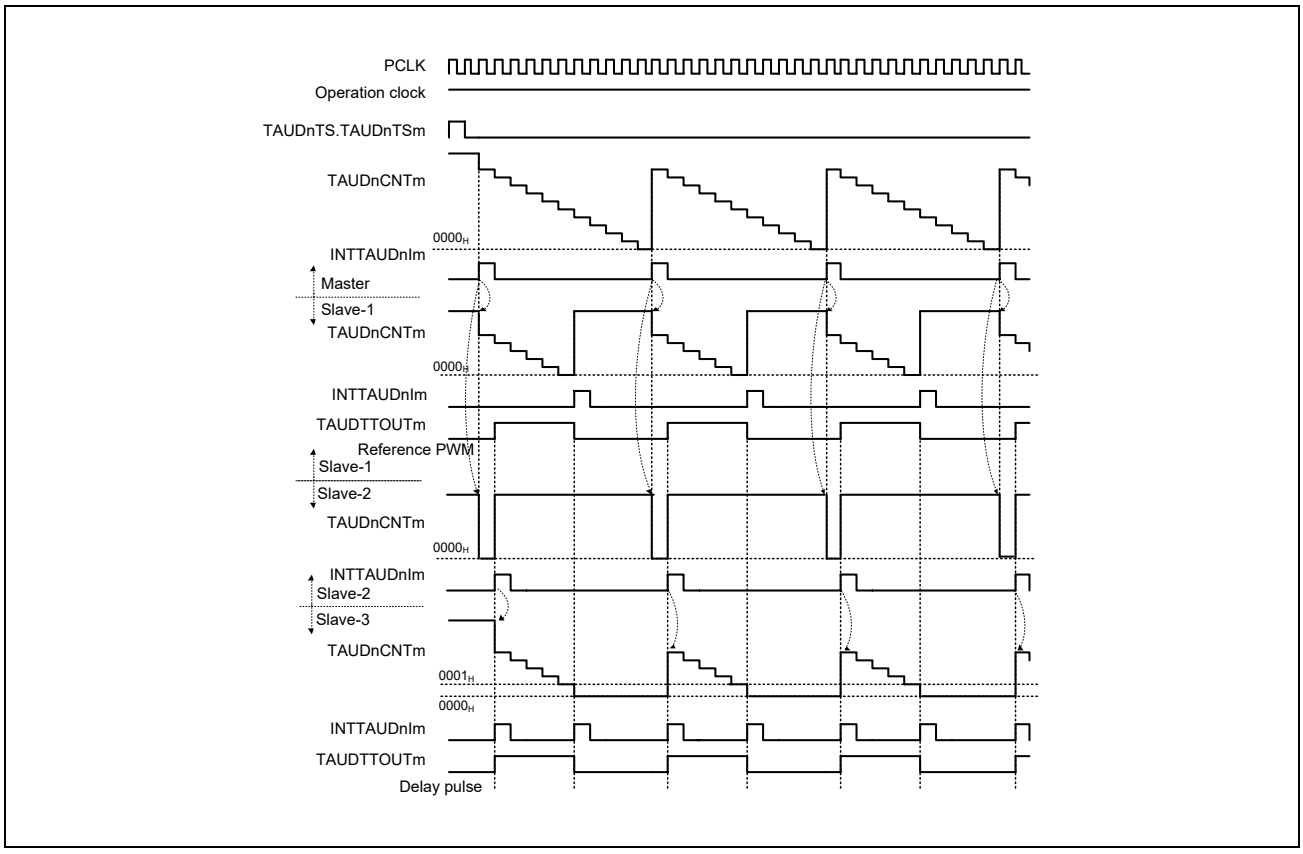


Figure 18.95 TAUDTTOUTm (Slave 1) = TAUDTTOUTm (Slave 3)

- If TAUDnCDRm (slave 2) = 0000_H, the counter of slave channel 3 starts counting one count clock later than the counter of slave channel 1. The reference pulse and the delay pulse are output with a delay of one clock count.

18.4.12.5 Offset Trigger Output Function

(1) Overview

Summary

This function generates a PWM output using a master channel and a slave channel, enabling the pulse width (duration) of the TAUDTTOUTm to be set. The pulse cycle is set by detecting an effective input edge of master channel. The pulse width is specified on the slave channel.

Prerequisites

- It requires two channels.
- The operating mode for the master channel should be set to capture mode (see **Table 18.155, Contents of TAUDnCMORm Register for Master Channels of Offset Trigger Output Function**).
- The operating mode for slave channels should be set to one-count mode (see **Table 18.158, Contents of TAUDnCMORm Register for Slave Channels of Offset Trigger Output Function**).
- The output mode for slave channels should be set to synchronous channel output mode 1 (see **Section 18.4.4, Channel Output Modes**).
- TAUDTTOUTm is not used with the master channel of this function.

Functional description

The counter can be started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This makes TAUDnTE.TAUDnTEm = 1, enabling the counter to count up. The master channel counter (TAUDnCNTm) starts to count up from 0000_H.

- Master channel:
  When an effective TAUDTTINm input edge is detected, the current value of the counter (TAUDnCNTm) is loaded into the data register of master channel (TAUDnCDRm). INTTAUDnIm is generated and the counter restarts to count up from 0000_H.
- Slave channels:
  The INTTAUDnIm of master channel sets the TAUDTTOUTm (slave) signal and triggers the counter of the slave channel. The current value of TAUDnCDRm (slave) is loaded into TAUDnCNTm (slave) and the counter starts to count down from this value.
  When the counter reaches 0000_H (duty time has elapsed), INTTAUDnIm is generated and TAUDTTOUTm signal is reset. The counter returns to $FFFF_H$ and awaits the next INTTAUDnIm of master channel.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

(2) Equations

Pulse width = (TAUDnCDRm (slave) + 1) × count clock cycle

Duty cycle [%] = [TAUDnCDRm (slave) / TAUDnCDRm (master) + 1)] × 100

- Duty cycle = 0%
  TAUDnCDRm (slave) = 0000_H
- Duty cycle = 100%
  TAUDnCDRm (slave) ≥ TAUDnCDRm (master) + 1

(3) Block Diagram and General Timing Diagram

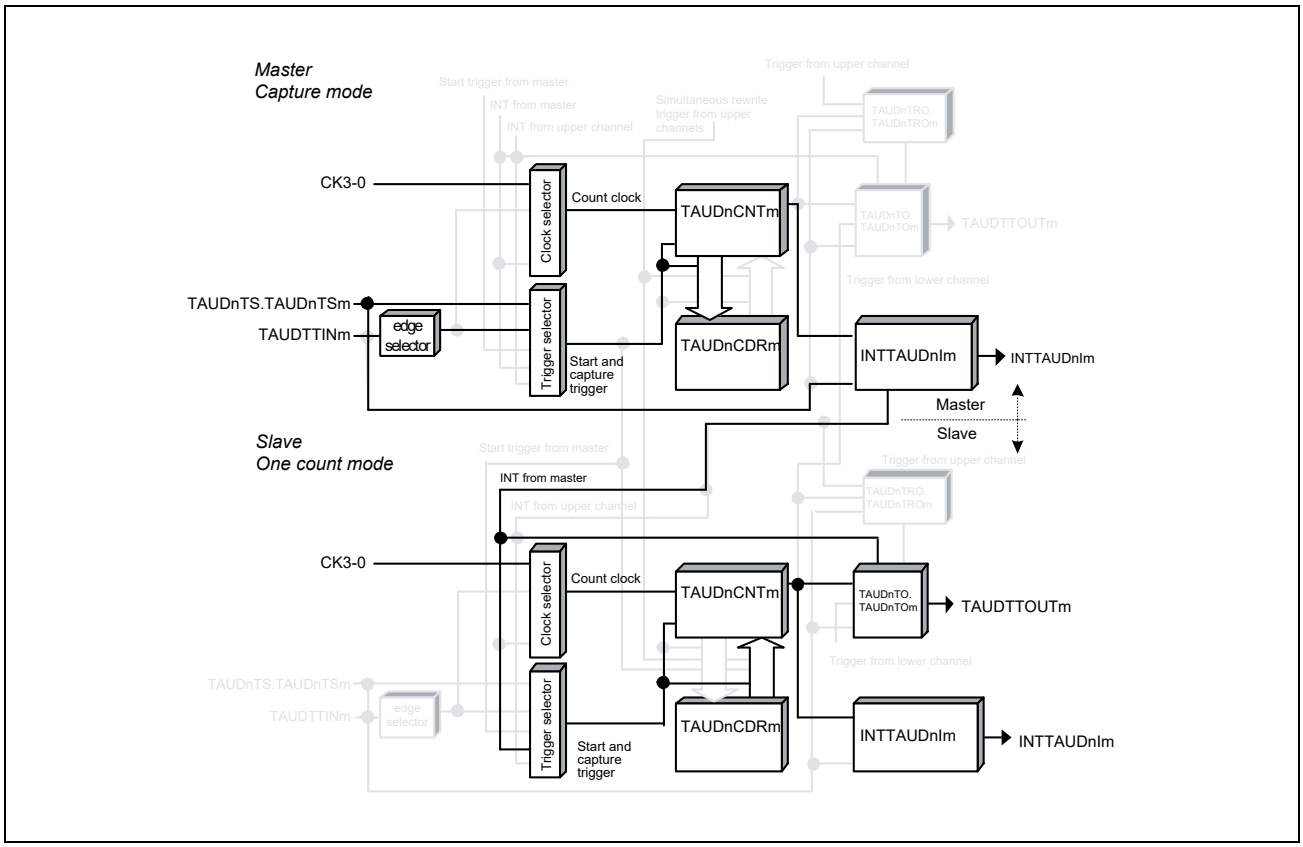


Figure 18.96 Block Diagram of Offset Trigger Output Function

The following settings apply to the general timing diagram.

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

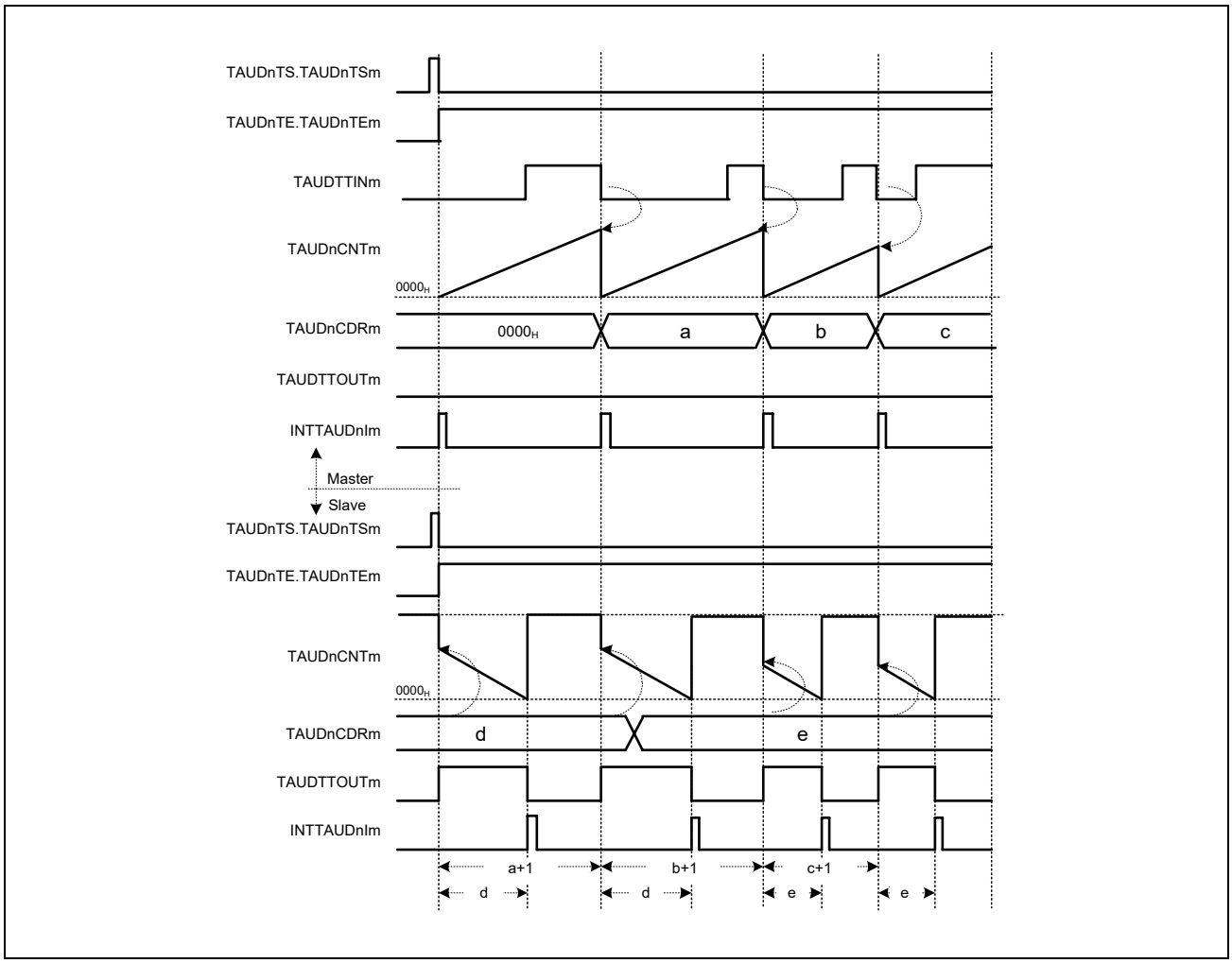


Figure 18.97 General Timing Diagram of Offset Trigger Output Function

NOTE

TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.155 Contents of TAUDnCMORm Register for Master Channels of Offset Trigger Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as the start trigger
7, 6	TAUDnCOS[1:0]	11: Capture register is updated upon detection of an effective TAUDTTINm input edge or when a counter overflow occurs: • Detection of an effective TAUDTTINm input edge: The counter value is written into TAUDnCDRm. • Occurrence of overflow: $FFFF_H$ is written into TAUDnCDRm. An effective TAUDTTINm input edge to be detected next is ignored. TAUDnCSRm.TAUDnOVF is set when a counter overflow occurs, and cleared by setting TAUDnCSCm.TAUDnCLOV = 1.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0010: Capture mode
0	TAUDnMD0	1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.156 Contents of TAUDnCMURm Register for Master Channels of Offset Trigger Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used with this function.

(d) Simultaneous rewrite for master channels

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.157 Simultaneous Rewrite Settings for Master Channels of Offset Trigger Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.158 Contents of TAUDnCMORm Register for Slave Channels of Offset Trigger Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Enables start trigger detection while counting.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.159 Contents of TAUDnCMURm Register for Slave Channels of Offset Trigger Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels

Table 18.160 Control Bit Settings for Slave Channel 1 in Synchronous Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channels

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.161 Simultaneous Rewrite Settings for Slave Channels of Offset Trigger Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(6) Operating Procedure for Offset Trigger Output Function

Table 18.162 Operating Procedure for Offset Trigger Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/TAUDnCMURm register and channel output mode as described in **Section (5), Register Settings for Slave Channels**. The TAUDnCDRm register of master channel functions as a capture register. Set the value of TAUDnCDRm register of slave channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start: • TAUDnCNTm (master) counts up. • TAUDnCDRm value is loaded into TAUDnCNTm (slave) to perform count down. INTTAUDnIm is generated on the master channel and TAUDTTOUTm (slave) is set.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCSCm.TAUDnCLOV can be set to 1. TAUDnCDRm of slave channel can be changed after the generation of INTTAUDnIm (master). TAUDnCNT.TAUDnCNTm and TAUDnCSRm can be read at any time.	When TAUDnCNTm of the slave = 0000_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is reset and the counter of slave channel stops. When TAUDTTINm input edge is detected on the master channel: • INTTAUDnIm (master) occurs. • TAUDnCNTm (master) is reset to 0000_H and then continues count operation subsequently. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave) to perform count down. • TAUDTTOUTm (slave) is set.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Duty cycle = 0%

The following settings apply to this diagram:

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

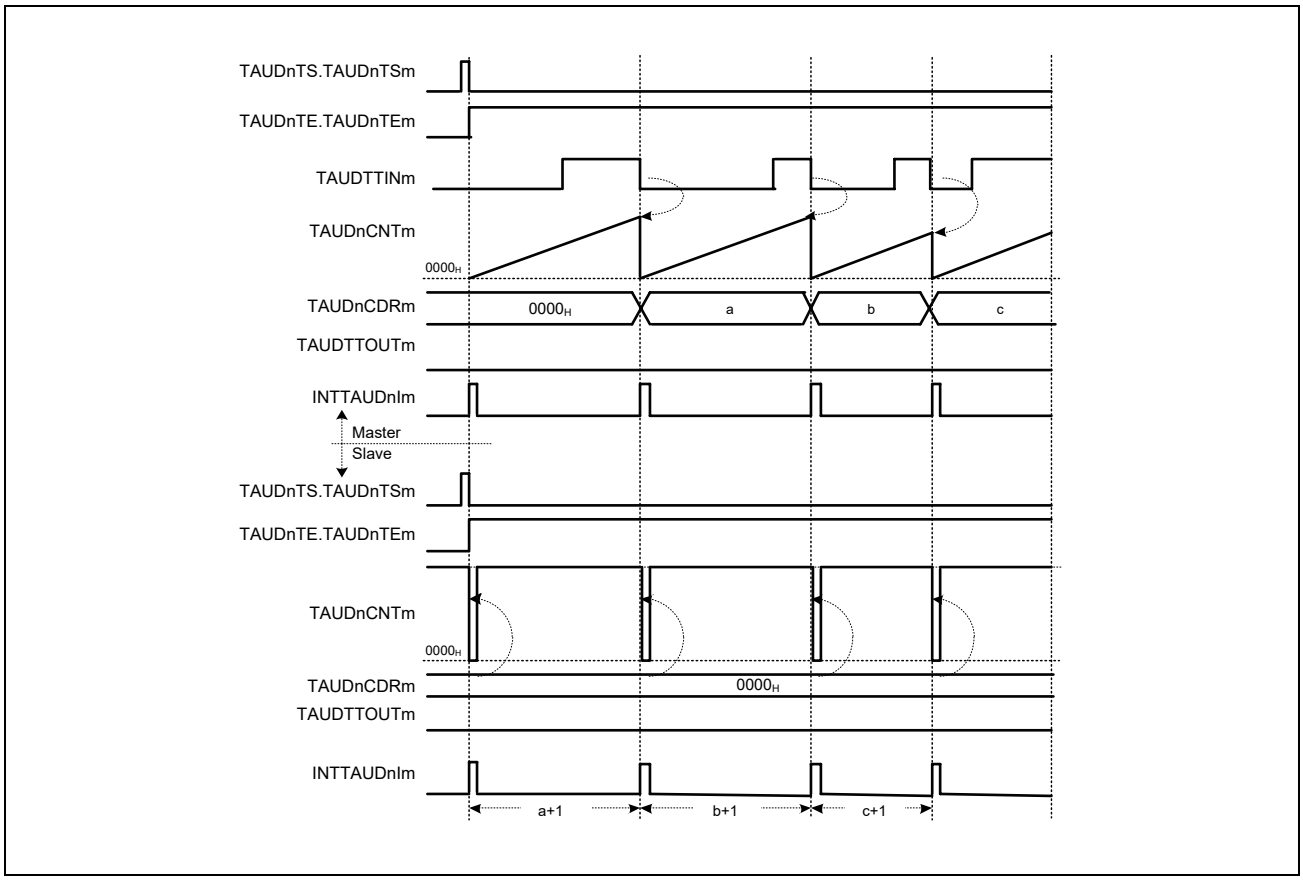


Figure 18.98 TAUDnCDRm (Slave) = 0000_H

- When TAUDnCDRm (slave) = 0000_H, 0000_H is written to TAUDnCNTm every time the master channel generates an interrupt (INTTAUDnIm), and TAUDnCNTm cannot start to count. The TAUDTTOUTm remains at not active state.
- TAUDnCNTm (slave) generates an interrupt every time the value of TAUDnCDRm is reloaded. The slave and the master channel generate interrupts in the same cycle.

(b) Duty cycle = 100%

The following settings apply to this diagram:

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

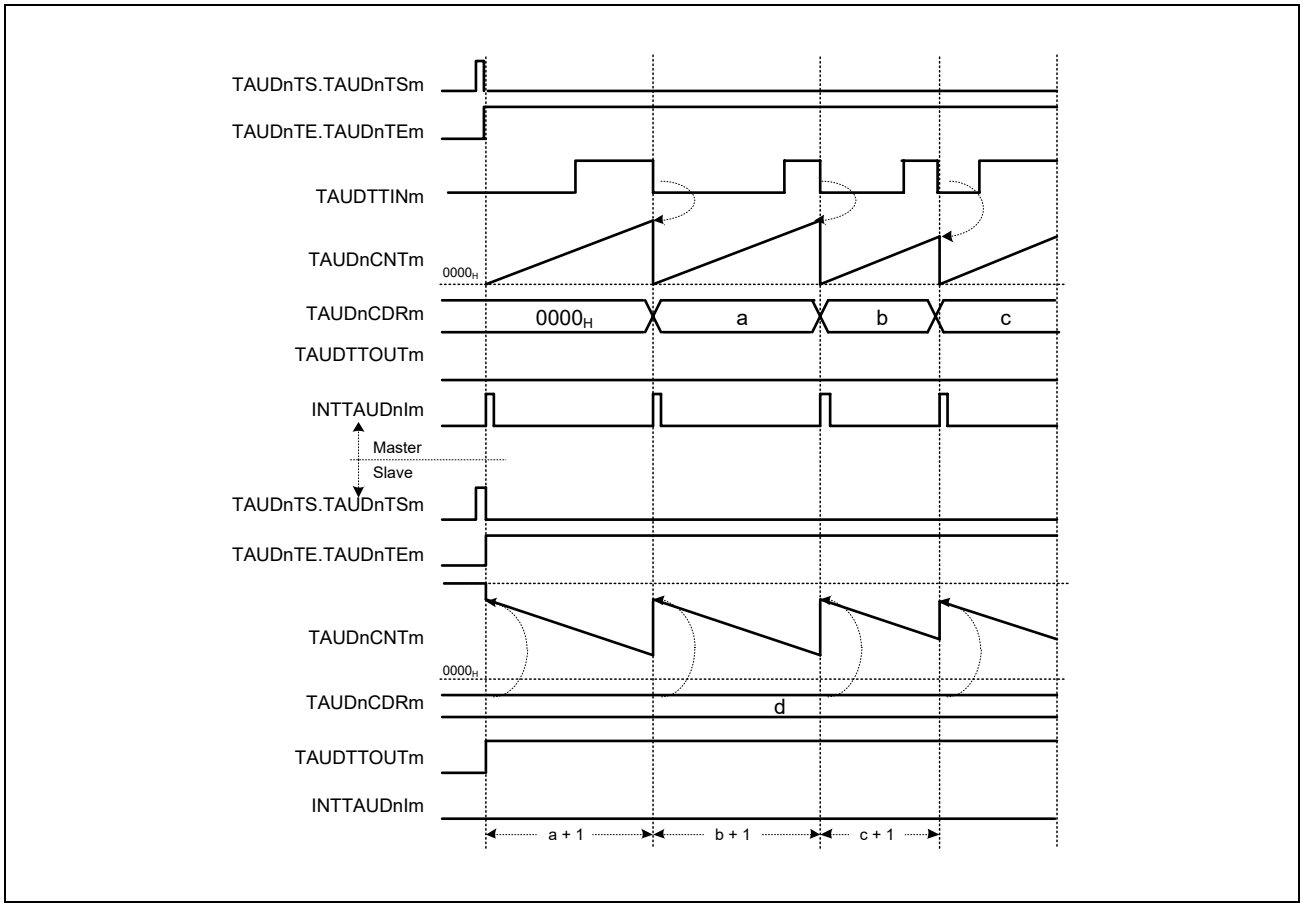


Figure 18.99 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1

- If the value TAUDnCDRm (slave) is higher than the interval of effective input edges, the counter of the slave channel cannot reach 0000_H and cannot generate interrupts. The TAUDTTOUTm remains at active state.

18.4.12.6 A/D Conversion Trigger Output Function Type 1

(1) Overview

Summary

This function is identical to **Section 18.4.12.1, PWM Output Function**, except that TAUDTTOUTm is not output.

This is achieved by setting the channel output mode for the slave to independent channel output mode controlled by software.

(2) Block Diagram and General Timing Diagram

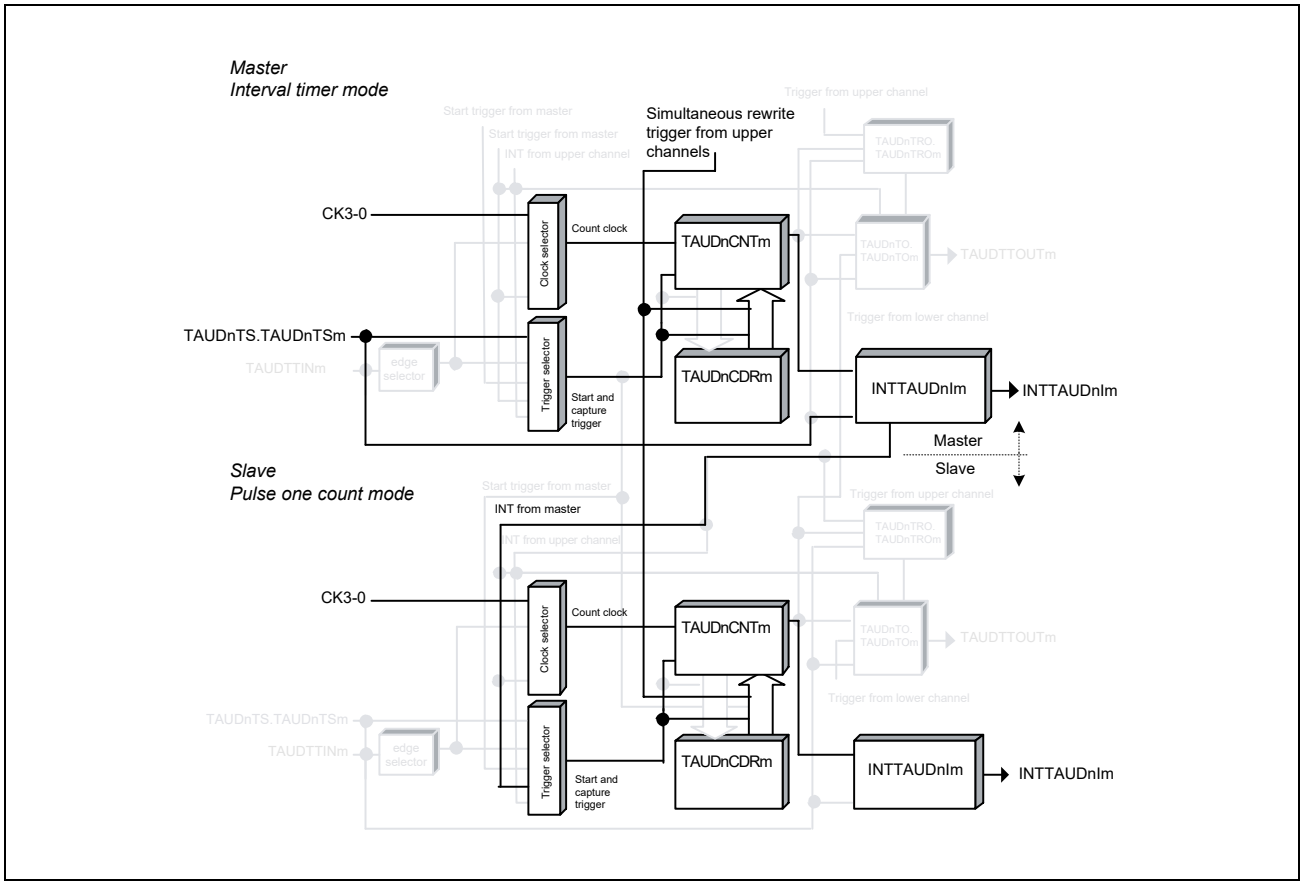


Figure 18.100 Block Diagram of A/D Conversion Trigger Output Function Type 1

The following settings apply to the general timing diagram.

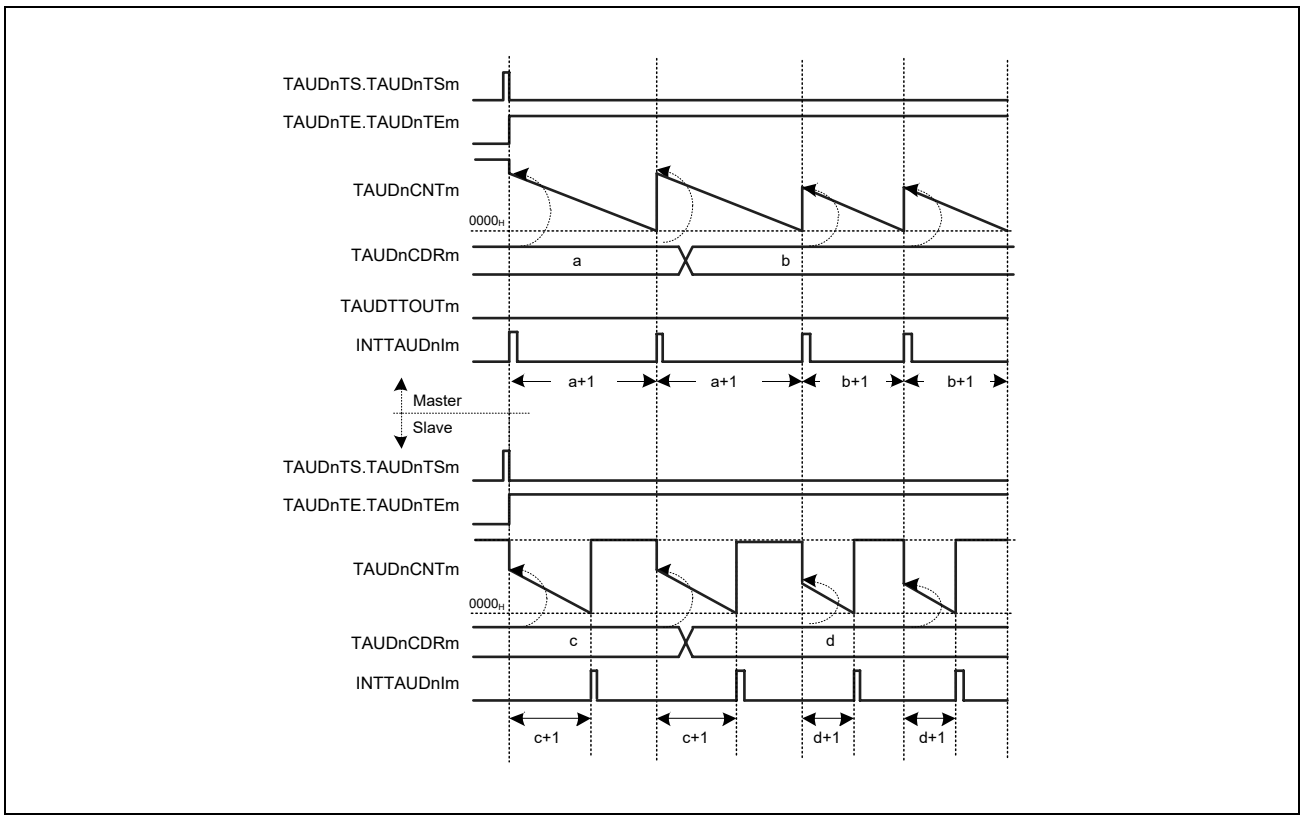


Figure 18.101 General Timing Diagram of A/D Conversion Trigger Output Function Type 1

18.4.12.7 Triangle PWM Output Function

(1) Overview

Summary

This function generates multiple triangle PWM outputs by using a master and one or more slave channels. It enables the pulse cycle (frequency) and the duty cycle of TAUDTTOUTm to be set using the master and slave channels respectively.

The master channel generates a carrier cycle. The first cycle of master channel controls the down status and the second cycle controls the up status of the slave counter.

Prerequisites

- It requires two channels.
- The operating mode for master channels should be set to interval timer mode (see **Table 18.163, Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function**).
- The operating mode for slave channels should be set up/down count mode (see **Table 18.167, Contents of TAUDnCMORm Register for Slave Channels of Triangle PWM Output Function**).
- The channel output mode for master channels should be set to independent channel output mode 1 (see **Section 18.4.4, Channel Output Modes**).
- The channel output mode for slave channels should be set to synchronous channel output mode 2 (see **Section 18.4.4, Channel Output Modes**).
- The following settings allows TAUDTTOUTm signal to be at high level during the down status of a carrier cycle.
  - If TAUDnCMORm.TAUDnMD0 (master) bit is set to 0, TAUDnTO.TAUDnTOm should be set to 1 while TAUDnTOE.TAUDnTOEm is set to 0 (recommended setting).
  - If TAUDnCMORm.TAUDnMD0 (master) bit is set to 1, TAUDnTO.TAUDnTOm should be set to 0 while TAUDnTOE.TAUDnTOEm is set to 0.

Functional description

The counters are started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1 for every channel. This in turn sets TAUDnTE.TAUDnTEm, enabling count operation.

The current values of TAUDnCDRm (master and slave) are loaded into TAUDnCNTm (master and slave) and the counters start counting down from these values. When the TAUDnCMORm.TAUDnMD0 bit of master channel is set to 1, an interrupt is generated and TAUDTTOUTm signal of master toggles.

- Master channel:
  When the counter of master channel reaches 0000_H (pulse cycle time has elapsed), INTTAUDnIm is generated and the TAUDTTOUTm signal toggles. TAUDnCNTm then reloads the TAUDnCDRm value and counts down.
- Slave channels:
  The INTTAUDnIm of master channel triggers the counter of the slave channel:
  - If the slave counter is counting down, the count direction changes.
  - If the slave counter is counting up, TAUDnCDRm value is reloaded and the counter starts to count down.

  When the counter of the slave channel reaches 0001_H while counting up or down, INTTAUDnIm is generated and the TAUDTTOUTm (slave) signal is set/reset.

  The counter continues count-up/-down and waits for the next INTTAUDnlm of master channel. Setting TAUDnTOL.TAUDnTOLm allows TAUDTTOUTm signal switching between normal phase and reverse phase during operation.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm = 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values.

Conditions

This function enables simultaneous rewrite. See **Section 18.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

$0000_H \leq$ TAUDnCDRm (master) $< FFFF_H$

Carrier cycle (down/up) = (TAUDnCDRm (master) + 1) × 2 × count clock cycle

Duty cycle [%] =
[(TAUDnCDRm (master) + 1 – TAUDnCDRm (slave)) / (TAUDnCDRm (master) + 1)] × 100

- Duty cycle = 100%
  TAUDnCDRm (slave) = 0000_H
- Duty cycle = 0%
  TAUDnCDRm (slave) ≥ TAUDnCDRm (master) + 1

(3) Block Diagram and General Timing Diagram

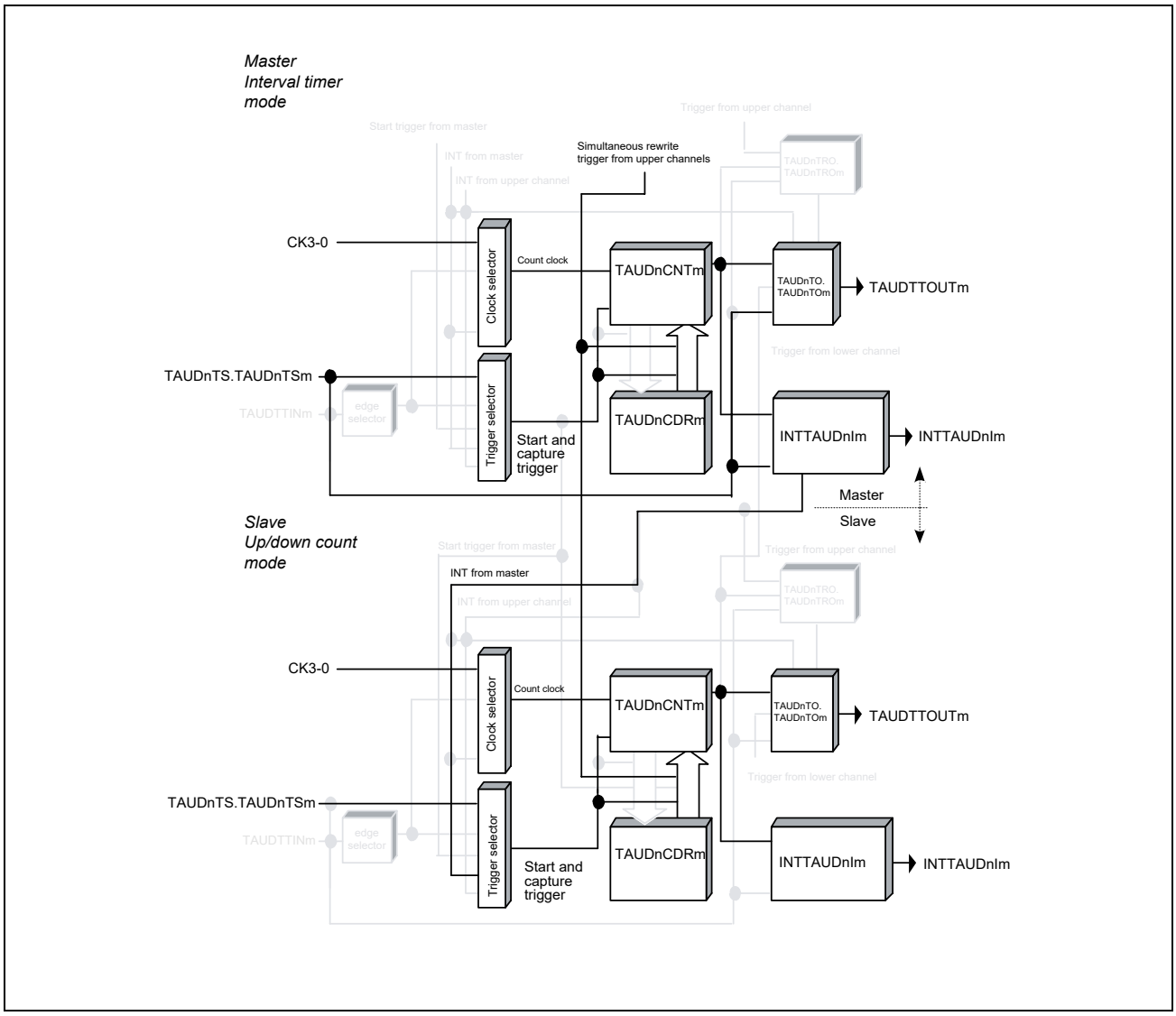


Figure 18.102 Block Diagram of Triangle PWM Output Function

The following settings apply to the general timing diagram.

- Master channel

INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

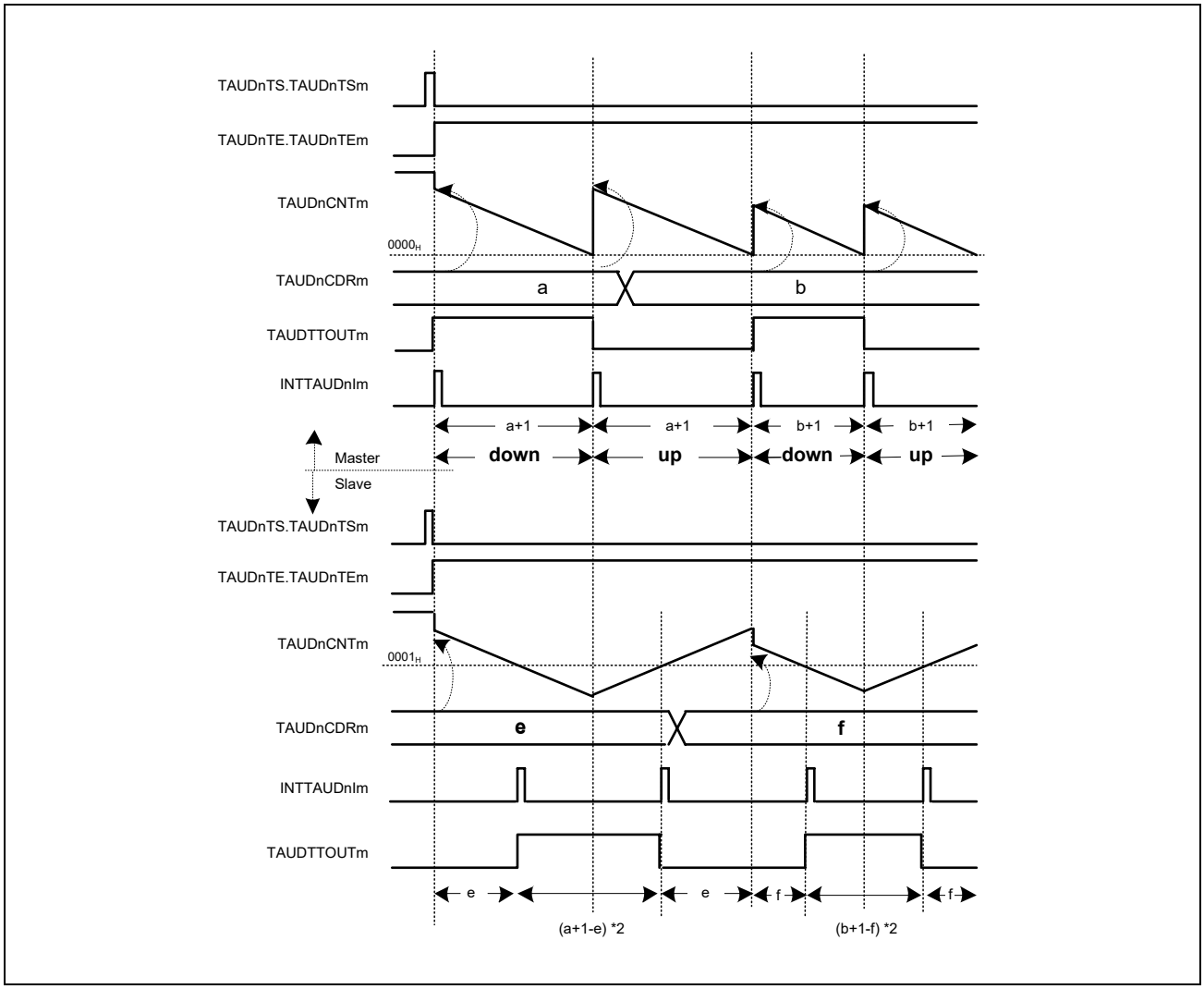


Figure 18.103 General Timing Diagram of Triangle PWM Output Function

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.163 Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated to toggle TAUDTTOUTm at the beginning of an operation. 1: INTTAUDnIm is generated and TAUDTTOUTm is toggled at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.164 Contents of TAUDnCMURm Register for Master Channels of Triangle PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

Table 18.165 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0.
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.166 Simultaneous Rewrite Settings for Master Channels of Triangle PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: A simultaneous rewrite trigger signal is generated when master channel starts to count and the corresponding slave channel is at the peak of a triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.167 Contents of TAUDnCMORm Register for Slave Channels of Triangle PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	111: Up/down output trigger signal of master channel
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1001: Up/down count mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.168 Contents of TAUDnCMURm Register for Slave Channels of Triangle PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels

Table 18.169 Control Bit Settings in Synchronous Channel Output Mode 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.170 Simultaneous Rewrite Settings for Slave Channels of Triangle PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: A simultaneous rewrite trigger signal is generated when master channel starts to count and the corresponding slave channel is at the peak of a triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for Triangle PWM Output Function

Table 18.171 Operating Procedure for Triangle PWM Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm (master) is generated on the master channel if TAUDnCMORm.TAUDnMD0 is set to 1.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master and slave channels is loaded into TAUDnCNTm to perform counting down. When the counter of master channel reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDTTOUTm (master) is toggled. • TAUDnCDRm value is reloaded into TAUDnCNTm (master) to continue count operation. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave) or counting is started in opposite direction. When TAUDnCNTm of slave channel reaches 0001_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is set in the count-down status or reset in count-up status.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Duty cycle = 0%

The following settings apply to the general timing diagram.

- Master channel:
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)
  - TAUDnCDRm = a = 5_H
- Slave channels:
  - TAUDnCDRm = 6_H

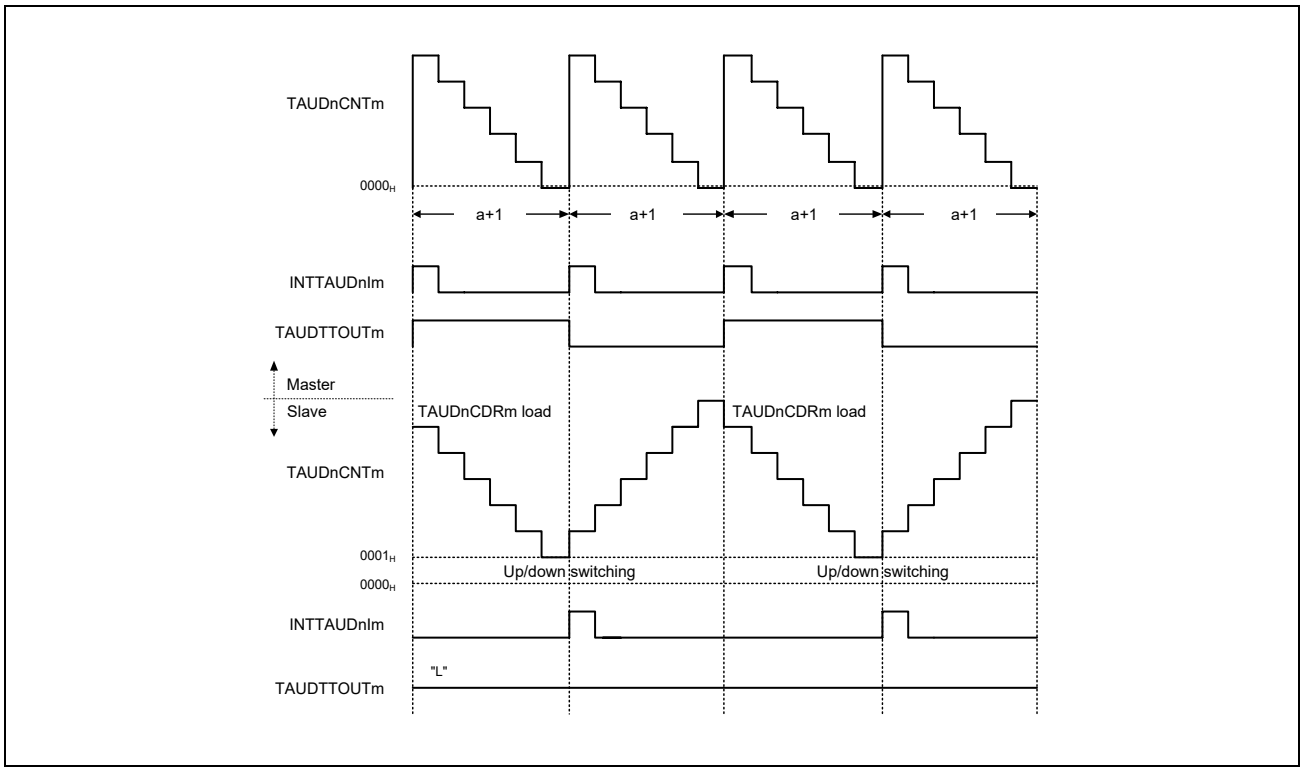


Figure 18.104 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1

- If the TAUDnCDRm (slave) value is greater than the TAUDnCDRm (master) value + 1, INTTAUDnIm of slave channel is not generated while counting down. TAUDTTOUTm remains low because there is no set signal to be detected.

(b) Duty cycle = 100%

The following settings apply to the general timing diagram.

- Master channel:
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)
  - TAUDnCDRm = a = 5_H
- Slave channels:
  - TAUDnCDRm = 0_H

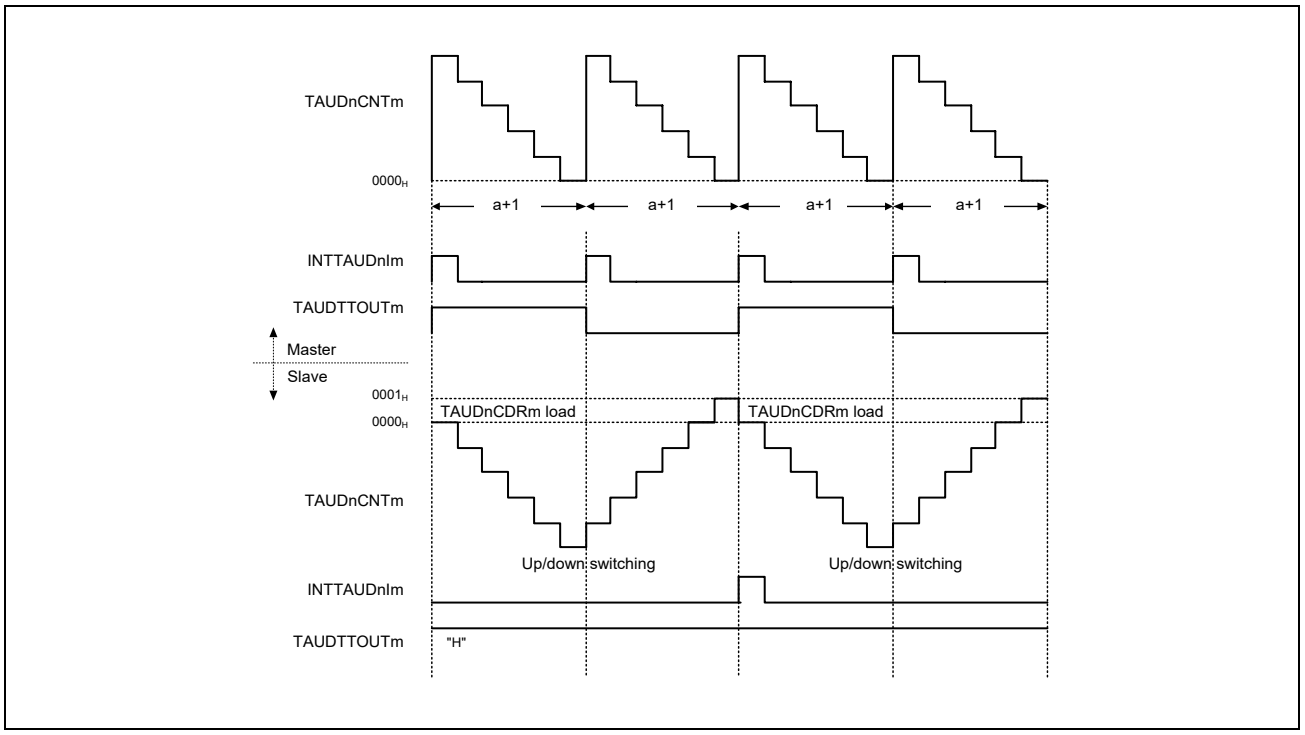


Figure 18.105 TAUDnCDRm (Slave) = 0000_H

- If TAUDnCDRm (slave) = 0000_H, INTTAUDnIm of slave channel is not generated while counting up. TAUDTTOUTm remains high because there is no reset signal to be detected.

18.4.12.8 Triangle PWM Output Function with Dead Time

(1) Overview

Summary

This function generates multiple triangle PWM outputs with a predefined dead time added by using a master and two or more slave channels. The resulting PWM signals are output via TAUDTTOUTm of the slave channels 2 and 3, enabling the pulse cycle (frequency) and the duty cycle of TAUDTTOUTm to be set using the master and slave channels.

Carrier cycles are generated on master channel. The first pulse controls the down status of slave counter and the second one controls the up status.

An interrupt on slave 2 causes TAUDTTOUTm of slave channels to be set/reset. Depending on the settings of TAUDnTDL.TAUDnTDLm, delay time is added to positive or negative logic side of the signal (i.e., whether TAUDTTOUTm is set/reset immediately or after dead time has elapsed). The duration of the dead time is specified by slave channel 3.

Prerequisites

- It requires three channels. For slave channels 2 and 3, select even-numbered channel CH (a) and odd-numbered channel CH (a + 1).
- The operating mode for master channels should be set to interval timer mode (see **Table 18.173, Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function with Dead Time**).
- Slave channel 1 is not used for this function. This ensures that slave channel 2 is an even-numbered channel, and slave channel 3 is an odd-numbered channel. Slave channel 1 can be used as a separate timer (independent function).
- The operating mode for slave channel 2 should be set to up/down count mode (see **Table 18.177, Contents of TAUDnCMORm Register for Slave Channel 2 of Triangle PWM Output Function with Dead Time**). Slave channel 2 should be an even-numbered channel.
- The operating mode for slave channel 3 should be set to one-count mode (see **Table 18.181 Contents of TAUDnCMORm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time**). Slave channel 3 should be an odd-numbered channel.
- The channel output mode for master channels should be set to independent channel output mode 1 (see **Section 18.4.4, Channel Output Modes**).
- The output mode for slave channels 2 and 3 should be set to synchronous channel output mode 2 with dead time output (see **Section 18.4.4, Channel Output Modes**).
- The following settings make a TAUDTTOUTm signal at high level during the down status of the carrier cycle:
  - If TAUDnCMORm.TAUDnMD0 (master) bit is set to 0, TAUDnTO.TAUDnTOm should be set to 1 while TAUDnTOE.TAUDnTOEm is set to 0. (recommended setting)
  - If TAUDnCMORm.TAUDnMD0 (master) bit is set to 1, TAUDnTO.TAUDnTOm should be set to 0 while TAUDnTOE.TAUDnTOEm is set to 0.

NOTE

The triangle PWM output function with dead time does not use slave channel 1.

Slave channel 1 can be used as a separate timer (independent function).

Functional description

The counter starts by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This makes TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from the TAUDnCDRm value. If TAUDnCMORm.TAUDnMD0 bit of master channel is set to 1, an interrupt is generated and the master's TAUDTTOUTm signal is toggled.

- Master channel:
  When the counter of master channel reaches 0000_H, an INTTAUDnIm is generated and the TAUDTTOUTm signal is toggled. The TAUDnCDRm value is reloaded to continue countdown.
- Slave channel 2:
  If INTTAUDnIm is generated on the master channel, the counter of slave channel 2 is triggered.
  - If the slave counter is counting down, the counting direction changes.
  - If the slave counter is counting up, the TAUDnCDRm value is reloaded and the counter starts to count down.

  The counter continues to count down/up and waits for the next INTTAUDnlm of master channel. When the counter value of slave channel 2 reaches 0001_H, INTTAUDnIm is generated.
- Slave channel 3:
  If INTTAUDnIm is generated on slave channel 2, the counter of slave channel 3 is triggered. The current value of TAUDnCDRm (slave 3) is loaded into TAUDnCNTm (slave 3) and the counter starts to count down from the TAUDnCDRm value.

  When the counter reaches 0000_H, INTTAUDnIm occurs. The counter returns to $FFFF_H$ and waits for the next INTTAUDnlm of slave channel 2.

As described in **Table 18.172, Operation of TAUDTTOUTm upon Occurrence of an Interrupt on Slave Channel 2**, the set/reset timing (right after occurrence of an interrupt or after dead time has elapsed) depends on the TAUDnTDL.TAUDnTDLm setting of the corresponding channel.

The setting of TAUDnTOL.TAUDnTOLm also determines whether a high level signal (TAUDnTOL.TAUDnTOLm = 0) or a low level signal (TAUDnTOL.TAUDnTOLm = 1) is output from the corresponding channel.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values.

TAUDTTOUTm can be 100% output by setting the TAUDnCDRm value of slave channel 2 to 0000_H.

Conditions

This function enables simultaneous rewrite. See **Section 18.4.3, Simultaneous Rewrite**.

TAUDnTOL.TAUDnTOLm and TAUDnTDL.TAUDnTDLm should be set before start of count operation. Slave channels 2 and 3 should have the opposite settings of TAUDnTDL.TAUDnTDLm.

Table 18.172 Operation of TAUDTTOUTm upon Occurrence of an Interrupt on Slave Channel 2

TAUDnTDL. TAUDnTDLm	Count Direction of Slave Channel 2 upon Occurrence of Interrupt	TAUDTTOUTm Set/Reset Timing
0	Down	Set after the dead time has elapsed
	Up	Reset right after interrupt occurs
1	Down	Set right after interrupt occurs
	Up	Reset after the dead time has elapsed

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

$0000_H \leq$ TAUDnCDRm (master) $< FFFF_H$

Carrier cycle (down/up) = (TAUDnCDRm (master) + 1) × 2 × count clock cycle

PWM signal width (normal phase)
= [(TAUDnCDRm (master) + 1 – TAUDnCDRm (slave 2)) × 2 – (TAUDnCDRm (slave 3) + 1)] × count clock cycle

PWM signal width (reverse phase)
= [(TAUDnCDRm (master) + 1 – TAUDnCDRm (slave 2)) × 2 + (TAUDnCDRm (slave 3) + 1)] × count clock cycle

(3) Block Diagram and General Timing Diagram

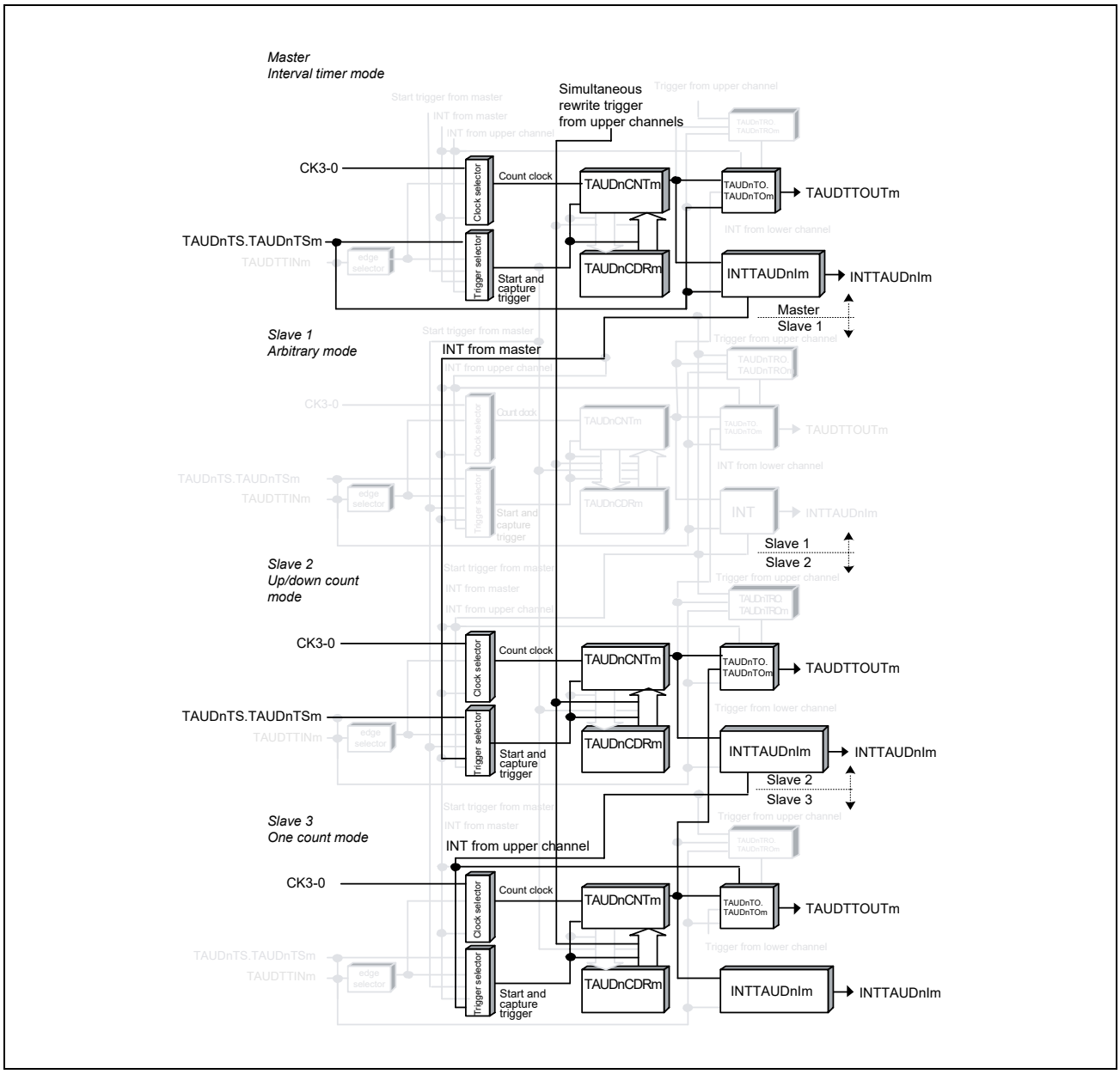


Figure 18.106 Block Diagram of Triangle PWM Output Function with Dead Time

The following settings apply to the general timing diagram.

- Master channel:
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)
- Slave channel 2:
  - INTTAUDnIm not generated at the beginning of operation.(TAUDnCMORm.TAUDnMD0 = 0)
  - TAUDnTDL.TAUDnTDLm = 0
  - Positive logic (TAUDnTOL.TAUDnTOLm = 0)
- Slave channel 3:
  - Enables start trigger detection during counting.(TAUDnCMORm.TAUDnMD0 = 1)
  - TAUDnTDL.TAUDnTDLm = 1
  - Positive logic (TAUDnTOL.TAUDnTOLm = 0)

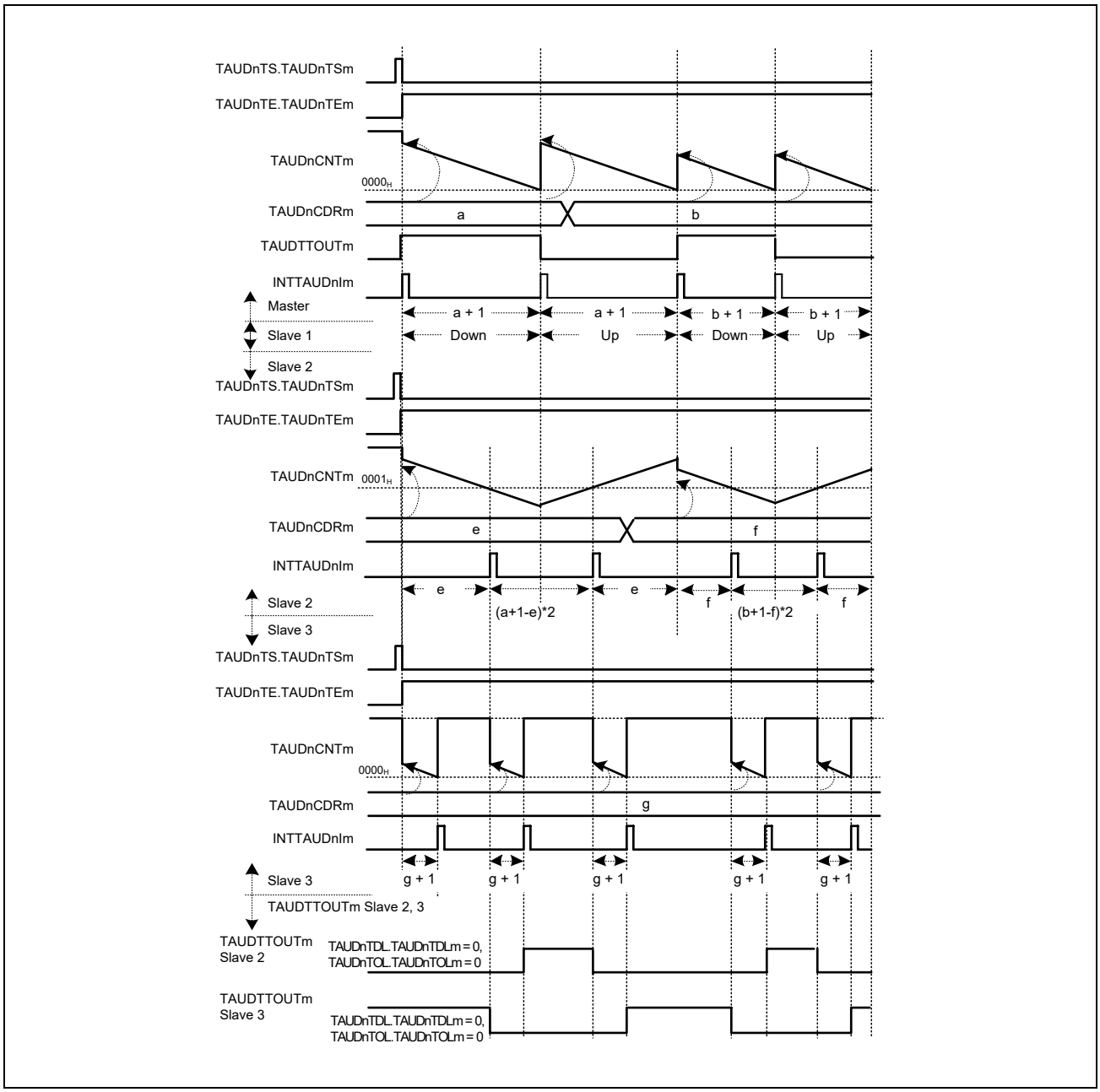


Figure 18.107 General Timing Diagram of Triangle PWM Output Function with Dead Time

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.173 Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated and TAUDTTOUTm is not toggled at the beginning of operation. 1: INTTAUDnIm is generated and TAUDTTOUTm is toggled at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.174 Contents of TAUDnCMURm Register for Master Channels of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

Table 18.175 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0.
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.176 Simultaneous Rewrite Setting for Master Channels of Triangle PWM Output Function with Dead Time

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

(5) Register Settings for Slave Channel 2

(a) TAUDnCMORm for slave channel 2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.177 Contents of TAUDnCMORm Register for Slave Channel 2 of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	111: Up/down output trigger signal of master channel
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1001: Up/down count mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation.

(b) TAUDnCMURm for slave channel 2

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.178 Contents of TAUDnCMURm Register for Slave Channel 2 of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 2

Table 18.179 Control Bit Settings in Synchronous Channel Output Mode 2 with Dead Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	0: Adds dead time if an interrupt is detected on an even upper channel and the conditions set by TAUDnTDL.TAUDnTDLm are satisfied.
TAUDnTDL.TAUDnTDLm	0: Adds dead time to normal phase. 1: Adds dead time to reverse phase.
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0.
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel.
TAUDnTME.TAUDnTMEm	0: Disables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm exclusively from odd-numbered channels.

(d) Simultaneous rewrite for slave channel 2

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.180 Simultaneous Rewrite Settings for Slave Channel 2 of Triangle PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Register Settings for Slave Channel 3

(a) TAUDnCMORm for slave channel 3

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.181 Contents of TAUDnCMORm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	110: Dead time output signal of the TAUDTTOUTm generation unit
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Enables start trigger detection while counting.

(b) TAUDnCMURm for slave channel 3

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.182 Contents of TAUDnCMURm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 3

Table 18.183 Control Bit Settings in Synchronous Channel Output Mode 2 with Dead Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	0: Adds dead time if an interrupt is detected on an even upper channel and the conditions set by TAUDnTDL.TAUDnTDLm are satisfied.
TAUDnTDL.TAUDnTDLm	0: Adds dead time to normal phase. 1: Adds dead time to reverse phase.
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0.
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel.
TAUDnTME.TAUDnTMEm	0: Disables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm exclusively from even-numbered channels.

(d) Simultaneous rewrite for slave channel 3

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.184 Simultaneous Rewrite Settings for Slave Channel 3 of Triangle PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(7) Operating Procedure for Triangle PWM Output Function with Dead Time

Table 18.185 Operating Procedure for Triangle PWM Output Function with Dead Time

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 2: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channel 2**. Slave channel 3: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (6), Register Settings for Slave Channel 3**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm (master) is generated on the master channel if TAUDnCMORm.TAUDnMD0 is set to 1.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master channel and slave channel 2 is loaded into TAUDnCNTm to perform counting down. When the counter of master channel reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCDRm value is reloaded into TAUDnCNTm (master) to continue count operation. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave 2) or counting is started in opposite direction. When TAUDnCNTm of slave channel 2 reaches 0001_H: • IINTTAUDnIm (slave 2) is generated. • TAUDnCDRm value of slave channel 3 is loaded into TAUDnCNTm perform counting down. When TAUDnCNTm of slave channel 3 reaches 0000_H: • INTTAUDnIm is generated.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(8) Specific Timing Diagrams

(a) Duty cycle = 0%

The following settings apply to the general timing diagram.

- Slave channel 2:
  - Positive logic (TAUDnTDL.TAUDnTDLm = 0)
- Slave channel 3:
  - Negative logic (TAUDnTDL.TAUDnTDLm = 1)

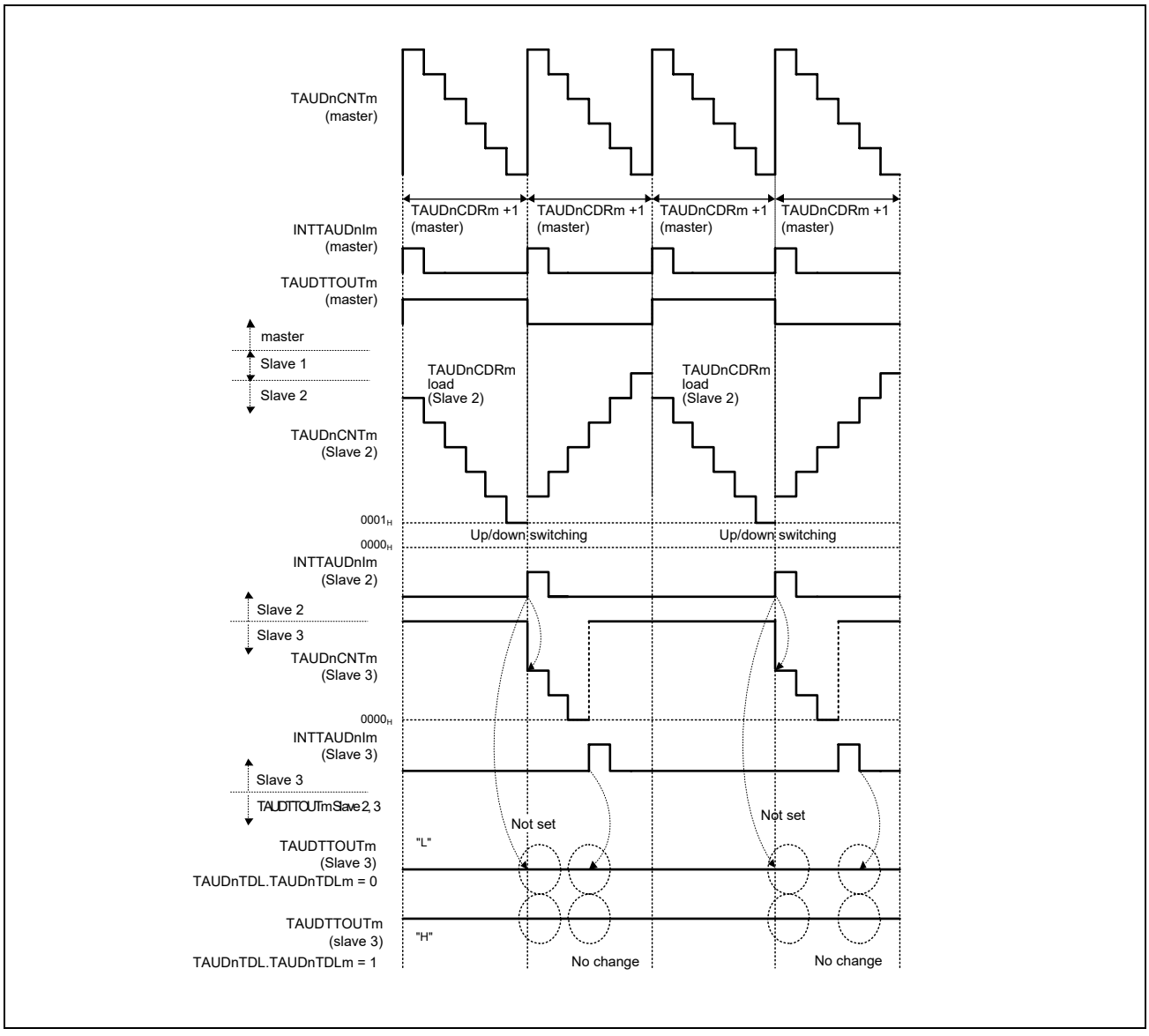


Figure 18.108 TAUDnCDRm (Slave 2) ≥ TAUDnCDRm (Master) + 1

- If TAUDnCDRm (slave 2) is greater than TAUDnCDRm (master), the counter of slave channel does not reach 0000_H while counting down. Therefore, TAUDTTOUTm signal is not set/reset and remains initial. This signal becomes a reset signal because an interrupt occurs on slave channel 2 during count-up operation.

(b) Duty cycle = 100%

The following settings apply to the general timing diagram.

- Slave channel 2:
  - Positive logic (TAUDnTDL.TAUDnTDLm = 0)
- Slave channel 3:
  - Negative logic (TAUDnTDL.TAUDnTDLm = 1)

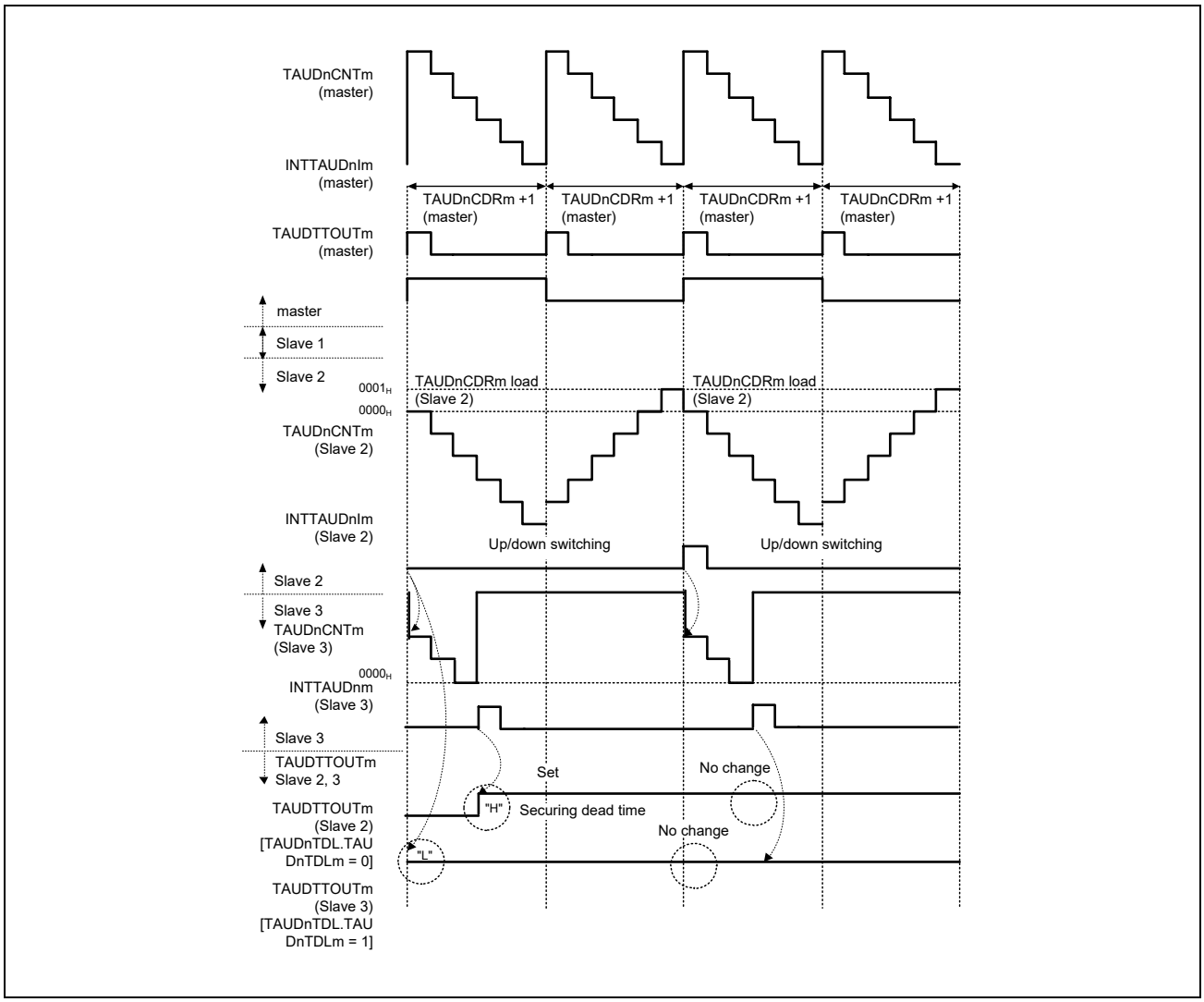


Figure 18.109 TAUDnCDRm (Slave) = 0000_H

- If TAUDnCDRm (slave 2) = 0000_H, the slave channel counter does not reach 0001_H while counting up. Therefore, no INTTAUDnlm occurs during count-up operation.
  - The set conditions for a channel with TAUDnTDL.TAUDnTDLm = 0 are met after elapse of dead time. TAUDTTOUTm is left in a newly set state even if a set/reset is made because no reset conditions are satisfied on such a channel.
  - Slave channel 3 in the above diagram is set when the counter starts. However, TAUDTTOUTm is left in an initial state on the slave channel with TAUDnTDL.TDLm = 1 because no reset conditions are satisfied on that channel.

18.4.12.9 A/D Conversion Trigger Output Function Type 2

(1) Overview

Summary

This function is identical to **Section 18.4.12.7, Triangle PWM Output Function**, except that TAUDTTOUTm is not output.

This function is enabled by setting channel output mode for the slave to independent channel output mode controlled by software.

(2) Block Diagram and General Timing Diagram

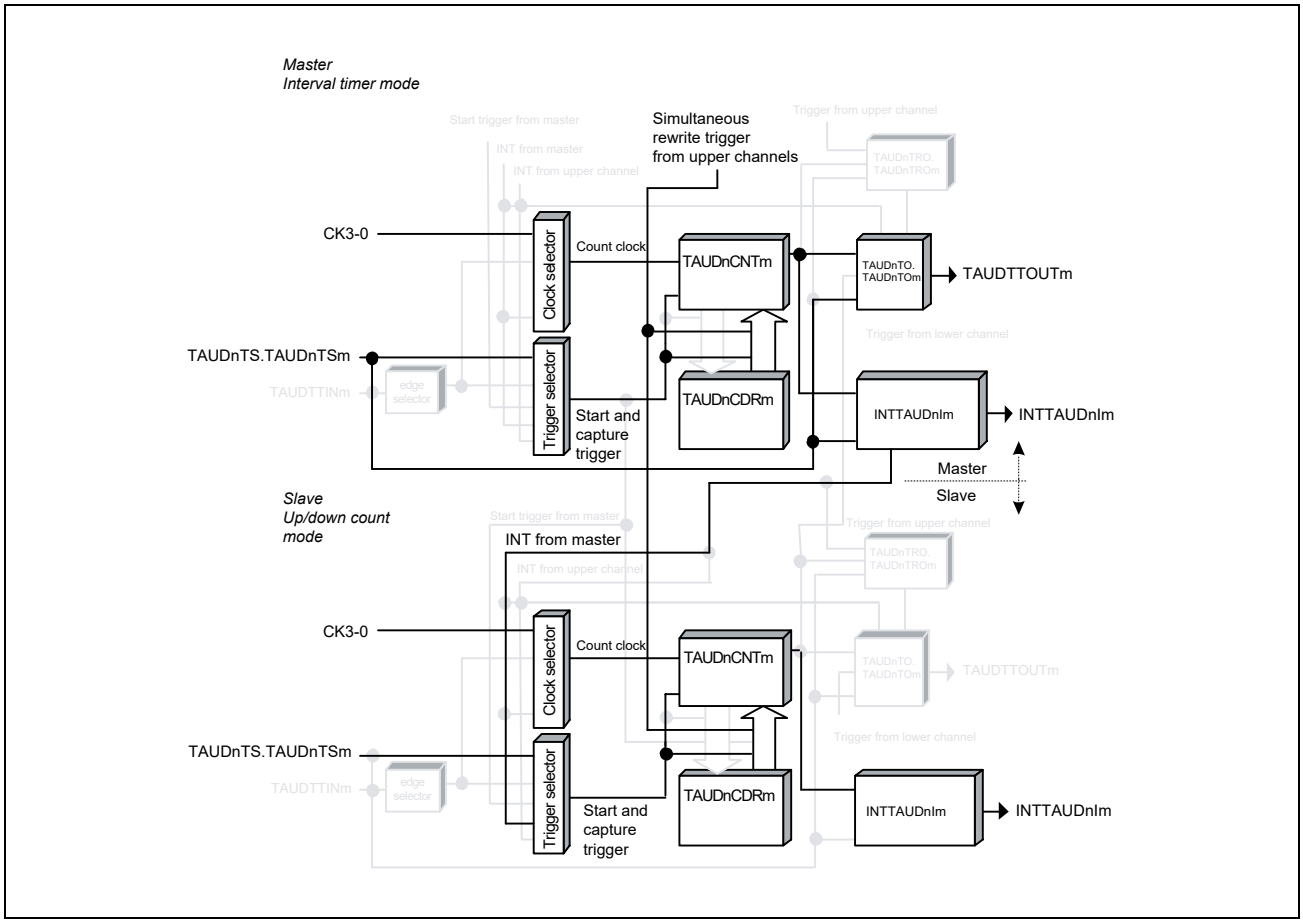


Figure 18.110 Block Diagram of A/D Conversion Trigger Output Function Type 2

The following settings apply to the general timing diagram.

- Master channel
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

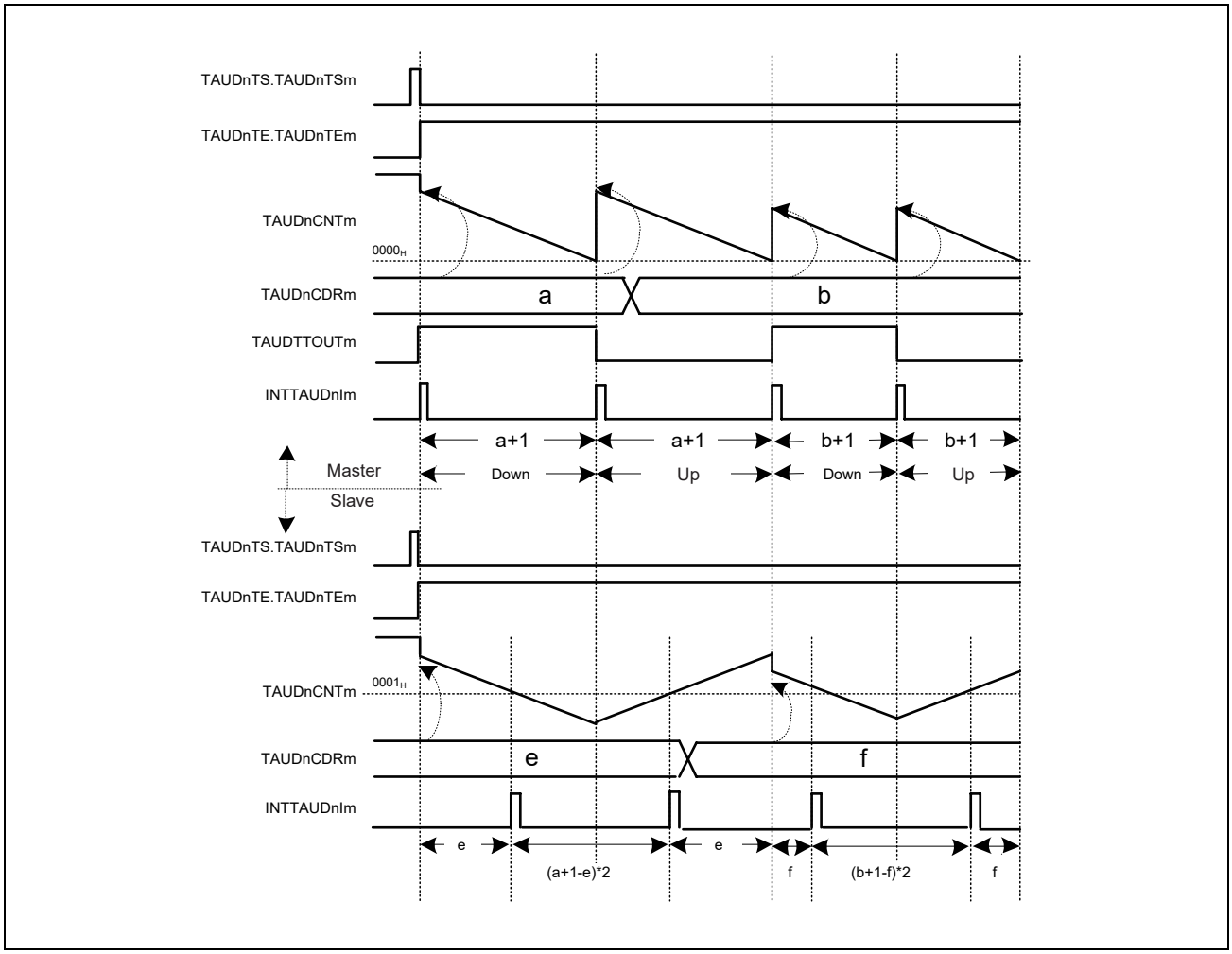


Figure 18.111 General Timing Diagram of A/D Conversion Trigger Output Function Type 2

18.4.12.10 Interrupt Request Signals Culling Function

(1) Overview

Summary

This function divides the number of interrupts of the master channel by a specified value using a slave channel.

The interrupt request signals Culling function is a sub function of the following functions:

- PWM Output Function (see **Section 18.4.12.1, PWM Output Function**)
- Triangle PWM Output Function (see **Section 18.4.12.7, Triangle PWM Output Function**)
- Triangle PWM Output Function with Dead Time
  (see **Section 18.4.12.8, Triangle PWM Output Function with Dead Time**)

Prerequisites

- It requires two channels.
- The operation mode of the master channel must be set to interval timer mode (see **Table 18.186, Contents of TAUDnCMORm Register for Master Channels of Interrupt Request Signals Culling Function**).
- The operation mode of the slave channel must be set to Event Count Mode (see **Table 18.189, Contents of TAUDnCMORm Register for Slave Channels of Interrupt Request Signals Culling Function**).
- TAUDTTOUTm is not used for the master or slave channel of this function.

Functional description

The counters (master and slave) are started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1 for both channels. This in turn sets TAUDnTE.TAUDnTEm, enabling count operation. The current value of the data register of the master channel and slave channel (TAUDnCDRm) are written to the counter (TAUDnCNTm).

- Master channel:
  When the counter of the master channel reaches 0000_H, INTTAUDnIm is generated and TAUDnCDRm value is reloaded to TAUDnCNTm.
- Slave channel:
  Every time the master channel generates an INTTAUDnIm, the counter of the slave channel reduces by one. When the counter reaches 0000_H, it awaits the next interrupt from the master channel. This causes TAUDnCNTm (slave) to reload the value of TAUDnCDRm, and an INTTAUDnIm is generated.

Forced restart is not possible for this function. The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1 for the master and slave channel(s), which in turn sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm of master and slave channel(s) stop but retain their values.

Conditions

This function enables simultaneous rewrite. See **Section 18.4.3, Simultaneous Rewrite**.

(2) Equations

Interrupt division operator = TAUDnCDRm (slave channel)

- One INTTAUDnlm is generated for the INTTAUDnlm count of the master channel defined by TAUDnCDRm (slave channel) + 1.

(3) Block Diagram and General Timing Diagram

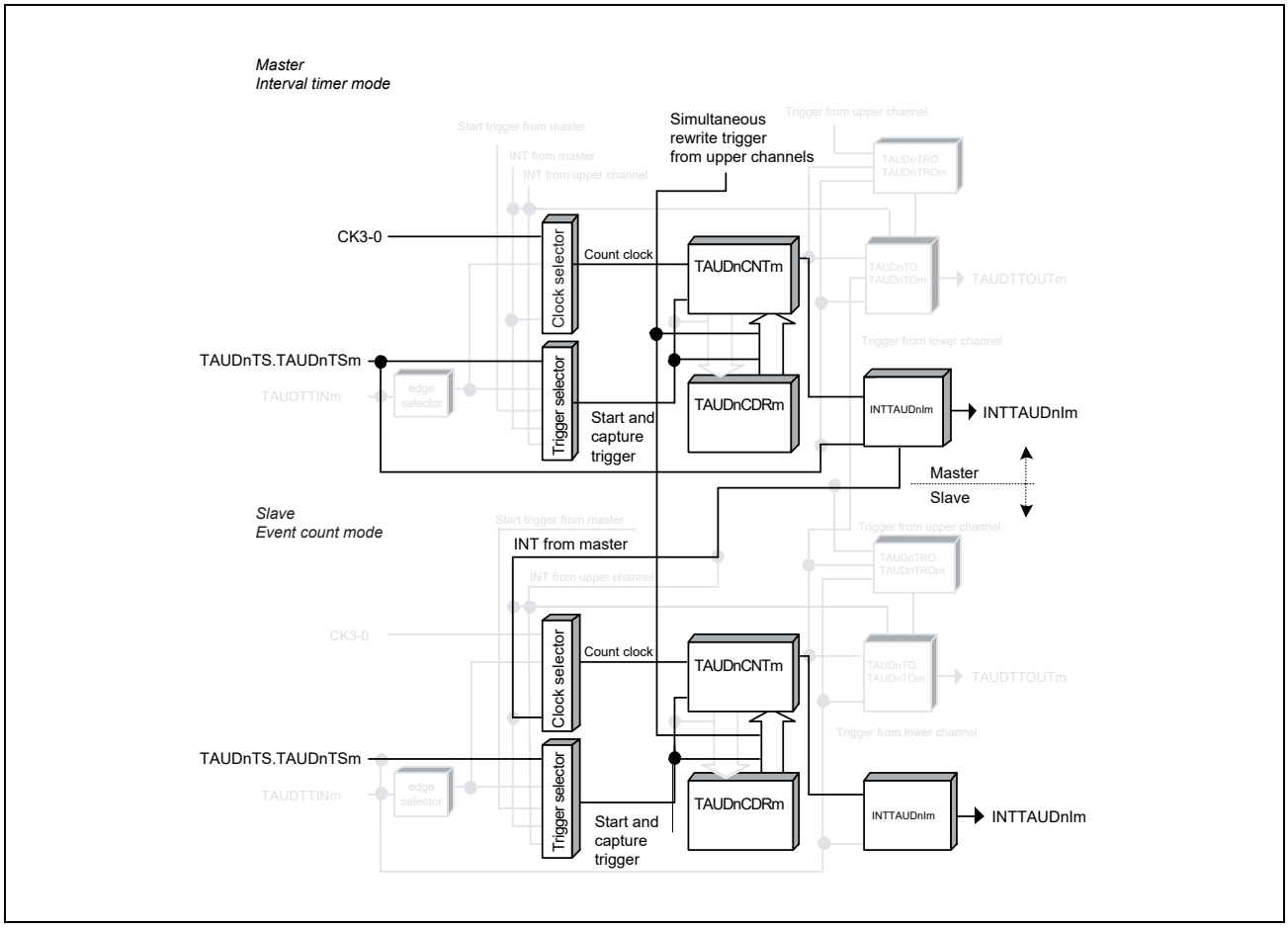


Figure 18.112 Block Diagram of Interrupt Request Signals Culling Function

The following settings apply to the general timing diagram.

- Master channel:
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

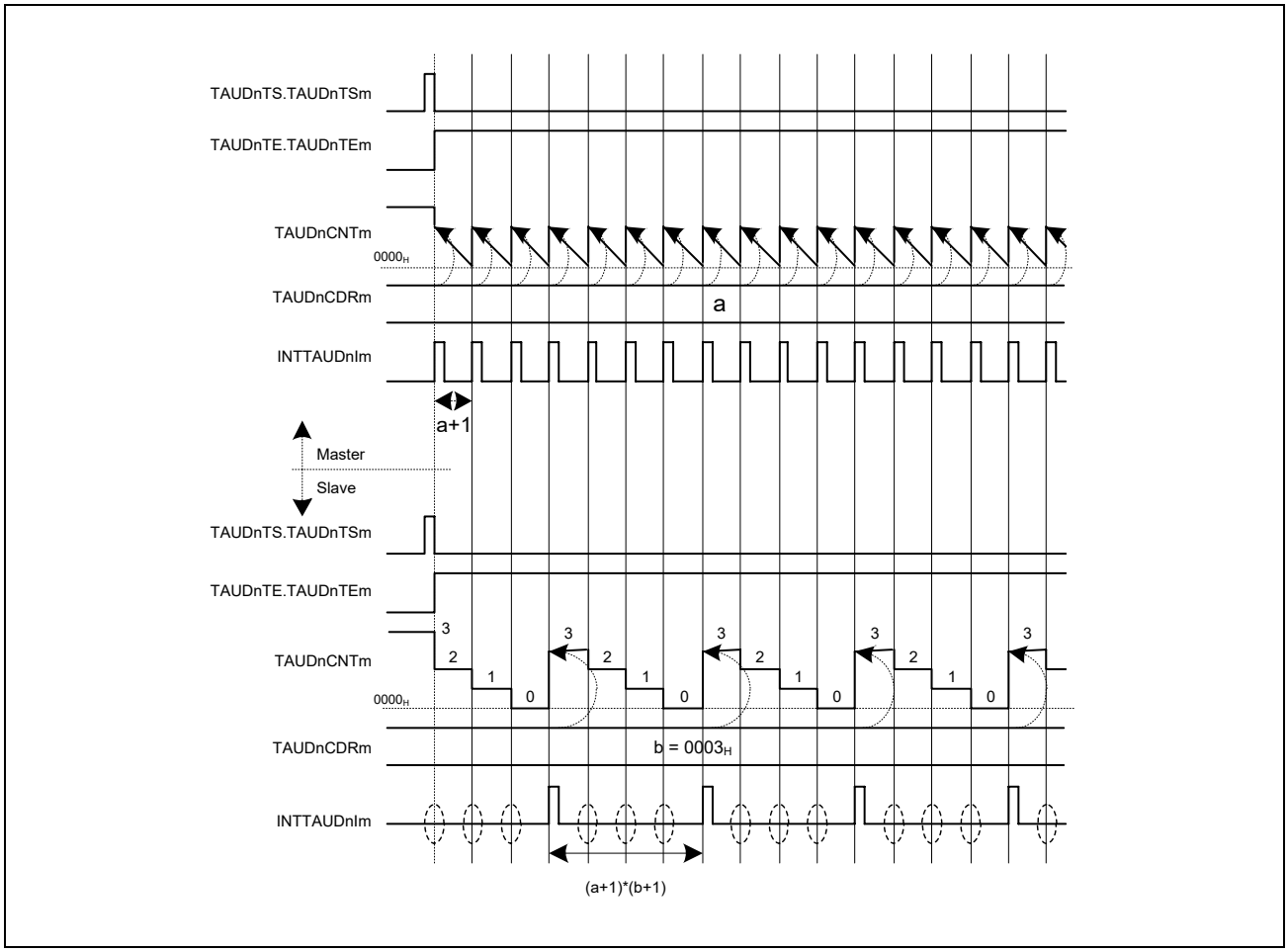


Figure 18.113 General Timing Diagram of Interrupt Request Signals Culling Function

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.186 Contents of TAUDnCMORm Register for Master Channels of Interrupt Request Signals Culling Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.187 Contents of TAUDnCMURm Register for Master Channels of Interrupt Request Signals Culling Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.188 Simultaneous Rewrite Settings for Master channels of Interrupt Request Signals Culling Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count. 1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.189 Contents of TAUDnCMORm Register for Slave Channels of Interrupt Request Signals Culling Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	11: INTTAUDnIm of the master channel is used as the count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0011: Event count mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.190 Contents of TAUDnCMURm Register for Slave Channels of Interrupt Request Signals Culling Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for the slave channel

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.191 Simultaneous Rewrite Settings for Slave Channels of Interrupt Request Signals Culling Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count. 1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for Interrupt Request Signals Culling Function

Table 18.192 Operating Procedure for Interrupt Request Signals Culling Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm is generated on the master channel.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCNTm of master channel loads TAUDnCDRm value and counts down. When the counter reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCNTm (master) loads TAUDnCDRm value and continues count operation. • TAUDnCNTm of slave channels counts down each time INTTAUDnIm of master channel is detected. When TAUDnCNTm of the slave = 0000_H: • INTTAUDnIm (slave) occurs. • TAUDnCNTm (slave) loads the TAUDnCDRm value and continues counting.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Interrupt count (master) = interrupt count (slave)

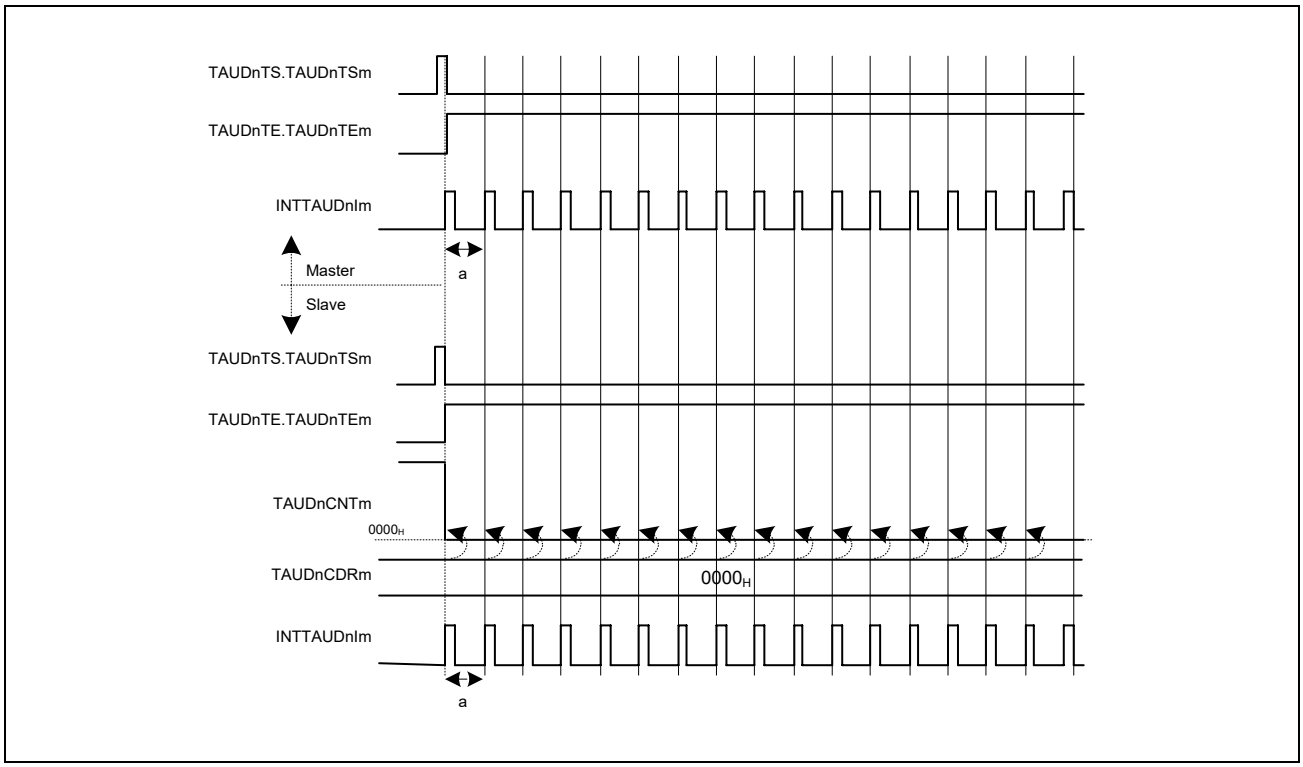


Figure 18.114 TAUDnCDRm (Slave) = 0000_H

- If TAUDnCDRm = 0000_H, TAUDnCDRm value of slave channels is loaded into TAUDnCNTm each time INTTAUDnIm of master channel is detected. In other words, TAUDnCNTm is always 0000_H.
- Therefore, an interrupt occurs on the master channel and simultaneously an interrupt occurs on slave channels.

18.4.12.11 One-Phase PWM Output Function

(1) Overview

Summary

This function adds dead time to a TAUDTTINm input signal. The resulting PWM signal is output via TAUDTTOUTm of the channel and TAUDTTOUTm of upper channels.

Prerequisites

- Each of two (or more) channels is enabled for dead time control (TAUDnTDE.TAUDnTDEm = 1).
- The operating mode for the lower channel should be set to one-count mode (see **Table 18.194, Contents of TAUDnCMORm Register for One-Phase PWM Output Function**).
- Any operating mode can be set to upper channels.
- Channel output mode for upper and lower channels should be set to synchronous channel output mode 2 with one-phase PWN output (see **18.4.4, Channel Output Modes**).

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation.

The counter starts when an effective TAUDTTINm input start edge is detected. The value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from the TAUDnCDRm value.

When the counter reaches 0000_H, an interrupt occurs. The counter is reset to $FFFF_H$ and waits for the next effective TAUDTTINm input start edge.

Table 18.193 TAUDTTOUTm to which Dead Time is Added and State of TAUDTTINm

TAUDnCMURm. TAUDnTIS[1:0]	TAUDnTOL. TAUDnTOLm	TAUDTTOUTm to which Dead Time is Added	TAUDnTDL. TAUDnTDLm	TAUDTTINm_lower State when Added
10	0	TAUDTTOUTm low	0	High
			1	Low
	1	TAUDTTOUTm high	0	High
			1	Low
11	0	TAUDTTOUTm low	0	Low
			1	High
	1	TAUDTTOUTm high	0	Low
			1	High

Conditions

- TAUDnCMURm.TAUDnTIS[1:0] bits specify the type of width measurement:
  - TAUDnCMURm.TAUDnTIS[1:0] = 10_B: Uses both edges as effective edges for detection (Low width measurement).
  - TAUDnCMURm.TAUDnTIS[1:0] = 11_B: Uses both edges as effective edges for detection (High width measurement).
- The TAUDnTDL.TAUDnTDLm bit specifies the operation of TAUDTTOUTm for each channel when an interrupt or effective TAUDTTINm edge is detected on the lower channel:
  - If TAUDnTDL.TAUDnTDLm = 0, an interrupt is used as a TAUDTTOUTm set trigger and an effective TAUDTTINm edge as a TAUDTTOUTm reset trigger.
  - If TAUDnTDL.TAUDnTDLm = 1, an effective TAUDTTINm edge is used as a TAUDTTOUTm set trigger and an interrupt as a TAUDTTOUTm reset trigger.
- This function cannot make a forced restart.

(2) Block Diagram and General Timing Diagram

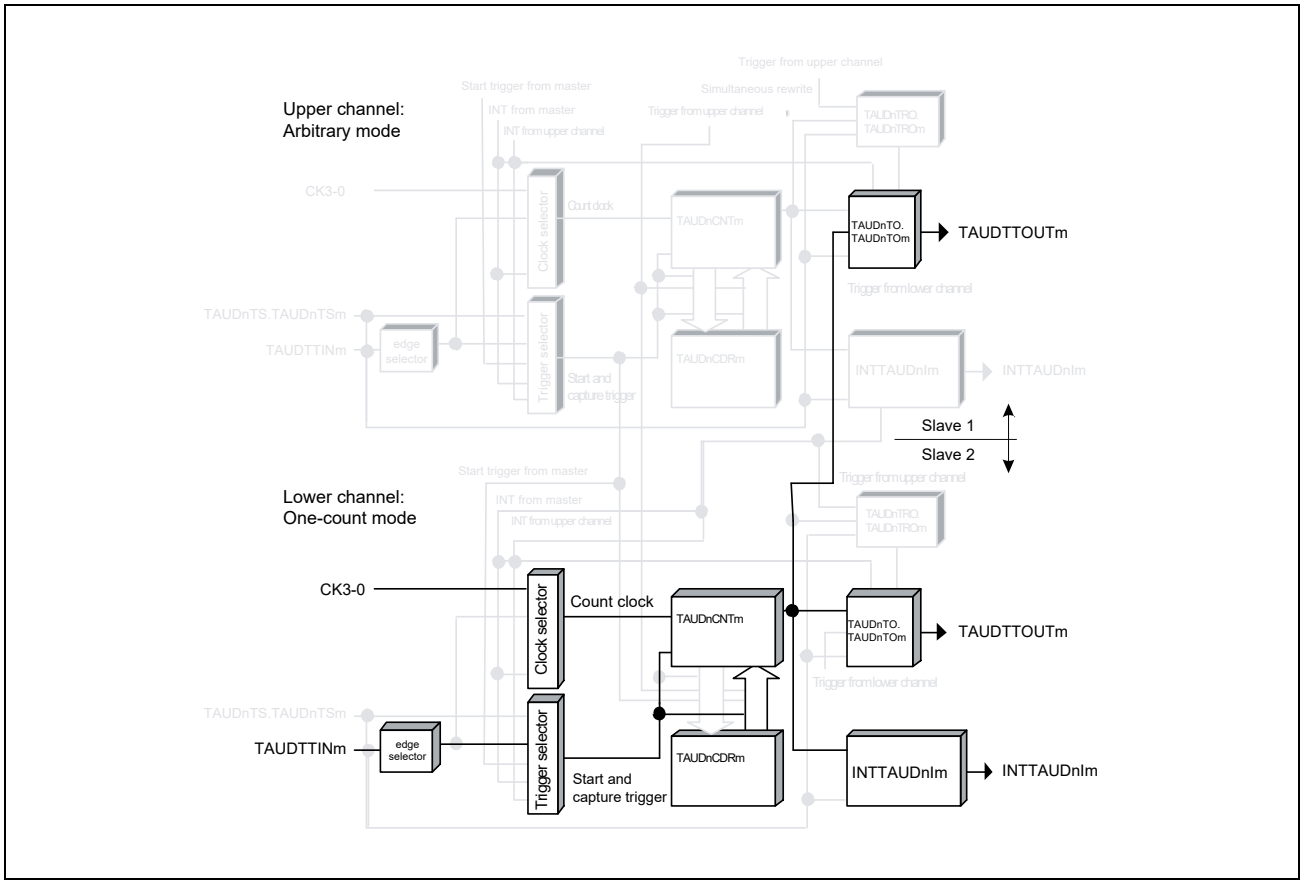


Figure 18.115 Block Diagram of One-Phase PWM Output Function

The following settings apply to the general timing diagram.

- Detection of rising and falling edges = high width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)

This setting considers a duty as active high.

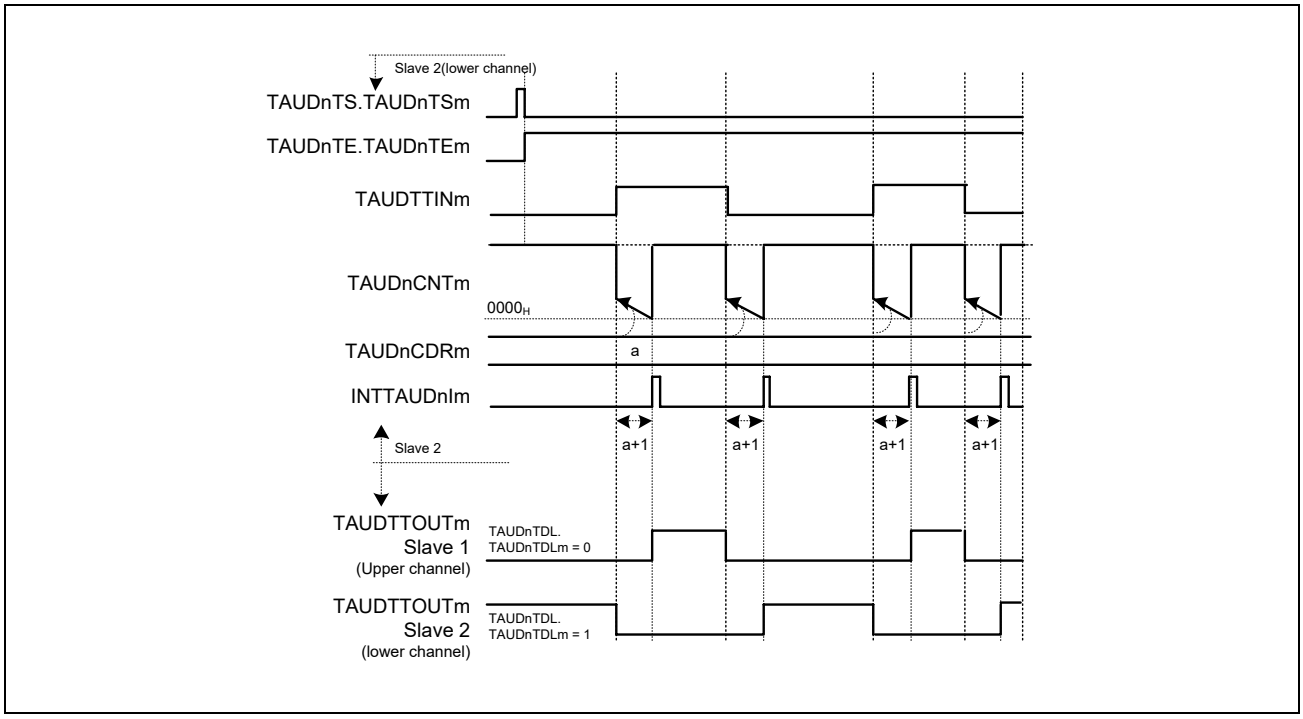


Figure 18.116 General Timing Diagram of One-Phase PWM Output Function

(3) Register Settings for Lower Channels

(a) TAUDnCMORm for lower channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS[1:0]		TAUDnCCS[1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS[1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.194 Contents of TAUDnCMORm Register for One-Phase PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: Effective edge of the TAUDTTINm input signal is used as an external start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Enables start trigger detection while counting.

(b) TAUDnCMURm for lower channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.195 Contents of TAUDnCMURm Register for One-Phase PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	10: Detection of rising and falling edges (low width measurement) 11: Detection of rising and falling edges (high width measurement)

(c) Channel output mode for lower channels

Table 18.196 Control Bit Settings in Synchronous Channel Output Mode 2 with One-Phase PWM Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	1: Adds dead time upon detection of a TAUDTTINm input edge on a lower odd-numbered channel.
TAUDnTDL.TAUDnTDLm	0: Adds dead time of the positive-phase width. 1: Adds dead time of the negative-phase width.
TAUDnTRE.TAUDnTREm	0: Disables real-time output.
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0.
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel.
TAUDnTME.TAUDnTMEm	0: Disables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm to upper channels exclusively.

(d) Simultaneous rewrite for lower channels

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 18.197 Simultaneous Rewrite Settings for One-Phase PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(4) Register Settings for Upper Channels

(a) TAUDnCMORm for upper channels

TAUDnCMORm register for upper channels can be set arbitrarily.

(b) TAUDnCMURm for upper channels

TAUDnCMURm register for upper channels can be set arbitrarily.

(c) Channel output mode for upper channels

Table 18.198 Control Bit Settings for Upper Channels in Synchronous Channel Output Mode 2 with One-Phase PWM Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	1: Adds dead time upon detection of a TAUDTTINm input edge on a lower odd-numbered channel.
TAUDnTDL.TAUDnTDLm	0: Uses a lower channel interrupt as a TAUDTTOUTm set trigger and an effective lower channel TAUDTTINm edge as a TAUDTTOUTm reset trigger. 1: Uses an effective lower channel TAUDTTINm edge as a TAUDTTOUTm set trigger and a lower channel interrupt as a TAUDTTOUTm reset trigger.
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set this bit to 0.
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel.
TAUDnTME.TAUDnTMEm	0: Disables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm to lower channels exclusively.

(d) Simultaneous rewrite for upper channels

Simultaneous rewrite register for upper channels can be set arbitrarily.

(5) Operating Procedure for One-phase PWM Output Function

Table 18.199 Operating Procedure for One-phase PWM Output Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers for the lower channel as described in **Table 18.194, Contents of TAUDnCMORm Register for One-Phase PWM Output Function** and **Table 18.195, Contents of TAUDnCMURm Register for One-Phase PWM Output Function**. Set TAUDnCMORm and TAUDnCMURm registers for the upper channel as described in **Section (4), Register Settings for Upper Channels**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 18.196, Control Bit Settings in Synchronous Channel Output Mode 2 with One-Phase PWM Output**.	Channel operation is stopped.
Start Operation	Set TAUDnTOE.TAUDnTOEm (slave channels 1 and 2) to 1 (at restart time only). Set TAUDnTS.TAUDnTSm = 1 for slave channel 2. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0. Detection of TAUDTTINm start edge	TAUDnTE.TAUDnTEm is set to 1 (slave channel 2) and TAUDnCNTm waits for detection of TAUDTTINm start edge. TAUDnCNTm loads TAUDnCDRm value.
During Operation	The TAUDnCDRm register value can be changed at any time. The TAUDnCNTm register can be read at any time.	TAUDnCNTm of slave channel 2 counts down. When the counter reaches 0000_H: • INTTAUDnIm is generated. • TAUDnCNTm stops counting. TAUDTTOUTm is changed by a TAUDTTINm edge detection signal and slave channel 2 INTTAUDnIm signal to output one-phase PWM waveform with dead time. Afterwards, this operation is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm = 1 for slave channel 2. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm and TAUDTTOUTm retain their current values.

Restart

18.4.13 Synchronous Non-Complementary and Complementary Modulation OutputFunctions

This section describes functions that generate six-phase PWM output or triangle PWM output by using a master channel and seven slave channels.

- **Section 18.4.13.1, Non-Complementary Modulation Output Function Type 1**
- **Section 18.4.13.2, Non-Complementary Modulation Output Function Type 2**
- **Section 18.4.13.3, Complementary Modulation Output Function**

18.4.13.1 Non-Complementary Modulation Output Function Type 1

(1) Overview

Summary

This function outputs a PWM signal, a high-level signal, or a low-level signal from TAUDTTOUTm depending on the values of the real-time output bits (TAUDnTRO.TAUDnTROm) and the modulation output enable bits (TAUDnTME.TAUDnTMEm) of a pair of slave channels. Three pairs of channels are typically used.

Prerequisites

- It requires one master channel and seven slave channels.
- The operation mode of the master channel must be set to interval timer mode (see **Table 18.201, Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 1**).
- The operating mode for slave channels 1 to 7 should be set to one-count mode (see **Table 18.204, Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1**).
- TAUDTTOUTm is not used with the master channel of this function.
- TAUDTTOUTm of slave channel 1 is not used with this function, but TAUDnTRC.TRCm should be set to 1 (see **18.4.4, Channel Output Modes**).
- The channel output mode for slave channels 2 to 7 should be set to synchronous channel output mode 1 with non-complementary modulation output (see **18.4.4, Channel Output Modes**).
- TAUDnCDRm of slave channel 1 should be set to 0000_H.

Functional description

The master/slave channel counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The value of data register (TAUDnCDRm) is loaded into the counter (TAUDnCNTm) and the counter starts to count down. When the counter reaches 0000_H, INTTAUDnlm occurs.

- Slave channel 1:
  Slave channel 1 is set as a channel that triggers real-time output (TAUDnTRC.TAUDnTRCm =1). If an interrupt occurs on slave channel 1 (TAUDnCDRm is fixed to 0000_H), the value of real-time output bit (TAUDnTRO.TAUDnTROm) of the channel that has detected the interrupt on slave channel 1 changes. After that, the counter returns to $FFFF_H$ and waits for the next interrupt of master channel.
- Slave channel 2:
  Slave channel 2 generates a PWM output. The master channel specifies a PWM output cycle and slave channel 2 specifies a duty cycle. After generating an interrupt, the counter returns to $FFFF_H$ and awaits the next interrupt from the master channel.

Slave channels 3 to 7 operate like slave channel 2.

As described in **Table 18.200, TAUDTTOUTm Output from One Pair of Slave Channels of Non-Complementary Modulation Output Function Type 1**, a signal output from TAUDTTOUTm depends on the value of the real-time output bit (TAUDnTRO.TAUDnTROm) and modulation output bit (TAUDnTME.TAUDnTMEm) of slave channel.

This function cannot use a forced restart. The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

- If TAUDnTME.TMEm = 0 on slave channels 2 to 7:
  - If the channel's TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm outputs a high-level signal.
  - If the channel's TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm outputs a low-level signal.
- If TAUDnTME.TMEm = 1 on slave channels 2 to 7:
  - If the channel's TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm outputs PWM corresponding to the channel.
  - If the channel's TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm outputs a low-level signal.
- If TAUDnTOL.TAUDnTOLm is set to 1, high-level and low-level signals output from TAUDTTOUTm are inverted. The PWM signal becomes inverted logic. Only the initial setting of TAUDnTOL.TAUDnTOLm is permitted (cannot be changed during operation).

Table 18.200 TAUDTTOUTm Output from One Pair of Slave Channels of Non-Complementary Modulation Output Function Type 1

TAUDnTME.TAUDnTMEm	TAUDnTRO.TAUDnTROm	TAUDTTOUTm Output
0	0	Low level
	1	High level
1	0	Low level
	1	PWM (positive logic)

- This function enables simultaneous rewrite. See **Section 18.4.3, Simultaneous Rewrite**.
- TAUDnCDRm value of slave channel 1 should be set to 0000_H so that a real-time output is triggered at the same time with PWM generation on slave channels 2 to 7.
- If TAUDnTOL.TAUDnTOLm is set to 0 on slave channels 2 to 7, TAUDnTO.TAUDnTOm is set low before TAUDnTE.TAUDnTEm is set to 0.
- If TAUDnTOL.TAUDnTOLm is set to 1 on slave channels 2 to 7, TAUDnTO.TAUDnTOm is set high before TAUDnTE.TAUDnTEm is set to 0.

(2) Equations

Slave channels 2 to 7:

PWM output cycle = [TAUDnCDRm (master) + 1] × count clock cycle

PWM output duty time = [TAUDnCDRm (slave)] × count clock cycle

(3) Block Diagram and General Timing Diagram

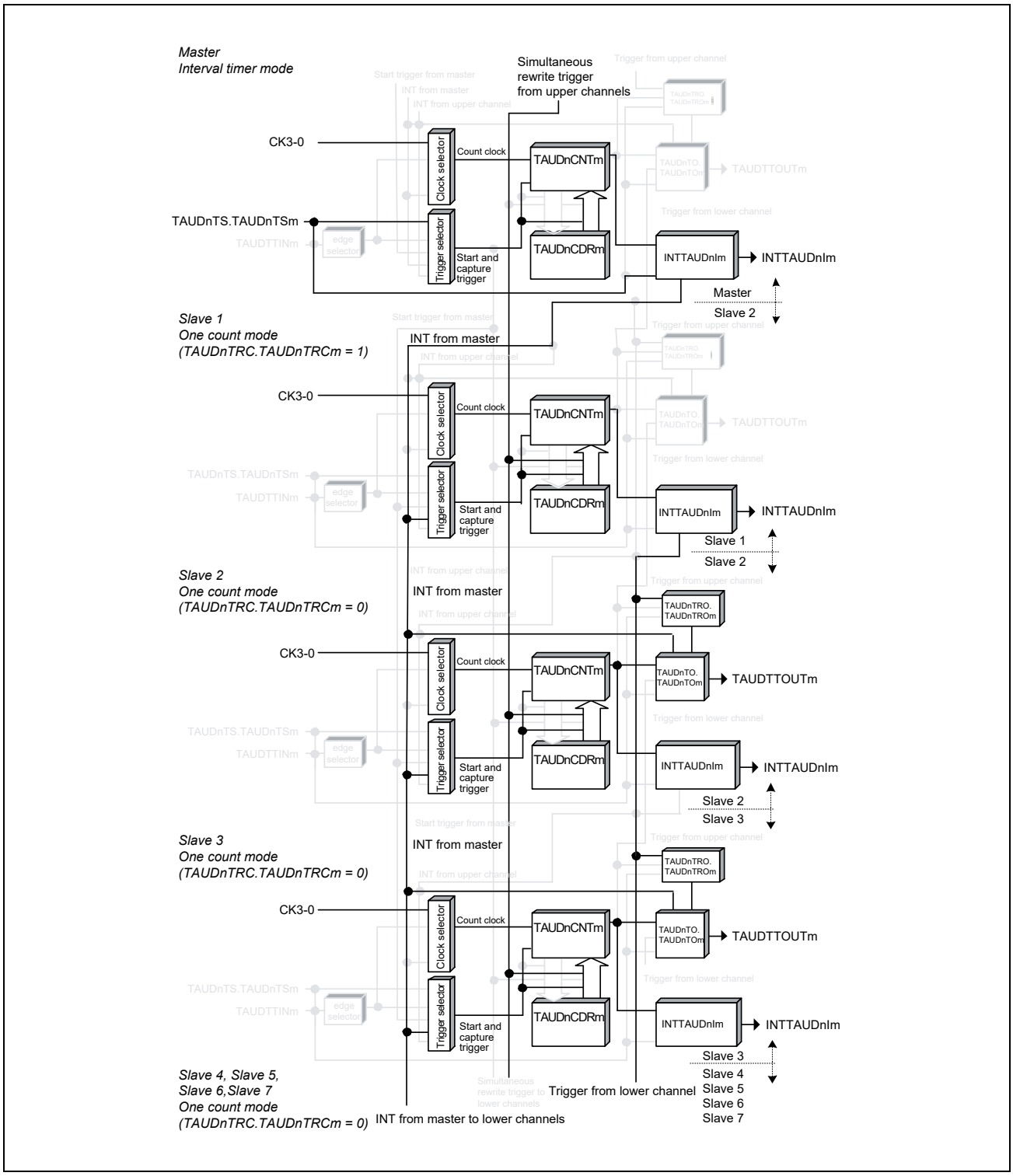


Figure 18.117 Block Diagram of Non-Complementary Modulation Output Function Type 1

The following settings apply to the general timing diagram.

- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

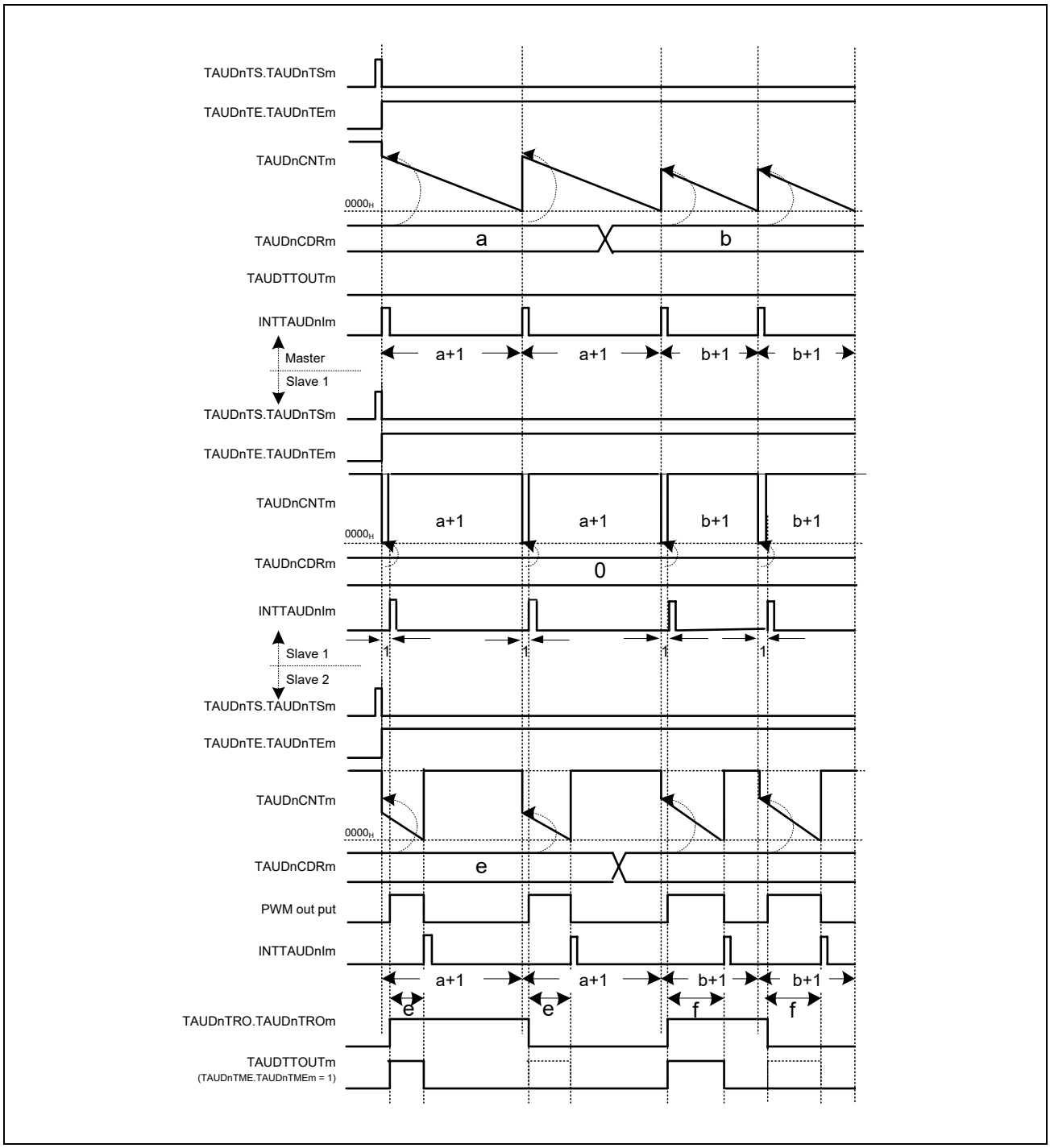


Figure 18.118 General Timing Diagram of Non-Complementary Modulation Output Function Type 1

NOTE

TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.201 Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	1: INTTAUDnIm is generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.202 Contents of TAUDnCMURm Register for Master Channels of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used with this function.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.203 Simultaneous Rewrite Settings for Master Channels of Non-Complementary Modulation Output Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

Conduct operation settings under the following conditions.

- Simultaneous rewrite trigger output function type 1 setting channel:
  TAUDnRDCm = 1, TAUDnRDSm = 1
  In addition, TAUDnCDRm settings for this channel are as follows.
  = ((TAUDnCDRm setting for the master channel targeted for simultaneous rewrite + 1) × Interrupt count) – 1
- Master channel: TAUDnRDCm = 0, TAUDnRDSm = 1
- Slave channel: TAUDnRDCm = 0, TAUDnRDSm = 1

(5) Register Settings for Slave Channel 1

(a) TAUDnCMORm for slave channel 1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.204 Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Start trigger during operation is enabled.

(b) AUDnCMURm for slave channel 1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.205 Contents of TAUDnCMURm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used on slave channel 1 with this function.

CAUTION

TAUDnTRC.TAUDnTRCm should be set to 1 because slave channel 1 is used as a real-time output trigger channel.

(d) Simultaneous rewrite for slave channel 1

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.206 Simultaneous Rewrite Settings for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(6) Register Settings for Slave Channels 2 to 7

(a) TAUDnCMORm for slave channels 2 to 7

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.207 Contents of TAUDnCMORm Register for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Start trigger during operation is enabled.

(b) TAUDnCMURm for slave channels 2 to 7

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.208 Contents of TAUDnCMURm Register for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels 2 to 7

Table 18.209 Control Bit Settings for Slave Channels 2 to 7 in Synchronous Channel Output Mode 1 with Non-Complementary Modulation Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation 1: Enables modulation

(d) Simultaneous rewrite of slave channels 2 to 7

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.210 Simultaneous Rewrite Settings for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(7) Operating Procedure for Non-Complementary Modulation Output Function Type 1

Table 18.211 Operating Procedure for Non-Complementary Modulation Output Function Type 1 (1/2)

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 1: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channel 1**. Slave channels 2 to 7: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (6), Register Settings for Slave Channels 2 to 7**. Set the value of TAUDnCDRm register of every channel. Set a pulse cycle with TAUDnCDRm of master channel, 0000_H in TAUDnCDRm of slave channel 1, and duty width with TAUDnCDRm of slave channels 2 to 7. Set TAUDnTRC.TAUDnTRCm to 1 on slave channel 1.	Channel operation is stopped.

Table 18.211 Operating Procedure for Non-Complementary Modulation Output Function Type 1 (2/2)

	Operation	TAUDn Status
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm of master and slave channels is set to 1 and the counter starts counting down.
During Operation	TAUDnCDRm, TAUDnTRO.TAUDnTROm, and TAUDnTMEm.TAUDnTMEm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master channel, slave channel 1 and slave channels 2 to 7 is loaded into TAUDnCNTm to perform counting down. When the counter of master channel reaches 0000_H: • INTTAUDnIm is generated. • TAUDnCDRm value is reloaded into TAUDnCNTm to continue counting down. • PWM output signals of slave channels 2 to 7 are set/reset. • TAUDnCDRm value of slave channel 1 is reloaded into TAUDnCNTm to perform counting down. • TAUDnCDRm value of slave channels 2 to 7 is reloaded into TAUDnCNTm to perform counting down. When the counter of slave channel 1 or slave channels 2 to 7 reaches 0000_H: • INTTAUDnIm is generated. • The TAUDnTRO.TAUDnTROm value for slave channels 2 to 7 is reflected in the TAUDTTOUTm output. When the counter of slave channels 2 to 7 reaches 0000_H: • INTTAUDnIm is generated. • PWM output signals of slave channels 2 to 7 are reset. TAUDTTOUTm of slave channels 2 to 7 outputs a PWM signal, a high-level signal or low-level signal depending on the values of real-time output bits (TAUDnTRO.TAUDnTROm) and modulation output bit (TAUDnTME.TAUDnTMEm) of a pair of slave channels.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(8) Specific Timing Diagrams

The following settings apply to the specific timing diagram.

- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

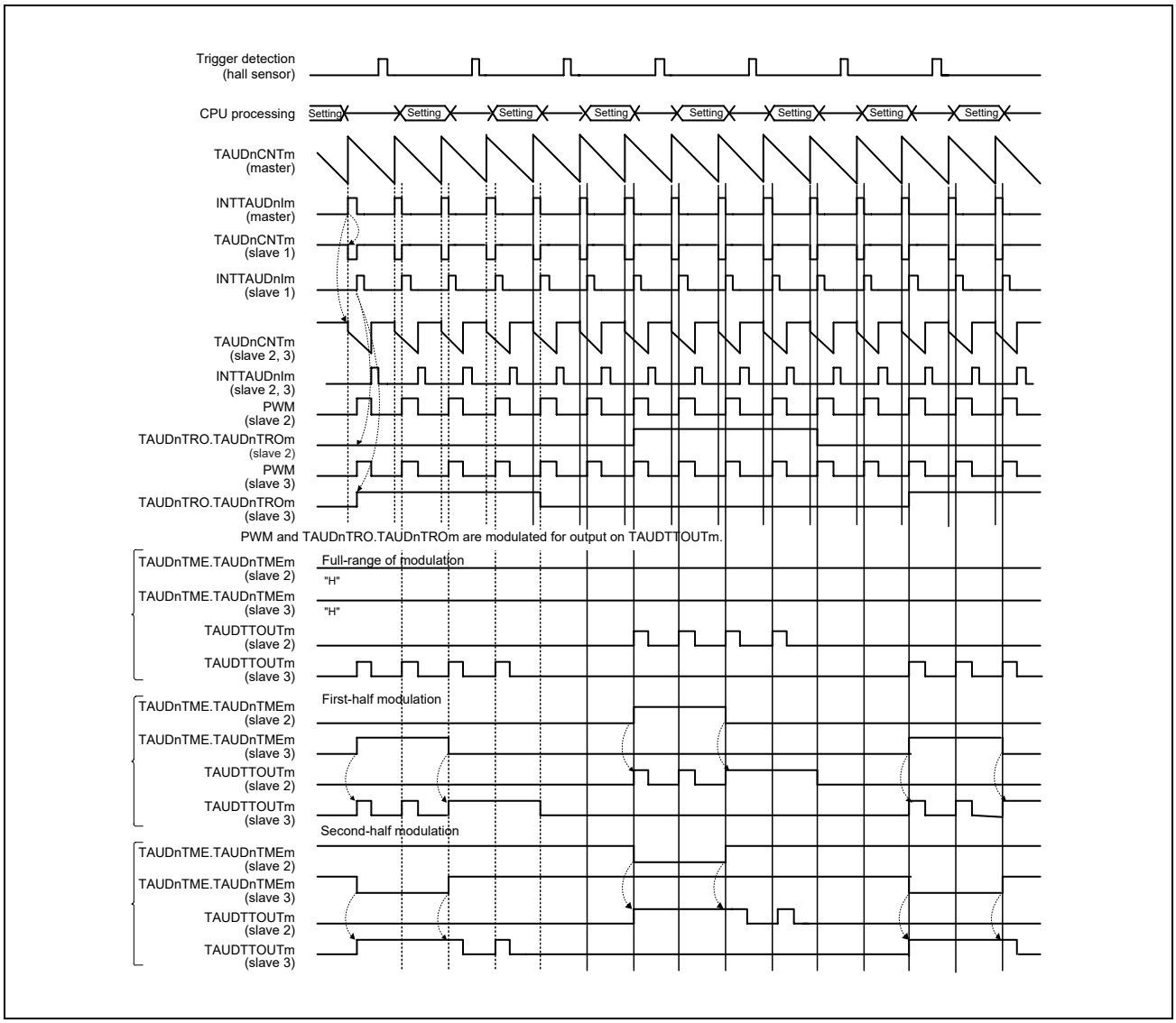


Figure 18.119 Specific Timing Diagram of Non-Complementary Modulation Output Function Type 1

The above timing diagram shows how full modulation, first-half modulation, and second-half modulation can be achieved by modifying the TAUDnTME.TAUDnTMEm bits of lower slave channels during operation.

The “Setting” symbol indicates a time period when the values of TAUDnCDRm, TAUDnTME.TAUDnTMEm, and TAUDnTRO.TAUDnTROm can be changed.

TAUDnTME.TAUDnTMEm setting is reflected by detecting the count start timing and master channel cycle. According to the modified setting, modulation waveforms are output from TAUDTTOUTm.

TAUDnTRO.TAUDnTROm bit value is set by software, but a new setting is applied only when an interrupt occurs on slave channel 1.

18.4.13.2 Non-Complementary Modulation Output Function Type 2

(1) Overview

Summary

This function outputs a PWM signal, a high-level signal, or low-level signal from TAUDTTOUTm depending on the real-time output bit value (TAUDnTRO.TAUDnTROm) and the modulation output enable bit value (TAUDnTME.TAUDnTMEm) of a pair of slave channels. Three pairs of channels are typically used.

Prerequisites

- It requires one master channel and seven slave channels.
- The operation mode of the master channel must be set to interval timer mode (see **Table 18.213, Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 2**).
- The operating mode for slave channel 1 should be set to event count mode (see **Table 18.217, Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2**).
- The operating mode for slave channels 2 to 7 should be set to up/down count mode (see **Table 18.220, Contents of TAUDnCMORm Register for slave channels 2 to 7 of Non-Complementary Modulation Output Function Type 2**).
- The output mode of the master channel must be set to independent channel output mode 1 (see **Section 18.4.4, Channel Output Modes**).
- This function does not use TAUDTTOUTm of slave channel 1 but TAUDnTRC.TAUDnTRCm should be set to 1 (see **Section 18.4.4, Channel Output Modes**).
- The channel output mode for slave channels 2 to 7 should be set to synchronous channel output mode 2 with non-complementary modulation output (see **Section 18.4.4, Channel Output Modes**).

Functional description

The master/slave channel counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The value of data register (TAUDnCDRm) is loaded into the counter (TAUDnCNTm).

- Master channel:
  The counter of master channel starts to count down. When the counter reaches 0000_H, INTTAUDnIm occurs.
- Slave channel 1:
  When slave channel 1 detects an interrupt from the master channel, the TAUDnCNTm value is decremented. When an interrupt from the master channel is detected for the (TAUDnCDRm + 1) times, INTTAUDnIm is generated. Then, the TAUDnCDRm value is loaded into TAUDnCNTm to continue operation subsequently. Since slave channel 1 is set as a real-time output trigger channel (TAUDnTRC.TAUDnTRCm = 1), the real-time output bit (TAUDnTRO.TAUDnTROm) of the channel which detects an interrupt on the corresponding channel is reflected to the respective TAUDTTOUTm outputs when an interrupt occurs on slave 1 channel.

- Slave channel 2:
  Once detecting an interrupt from the master channel, TAUDnCNTm counts in the reverse direction. When an interrupt is detected during count-up operation, TAUDnCDRm value is reloaded and then the counter starts to count down.
  If TAUDnCNTm = 0001_H, an interrupt occurs and a PWM output signal is set/reset.

The combined use of the master channel and slave channel 2 generates a PWM output signal. The master channel generates a PWM output cycle and slave channel 2 generate a duty cycle.

Slave channels 3 to 7 operates like slave channel 2.

A signal that is output from TAUDTTOUTm depends on a real-time output bit value (TAUDnTRO.TAUDnTROm) and a modulation output bit value (TAUDnTME.TAUDnTMEm) of the slave channel, as described in **Table 18.212, TAUDTTOUTm Output of a Pair of Slave Channels in Non-Complementary Modulation Output Function Type 2 (TAUDnTOL.TAUDnTOLm = 0)**.

This function cannot make a forced restart. The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

- If TAUDnTME.TAUDnTMEm = 0 on slave channels (TAUDnTOL.TAUDnTOLm = 0):
  - If the channel's TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm outputs a high-level signal.
  - If the channel's TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm outputs a low-level signal.
- If TAUDnTME.TAUDnTMEm = 1 on slave channels (TAUDnTOL.TAUDnTOLm = 0):
  - If the channel's TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm outputs PWM corresponding to the channel.
  - If the channel's TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm outputs a low-level signal.
- If TAUDnTOL.TAUDnTOLm is set to 1, high-level and low-level signals output from TAUDTTOUTm are inverted. The PWM signal becomes inverted logic. Only the initial setting of TAUDnTOL.TAUDnTOLm is permitted (cannot be changed during operation).

Table 18.212 TAUDTTOUTm Output of a Pair of Slave Channels in Non-Complementary Modulation Output Function Type 2 (TAUDnTOL.TAUDnTOLm = 0)

TAUDnTME.TAUDnTMEm	TAUDnTRO.TAUDnTROm	TAUDTTOUTm Output
0	0	Low level
	1	High level
1	0	Low level
	1	PWM (positive logic)

- This function enables simultaneous rewrite. See **Section 18.4.3, Simultaneous Rewrite**.
- If TAUDnTOL.TAUDnTOLm is set to 0 on slave channels 2 to 7, TAUDnTO.TAUDnTOm is set low before TAUDnTE.TAUDnTEm is set to 0.
- If TAUDnTOL.TAUDnTOLm is set to 1 on slave channels 2 to 7, TAUDnTO.TAUDnTOm is set high before TAUDnTE.TAUDnTEm is set to 0.

(2) Equations

Slave channels 2 to 7:

Carrier cycle (down/up) = [TAUDnCDRm (master) + 1] × 2 × count clock cycle

Duty time = [TAUDnCDRm (master) + 1 – TAUDnCDRm (slave)] × 2 × count clock cycle

(3) Block Diagram and General Timing Diagram

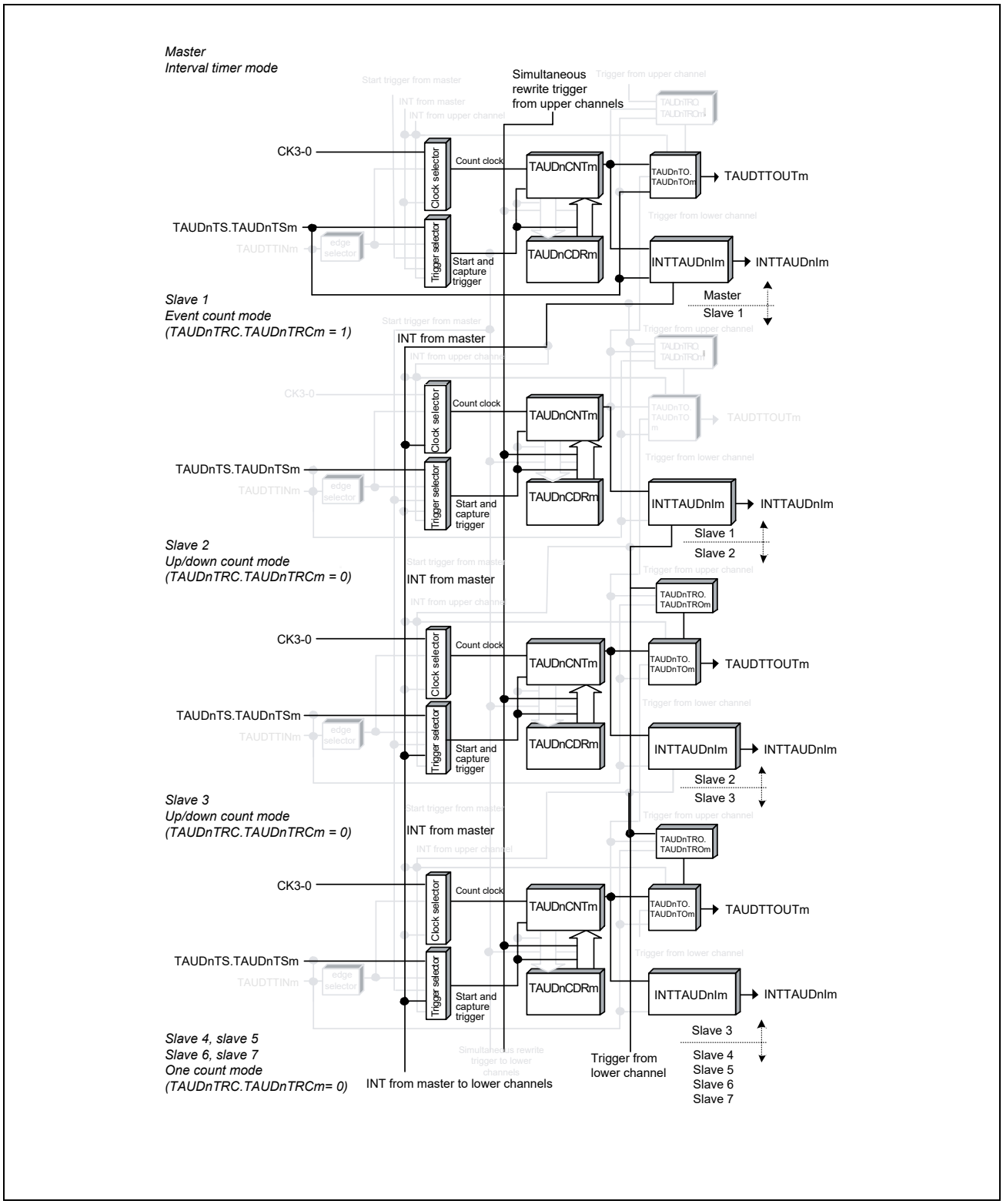


Figure 18.120 Block Diagram of Non-Complementary Modulation Output Function Type 2

The following settings apply to the general timing diagram.

- Master channel: INTTAUDnIm not generated at the beginning of operation.(TAUDnCMORm.TAUDnMD0 = 0)
- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

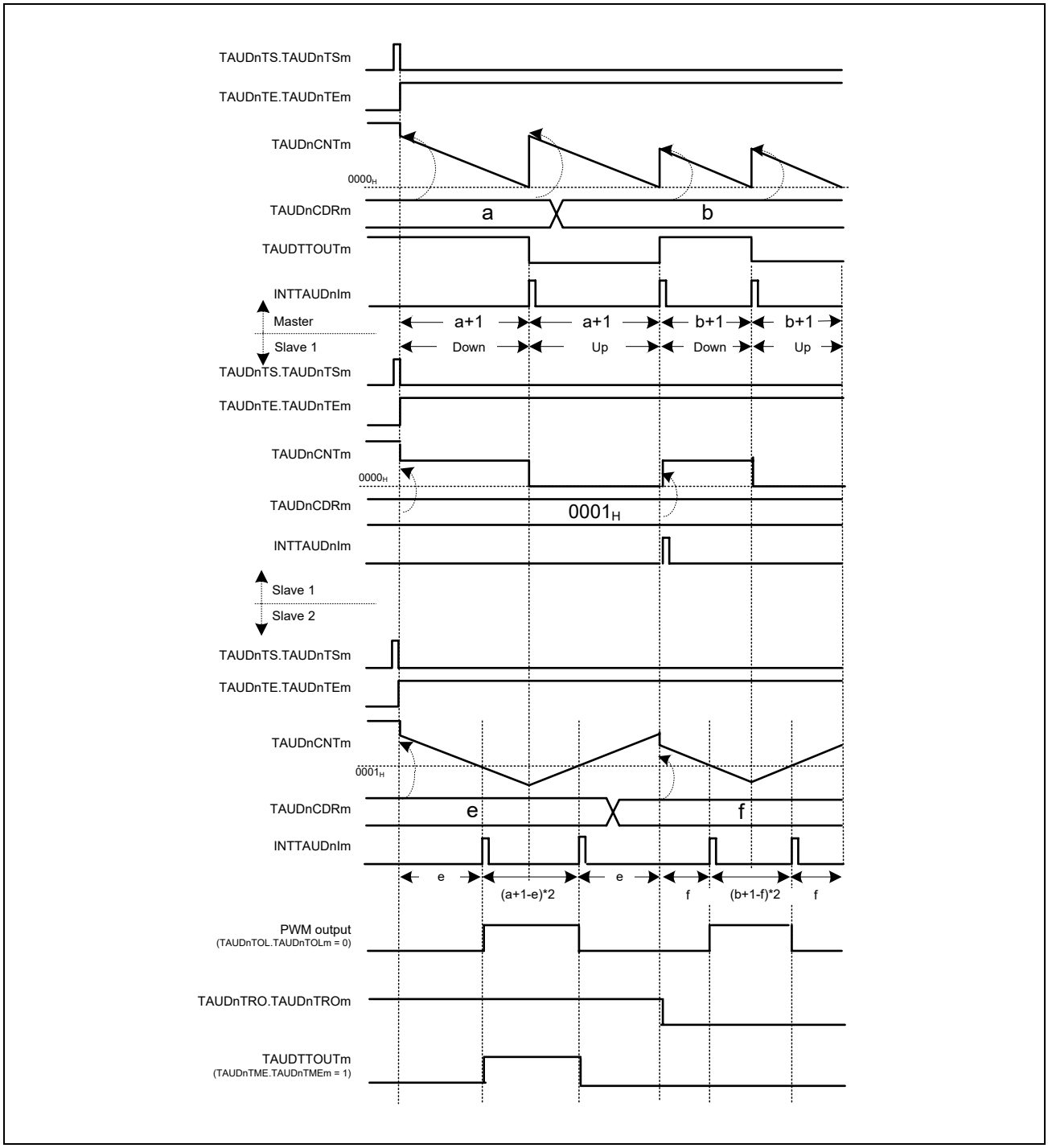


Figure 18.121 General Timing Diagram of Non-Complementary Modulation Output Function Type 2

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.213 Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time. 1: INTTAUDnIm is generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.214 Contents of TAUDnCMURm Register for Master Channels of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

Table 18.215 Control Bit Settings for Master Channels in Non-Complementary Modulation Output Function Type 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (toggle mode with TAUDnTOM.TAUDnTOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TAUDnTREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.216 Simultaneous Rewrite Settings for Master Channels of Non-Complementary Modulation Output Function Type 2

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: A simultaneous rewrite trigger signal is generated when master channel starts to count and the corresponding slave channel is at the peak of a triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

(5) Register Settings for Slave Channel 1

(a) TAUDnCMORm for slave channel 1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.217 Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	11: INTTAUDnIm of the master channel is used as the count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software. 011: Triggers simultaneous rewrite.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0011: Event count mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for slave channel 1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.218 Contents of TAUDnCMURm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used on slave channel 1 with this function.

CAUTION

TAUDnTRC.TAUDnTRCm should be set to 1 because slave channel 1 is used as a real-time output trigger channel.

(d) Simultaneous rewrite for slave channel 1

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.219 Simultaneous Rewrite Settings for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(6) Register settings for slave channels 2 to 7

(a) TAUDnCMORm for slave channels 2 to 7

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.220 Contents of TAUDnCMORm Register for slave channels 2 to 7 of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	111: The up/down output trigger signal of the master channel
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1001: Up/down count mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for slave channels 2 to 7

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.221 Contents of TAUDnCMURm Register for slave channels 2 to 7 of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Output mode for slave channels 2 to 7

Table 18.222 Control Bit Settings for Slave Channels 2 to 7 in Synchronous Channel Output Mode 2 with Non-Complementary Modulation Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: The upper channel generates the real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation 1: Enables modulation

(d) Simultaneous rewrite for slave channels 2 to 7

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.223 Simultaneous Rewrite Settings for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 2

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(7) Operating Procedure for Non-Complementary Modulation Output Function Type 2

Table 18.224 Operating Procedure for Non-Complementary Modulation Output Function Type 2 (1/2)

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 1: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channel 1**. Slave channels 2 to 7: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (6), Register settings for slave channels 2 to 7**. Set the value of TAUDnCDRm register of every channel. Set pulse cycle in TAUDnCDRm of master channel, and in TAUDnCDRm of slave channel 1, set the number of interrupts from master channel to be ignored before slave channel 1 generates an input signal. Set duty width in TAUDnCDRm of slave channels 2 to 7. Set TAUDnTRC.TAUDnTRCm to 1 on slave channel 1.	Channel operation is stopped.

Table 18.224 Operating Procedure for Non-Complementary Modulation Output Function Type 2 (2/2)

	Operation	TAUDn Status
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm of master and slave channels is set to 1 and the counter starts counting down.
Start Operation	TAUDnCDRm, TAUDnTRO.TAUDnTROm, and TAUDnTMEm.TAUDnTMEm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master channel and slave channels 2 to 7 is loaded into TAUDnCNTm to perform count down. The TAUDnCDRm value of slave channel 1 is loaded and the counter waits for an interrupt from the master channel. When the counter of master channel reaches 0000_H: • INTTAUDnIm is generated. • TAUDnCDRm value is reloaded into TAUDnCNTm to continue counting down. • The TAUDnCNTm value of slave channel 1 decrements by 1 and the counter waits for a next interrupt from the master channel. • TAUDnCNTm of slave channels 2 to 7 performs counting in opposite direction. • When the counter of slave channel 1 reaches 0000_H, it waits for a next interrupt from the master channel. When an interrupt is detected: - INTTAUDnIm is generated. • When the counter of slave channels 2 to 7 reaches 0001_H: - INTTAUDnIm is generated. - PWM output signals of slave channels 2 to 7 are set/reset. TAUDTTOUTm of slave channels 2 to 7 outputs a PWM signal, a high-level signal or low-level signal depending on the values of real-time output bits (TAUDnTRO.TAUDnTROm) and modulation output bit (TAUDnTME.TAUDnTMEm) of a pair of slave channels.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(8) Specific Timing Diagrams

The following settings apply to the general timing diagram.

- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

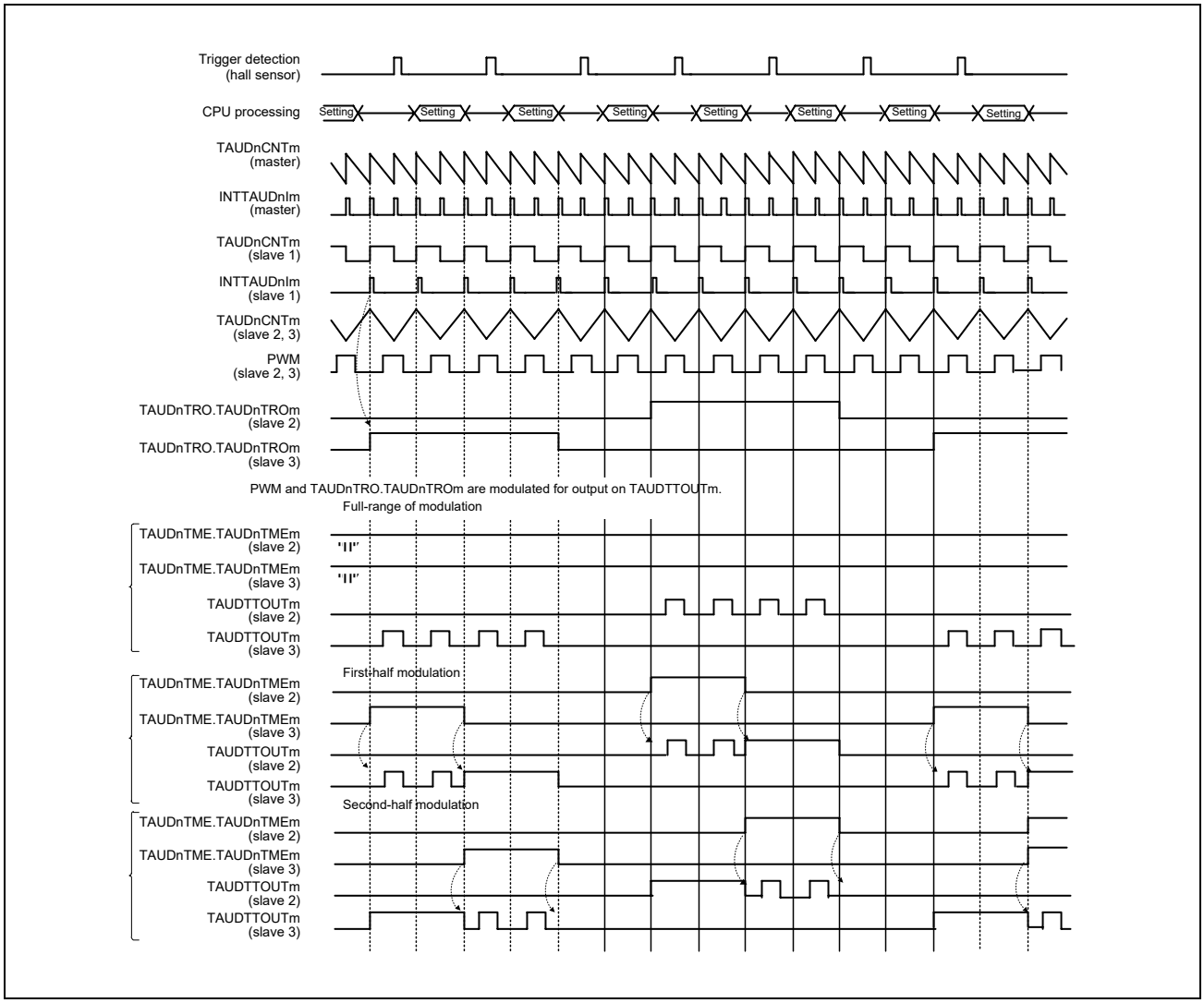


Figure 18.122 Specific Timing Diagram of Non-Complementary Modulation Output Function Type 2

The above timing diagram shows how full modulation, first-half modulation, and second-half modulation can be achieved by modifying the TAUDnTME.TAUDnTMEm bits of lower slave channels during operation.

The “Setting” symbol indicates a time period when the values of TAUDnCDRm, TAUDnTME.TAUDnTMEm, and TAUDnTRO.TAUDnTROm can be changed.

TAUDnTME.TAUDnTMEm setting is reflected by detecting the count start timing and triangle PWM carrier cycle (peak interrupt timing).

TAUDnTRO.TAUDnTROm bit value is set by software, but a new setting is applied only when an interrupt occurs on slave channel 1.

18.4.13.3 Complementary Modulation Output Function

(1) Overview

Summary

This function outputs a triangle PWM output signal, a high-level signal, or low-level signal from TAUDTTOUTm depending on the real-time output bit value (TAUDnTRO.TAUDnTROm) and the modulation output bit value (TAUDnTME.TAUDnTMEm) of a pair of slave channels. Three pairs of channels are typically used.

Prerequisites

- It requires one master channel and seven slave channels.
- The operation mode of the master channel must be set to interval timer mode (see **Table 18.227, Contents of TAUDnCMORm Register for Master Channels of Complementary Modulation Output Function**).
- The operating mode for slave channel 1 should be set to event count mode (see **Table 18.231, Contents of TAUDnCMORm Register for Slave Channel 1 of Complementary Modulation Output Function**).
- The operating mode for slave channels 2, 4 and 6 should be set to up/down count mode (see **Table 18.234, Contents of TAUDnCMORm Register for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function**).
- The operating mode for slave channels 3, 5 and 7 should be set to one-count mode (see **Table 18.238, Contents of TAUDnCMORm Register for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function**).
- The output mode for master channels should be set to independent channel output mode 1 (see **Section 18.4.4, Channel Output Modes**).
- This function does not use TAUDTTOUTm of slave channel 1 but TAUDnTRC.TAUDnTRCm should be set to 1 (see **Section 18.4.4, Channel Output Modes**).
- The channel output mode for slave channels 2 to 7 should be set to synchronous channel output mode 2 with complementary modulation output (see **Section 18.4.4, Channel Output Modes**).

Functional description

- Master channel:
  The counter of master channel is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The value of data register (TAUDnCDRm) of master channel is loaded into the counter (TAUDnCNTm) and the counter starts to count down from this value.
  When the counter of master channel reaches 0000_H, INTTAUDnIm occurs. This decrements the counter value of slave channel 1 by 1 and the counter of slave channel 2 starts to count in the opposite direction.
- Slave channel 1:
  When the counter reaches 0000_H, slave channel 1 waits for the next interrupt from the master channel. And the TAUDnCDRm value is reloaded into TAUDnCNTm (slave 1) and INTTAUDnIm is generated.
  Slave channel 1 is set as a real-time output trigger channel (TAUDnTRC.TAUDnTRCm = 1). The value of real-time output bit (TAUDnTRO.TAUDnTROm) of each channel is applied to the channel that detects the occurrence of an interrupt on slave channel 1. The real-time output bit value can be changed in any timing by application software but a new value is not applied until an interrupt occurs on slave channel 1.

- Slave channel 2:
  When the slave channel 2 counter reaches 0001_H, the slave channel 3 counter starts counting down. When the slave channel 3 counter reaches 0000_H, an interrupt occurs.
- Slave channels 2 and 3:
  The combined use of the master channel and slave channels 2 and 3 generates a PWM output signal. The master channel generates a PWM output cycle, slave channel 2 generates a duty cycle, and slave channel 3 generates dead time.
- Slave channels 4 to 7:
  Slave channels 4 and 6 operate like slave channel 2. Slave channels 5 and 7 operate like slave channel 3.

A signal that is output from TAUDTTOUTm depends on a real-time output bit value (TAUDnTRO.TAUDnTROm), a modulation output bit value (TAUDnTME.TAUDnTMEm), and an output level bit value (TAUDnTDL.TAUDnTDLm) of the slave channel, as described in **Table 18.225 TAUDTTOUTm Output (TAUDnTOL.TAUDnTOLm = 0) for a Pair of Slave Channels of Complementary Modulation Output Function**.

It is, however, prohibited that a high-level signal is output from both channel 2 and channel 3 (in order to prevent a motor driver short circuit).

Forced restart is not possible for this function. The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

- If TAUDnTME.TAUDnTMEm of a pair of channels is set to 1 (TAUDnTOL.TAUDnTOLm = 0):
  - If TAUDnTRO.TAUDnTROm of one channel is set to 1, TAUDTTOUTm outputs the corresponding PWM of the channel.
  - If TAUDnTRO.TAUDnTROm of both channels is set to 0, a pair of TAUDTTOUTm outputs a low-level signal.
- If TAUDnTME.TAUDnTMEm of a pair of channels is set to 0 (TAUDnTOL.TAUDnTOLm = 0):
  - If TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm of the channel outputs a high-level signal.
  - If TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm of the channel outputs a low-level signal.
- If TAUDnTOL.TAUDnTOLm is set to 1, high-level and low-level signals output from TAUDTTOUTm are inverted. No PWM signals are changed depending on the setting of TAUDnTOL.TAUDnTOLm. The PWM signal is negative logic.

Table 18.225 TAUDTTOUTm Output (TAUDnTOL.TAUDnTOLm = 0) for a Pair of Slave Channels of Complementary Modulation Output Function

TAUDnTME. TAUDnTME2	TAUDnTME. TAUDnTME3	TAUDnTRO. TAUDnTRO2	TAUDnTRO. TAUDnTRO3	TAUDnTDL. TAUDnTDL2	TAUDnTDL. TAUDnTDL3	TAUDTTOUT2 Output	TAUDTTOUT3 Output
0	0	0	0	—	—	Low level	Low level
		0	1	1	0	Low level	High level
		1	0	0	1	High level	Low level
		1	1	—	—	Setting prohibited	Setting prohibited
1	1	0	0	—	—	Low level	Low level
		0	1	1	0	~PWMm	PWMm
		1	0	0	1	PWMm	~PWMm
		1	1	—	—	Setting prohibited	Setting prohibited

NOTES

1. In the above table, PWM indicates a positive PWM signal and ~PWM indicates an inverted PWM signal (positive logic). PWM and ~PWM are set by TAUDnTDL.TAUDnTDLm.
2. Any settings not listed above are prohibited.

- If TAUDnTME.TAUDnTMEm is continuously set to 1 while TAUDnTRO.TAUDnTROm of one of paired channels is set to 1, full modulation is applied.
- If TAUDnTME.TAUDnTMEm is set to 1 at the first half of the period while TAUDnTRO.TAUDnTROm of one of paired channels is set to 1, first-half modulation is applied.
- If TAUDnTME.TAUDnTMEm is set to 1 at the second half of the period while TAUDnTRO.TAUDnTROm of one of paired channels is set to 1, second-half modulation is applied.
- Whether dead time is added to a normal or reverse phase PWM signal when two channels become high-level signal outputs simultaneously depends on a TAUDnTDL.TAUDnTDLm bit value.
  - If TAUDnTDL.TAUDnTDLm = 0, dead time is added to a normal phase PWM signal.
  - If TAUDnTDL.TAUDnTDLm = 1, dead time is added to a reverse phase PWM signal.
  - The operation defined by a TAUDnTDL.TAUDnTDLm bit value should be conducted by application software during operation. To modify TAUDnTDL.TAUDnTDLm, rewrite it during the period when TAUDnTRO.TAUDnTROm is 00_B.
- The TAUDnCDRm value of slave channel 1 should be set to the value to generate INTTAUDnIm of slave channel 1 at a carrier cycle (peak interrupt timing).
- If TAUDnTOL.TAUDnTOLm is set to 0 on slave channels 2 to 7:
  - If TAUDnTDL.TAUDnTDLm is set to 0, TAUDnTO.TAUDnTOm is set to 0 (low) before TAUDnTE.TAUDnTEm is set to 0.
  - If TAUDnTDL.TAUDnTDLm is set to 1, TAUDnTO.TAUDnTOm is set to 1 (high) before TAUDnTE.TAUDnTEm is set to 0.

- If TAUDnTOL.TAUDnTOLm is set to 1 on slave channels 2 to 7:
  - If TAUDnTDL.TAUDnTDLm is set to 0, TAUDnTO.TAUDnTOm is set to 1 (high) before TAUDnTE.TAUDnTEm is set to 0.
  - If TAUDnTDL.TAUDnTDLm is set to 1, TAUDnTO.TAUDnTOm is set to 0 (low) before TAUDnTE.TAUDnTEm is set to 0.
  - This function enables simultaneous rewrite. See **Section 18.4.3, Simultaneous Rewrite**.

Table 18.226 TAUDnTDL.TAUDnTDLm Settings (TAUDnTOL.TAUDnTOLm = 0) for a Pair of Slave Channels of Complementary Modulation Output Function

TAUDnTME. TAUDnTME2	TAUDnTME. TAUDnTME3	TAUDnTRO. TAUDnTRO2	TAUDnTRO. TAUDnTRO3	TAUDnTDL. TAUDnTDL2	TAUDnTDL. TAUDnTDL3
0	0	0	0	1	1
		0	1	1	0
		1	0	0	1
1	1	0	0	1	1
		0	1	1	0
		1	0	0	1

- The value of TAUDnCDRm of slave channel 1 should be set to 1 so that INTTAUDnIm is generated on slave channel 1 at the peak of a carrier cycle.
- Set TAUDnCMORm.TAUDnMD0 of master channel to 0.
- This function enables simultaneous rewrite. See **Section 18.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

$0000_H \leq$ TAUDnCDRm (master) $< FFFF_H$

Carrier cycle (down/up) = (TAUDnCDRm (master) + 1) × 2 × count clock cycle

For slave channels 2 and 3:

PWM signal width (normal phase) = [(TAUDnCDRm (master) + 1 – TAUDnCDRm (slave 2) × 2) – (TAUDnCDRm (slave 3) + 1)] × count clock cycle

PWM signal width (reverse phase) = [(TAUDnCDRm (master) + 1 –TAUDnCDRm (slave 2) × 2) + (TAUDnCDRm (slave 3) + 1)] × count clock cycle

For slave channels 4 to 7:

Slave channels 4 and 6 are calculated in the same way as slave channel 2, whereas slave channels 5 and 7 are calculated in the same way as slave channel 3.

(3) Block Diagram and General Timing Diagram

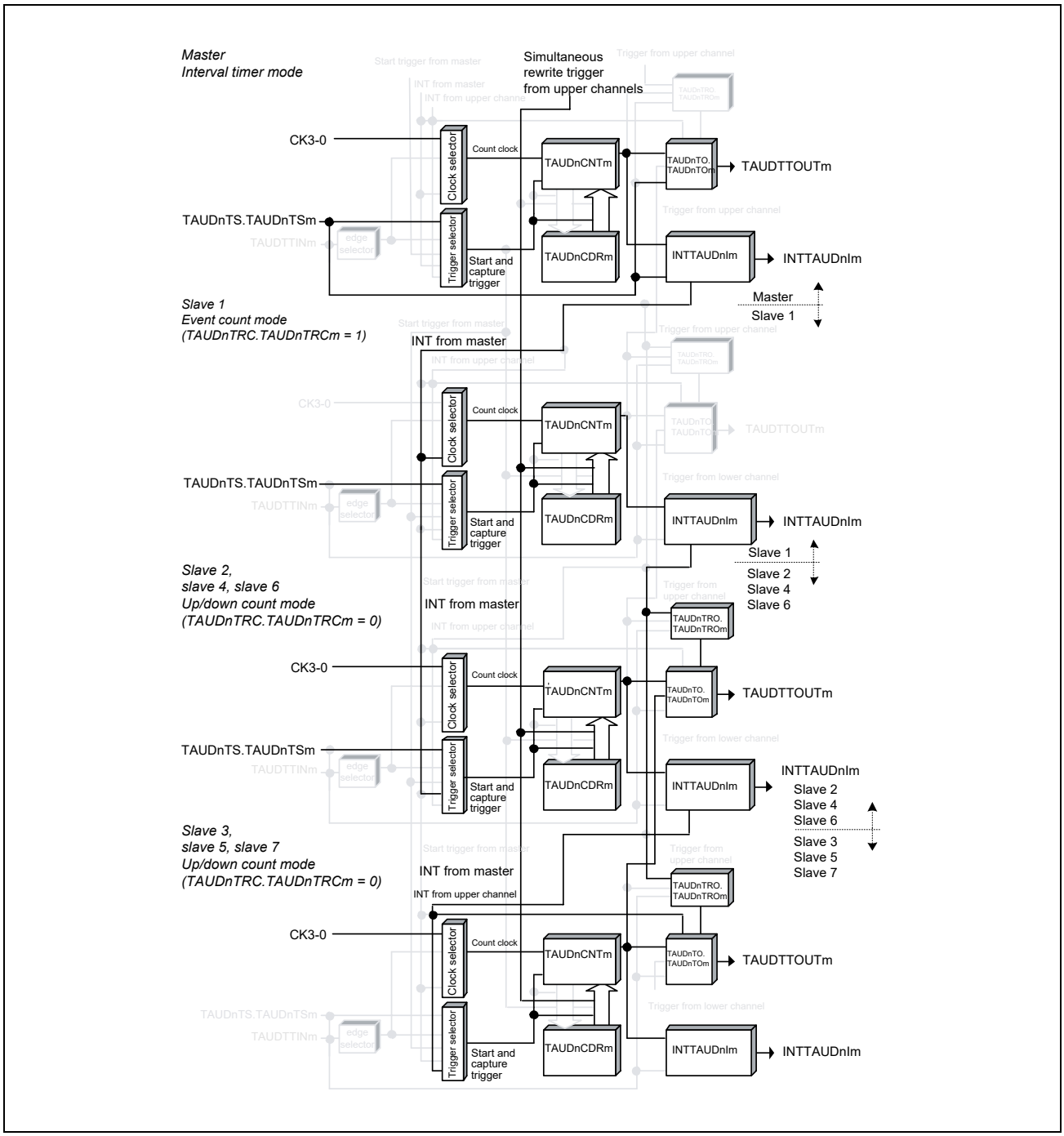


Figure 18.123 Block Diagram of Complementary Modulation Output Function

The following settings apply to the general timing diagram.

- Master channel: INTTAUDnIm not generated at the beginning of operation.(TAUDnCMORm.TAUDnMD0 = 0)
- Slave channels 1: TAUDnCDRm = 0001_H
- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

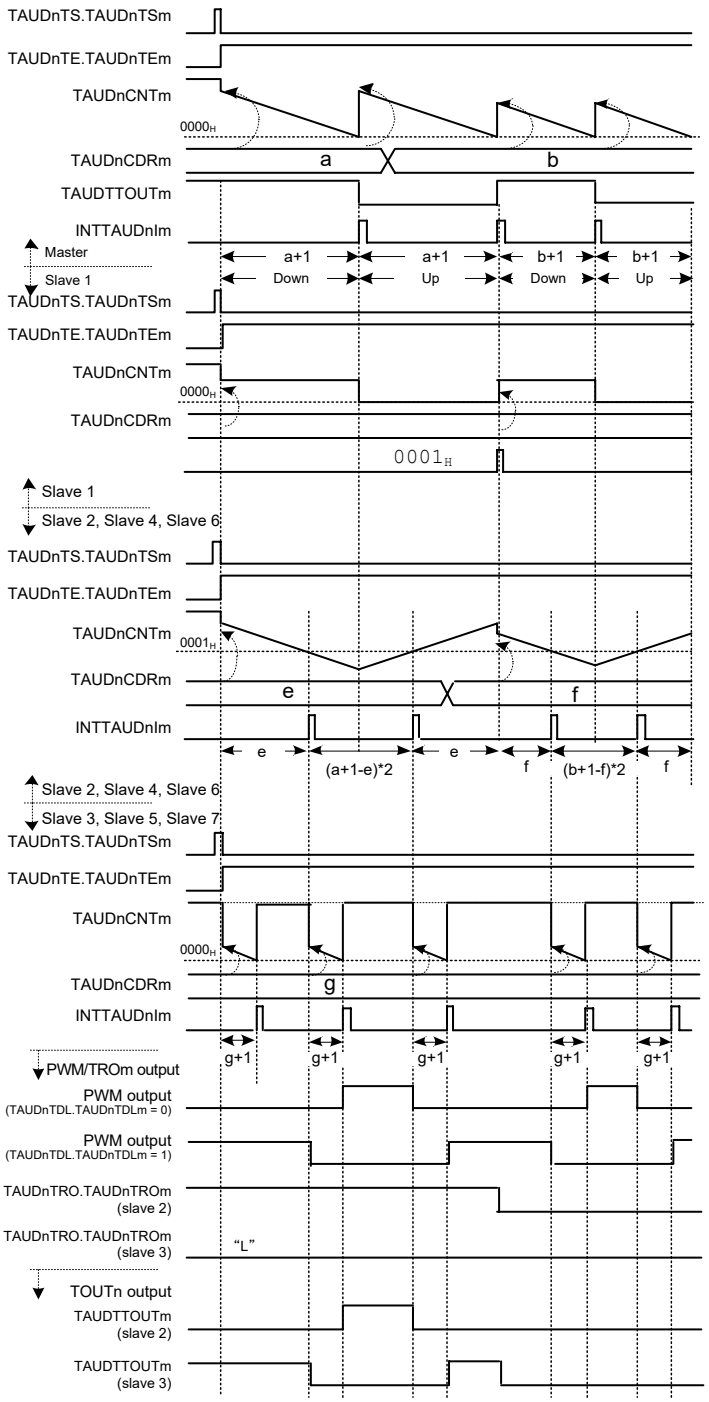


Figure 18.124 General Timing Diagram of Complementary Modulation Output Function

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.227 Contents of TAUDnCMORm Register for Master Channels of Complementary Modulation Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated and TAUDTTOUTm is not toggled at the beginning of operation or at a restart time. 1: INTTAUDnIm is generated to toggle TAUDTTOUTm at the beginning of an operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.228 Contents of TAUDnCMURm Register for Master Channels of Complementary Modulation Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode

Table 18.229 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.230 Simultaneous Rewrite Settings for Master Channels of Complementary Modulation Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

(5) Register Settings for Slave Channel 1

(a) TAUDnCMORm for slave channel 1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.231 Contents of TAUDnCMORm Register for Slave Channel 1 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	11: INTTAUDnIm of the master channel is used as the count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software. 011: Triggers simultaneous rewrite.
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0011: Event count mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for slave channel 1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.232 Contents of TAUDnCMURm Register for Slave Channel 1 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used on slave channel 1 with this function.

CAUTION

TAUDnTRC.TAUDnTRCm should be set to 1 because slave channel 1 is used as a real-time output trigger channel.

(d) Simultaneous rewrite for slave channel 1

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.233 Simultaneous Rewrite Settings for Slave Channel 1 of Complementary Modulation Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(6) Register settings for slave channels 2, 4, and 6

(a) TAUDnCMORm for slave channels 2, 4, and 6

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.234 Contents of TAUDnCMORm Register for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	111: Up/down output trigger signal of master channel
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1001: Up/down count mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for slave channels 2, 4, and 6

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.235 Contents of TAUDnCMURm Register for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Output mode for slave channels 2, 4, and 6

Table 18.236 Control Bit Settings in Synchronous Channel Output Mode 2 with Complementary Modulation Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	0: Adds dead time if an interrupt is detected on an even upper channel and the conditions set by TAUDnTDL.TAUDnTDLm are satisfied.
TAUDnTDL.TAUDnTDLm	0: Adds dead time to normal phase. 1: Adds dead time to reverse phase.
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation 1: Enables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm exclusively from odd-numbered channels.

(d) Simultaneous rewrite for slave channels 2, 4, and 6

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.237 Simultaneous Rewrite Settings for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: A simultaneous rewrite trigger signal is generated when master channel starts to count and the corresponding slave channel is at the peak of a triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(7) Register settings for slave channels 3, 5, and 7

(a) TAUDnCMORm for slave channels 3, 5, and 7

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.238 Contents of TAUDnCMORm Register for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS[1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of TAUDnCKS[1:0] bits must be the same for the master channel and the slave channel.
13, 12	TAUDnCCS[1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	110: Dead time trigger
7, 6	TAUDnCOS[1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Enables start trigger detection while counting.

(b) UDnCMURm for slave channels 3, 5, and 7

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.239 Contents of TAUDnCMURm Register for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Output mode for slave channels 3, 5, and 7

Table 18.240 Control Bit Settings in Synchronous Channel Output Mode 2 with Complementary Modulation Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	0: Adds dead time if an interrupt is detected on an even upper channel and the conditions set by TAUDnTDL.TAUDnTDLm are satisfied.
TAUDnTDL.TAUDnTDLm	0: Adds dead time to normal phase. 1: Adds dead time to reverse phase.
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation 1: Enables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm exclusively from even-numbered channels.

(d) Simultaneous rewrite for slave channels 3, 5, and 7

Both master and slave channels should have the same simultaneous rewrite settings.

Table 18.241 Simultaneous Rewrite Settings for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(8) Operating Procedure for Complementary Modulation Output Function

Table 18.242 Operating Procedure for Complementary Modulation Output Function (1/2)

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 1: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channel 1**. Slave channels 2, 4, and 6: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (6), Register settings for slave channels 2, 4, and 6**. Slave channels 3, 5, and 7: Set TAUDnCMORm/TAUDnCMURm register and the channel output mode as described in **Section (7), Register settings for slave channels 3, 5, and 7**. Set the value of TAUDnCDRm register of every channel. Set a pulse cycle using TAUDnCDRm of master channel, and an interrupt count of master channel ignored using TAUDnCDRm of slave channel 1. Also set a duty width in TAUDnCDRm of slave channels 2, 4, and 6, and a dead time delay on slave channels 3, 5, and 7. Set TAUDnTRC.TAUDnTRCm to 1 on slave channel 1.	Channel operation is stopped.

Table 18.242 Operating Procedure for Complementary Modulation Output Function (2/2)

	Operation	TAUDn Status
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm of master and slave channels is set to 1 and the counter starts counting down.
During Operation	TAUDnCDRm, TAUDnTRO.TAUDnTROm, TAUDnTME.TAUDnTMEm, and TAUDnTDL.TAUDnTDLm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master channel and slave channels 2 to 7 is loaded into TAUDnCNTm to perform count down. TAUDnCDRm value of slave channel 1 is loaded and the counter waits for a master channel interrupt. When the counter of master channel reaches 0000_H: • INTTAUDnIm is generated. • TAUDnCDRm value is reloaded into TAUDnCNTm to continue counting down. • TAUDnCNTm value of slave channel 1 decreases by 1 and the counter waits for the next master channel interrupt. • TAUDnCNTm of slave channels 2, 4, and 6 performs counting in the reverse direction. • The TAUDnTME.TAUDnTMEm value of slave channels 2 to 7 is reflected in TAUDTTOUTm output when the TAUDnCDRm value of slave channels 2, 4, and 6 is loaded. • The counter of slave channel 1 waits for the next interrupt from the master channel when reaching 0000_H. When the interrupt is detected: – TAUDnCDRm value is reloaded into TAUDnCNTm and the counter waits for the next master channel interrupt. – INTTAUDnIm is generated. – TAUDnTRO.TAUDnTROm is changeable. When the counter of slave channels 2, 4, and 6 reaches 0001_H: • INTTAUDnIm is generated. • PWM output of slave channel m is set/reset (when the specified condition of the channel output mode is met). • TAUDnCDRm value of slave channels 3, 5, and 7 is loaded into TAUDnCNTm to perform count down. When the counter of slave channels 3, 5, and 7 reaches 0000_H: • INTTAUDnIm is generated. • PWM output of slave channel m is set/reset (when the specified condition of the channel output mode is met). TAUDTTOUTm of slave channels 2 to 7 outputs a PWM signal, a high-level signal or low-level signal depending on the values of real-time output bit (TAUDnTRO.TAUDnTROm), modulation output bit (TAUDnTME.TAUDnTMEm), and output level bit (TAUDnTDL.TAUDnTDLm) of a pair of slave channels.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(9) Specific Timing Diagrams

The following settings apply to the timing diagram.

- Master channel: INTTAUDnIm is not generated at the beginning of operation.(TAUDnCMORm.TAUDnMD0 = 0)
- Slave channel 1: TAUDnCDRm = 0001_H
- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

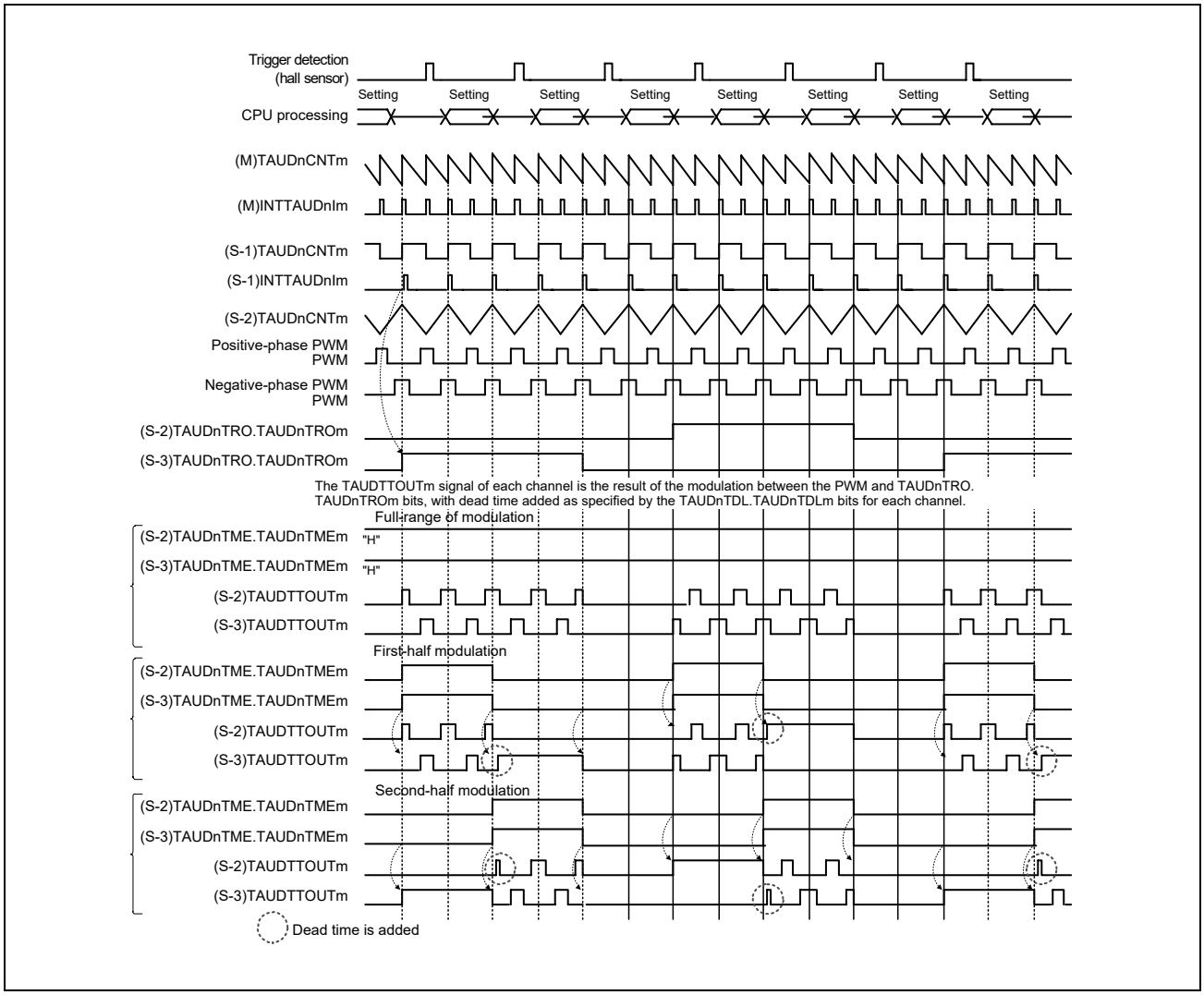


Figure 18.125 Specific Timing Diagram of Complementary Modulation Output Function

The above timing diagram shows how full modulation, first-half modulation, and second-half modulation can be achieved by modifying the TAUDnTME.TAUDnTMEm bits of lower slave channels during operation.

A modulated PWM output signal and TAUDnTRO.TAUDnTROm bit value are output from slave channels 2 and 3.

TAUDnTME.TAUDnTMEm and TAUDnTDL.TAUDnTDLm settings are reflected by detecting the count start timing and triangle PWM carrier cycle (peak interrupt timing).

The value of the TAUDnTRO.TAUDnTROm bit is set by software, but a new setting is applied only when an interrupt occurs on slave channel 1.

NOTE

Dead time is added to suppress simultaneous change of PWM edges of normal and reverse phases.

The “Setting” symbol indicates a time period when the values of TAUDnCDRm, TAUDnTME.TAUDnTMEm, TAUDnTRO.TAUDnTROm, and TAUDnTDL.TAUDnTDLm can be changed.

Section 19 Timer Array Unit J (TAUJ)

This section contains a generic description of the Timer Array Unit J (TAUJ).

The first part of this section describes all RH850/C1M-A specific properties, such as units, register base addresses, input/output signal names, etc.

The remainder of the section describes the features that apply to all implementations.

19.1 Overview of RH850/C1M-A TAUJ

19.1.1 Units

This microcontroller has following number of units of the Timer Array Unit J.

Table 19.1 Units

Product Name	RH850/C1M-A2	RH850/C1M-A1
Number of units	2	1
Name	TAUJn (n = 0,1)	TAUJn (n = 0)

Table 19.2 Indices

Index	Meaning
n	Throughout this section, the individual TAUJ units are identified by the index “n” (n = 0, 1) (n = 0, 1 for RH850/C1M-A2; n = 0 for RH850/C1M-A1); for example, TAUJnTOM is the TAUJn channel output mode register.
m	The TAUJ has four channels. Throughout this section, the individual channels are identified by the index “m” (m = 0 to 3). A certain channel is denoted as CHm. The even numbered channels (m = 0, 2) are denoted as CHm_even. The odd numbered channels (m = 1, 3) are denoted as CHm_odd.

19.1.2 Register Base Addresses

TAUJ base addresses are listed in the following table.

TAUJ register addresses are given as offsets from the base addresses in general.

Table 19.3 Register Base Addresses

Base Address Name	Base Address
<TAUJ0_base>	FFE5 0000_H
<TAUJ1_base>	FF85 1000_H*1

Note 1. C1M-A1 does not have TAUJ1.

19.1.3 Clock Supply

TAUJ clock supply is listed in the following table.

Table 19.4 TAUJn Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
TAUJ0	PCLK	CLKC_HSB (non-modulated high-speed peripheral clock)
TAUJ1*1	PCLK	CLKC_HSB (non-modulated high-speed peripheral clock)

Note 1. C1M-A1 does not have TAUJ1.

19.1.4 Interrupts and DMA/DTS

TAUJ interrupt requests are listed in the following table.

Table 19.5 Interrupt Requests

Interrupt Name	Outline	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
TAUJ0						
INTTAUJ0I0	Channel 0 interrupt	90	15	—	15	—
INTTAUJ0I1	Channel 1 interrupt	91	16	—	16	—
INTTAUJ0I2	Channel 2 interrupt	92	17	—	17	—
INTTAUJ0I3	Channel 3 interrupt	93	18	—	18	—
TAUJ1*2						
INTTAUJ10	Channel 0 interrupt	94	—	—	—	—
INTTAUJ11	Channel 1 interrupt	95	—	—	—	—
INTTAUJ12	Channel 2 interrupt	96	—	—	—	—
INTTAUJ13	Channel 3 interrupt	97	—	—	—	—

—: No number is assigned.

Note 1. 1st: Primary channel, 2nd: Secondary channel

Note 2. C1M-A1 does not have TAUJ1.

19.1.5 Reset Sources

TAUJ reset sources are listed in the following table. TAUJ is initialized by these reset sources.

Table 19.6 Reset Sources

Unit Name	Reset Source
TAUJ0	Reset by any reset source
TAUJ1*1	Reset by any reset source

Note 1. C1M-A1 does not have TAUJ1.

19.1.6 External Input/Output Signals

External input/output signals of TAUJ are listed in the following table.

Table 19.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal Name
TAUJ0		
TAUJTTIN0-TAUJTTIN3	Channel 0 to 3 input	TAUJ0I0-TAUJ0I3
TAUJTTOUT0-TAUJTTOUT3	Channel 0 to 3 output	TAUJ0O0-TAUJ0O3
TAUJ1*1		
TAUJTTIN0-TAUJTTIN3	Channel 0 to 3 input	TAUJ1I0-TAUJ1I3
TAUJTTOUT0-TAUJTTOUT3	Channel 0 to 3 output	TAUJ1O0-TAUJ1O3

Note 1. C1M-A1 does not have TAUJ1.

19.2 Overview

19.2.1 Functional Overview

The TAUJ has the following functions:

- Independent channel operation function (operated using a single channel)
- Synchronous channel operation function (operated using a master channel and multiple slave channels)

The timer array unit J is used to perform various count or timer operations and to output a signal which depends on the result of the operation. It contains one prescaler block for count clock generation and 4 channels, each equipped with a 32-bit counter TAUJnCNTm and a 32-bit data register TAUJnCDRm to hold the count start value or compare value.

It also contains several control and status registers.

Independent and synchronous operation

Every channel can operate in two operating modes, either independently or in combination with other channels (synchronously), i.e. multiple channels depend on each other with one master and one or more slave channels.

When a channel is operated independently, it can be operated independent of all other channels.

The synchronous operation function is implemented by using a combination of channel groups (consisted of master and slave channels).

Several rules apply to the settings of channels.

19.2.1.1 Terms

In this section, the following terms are used.

Independent channel operation function / synchronous channel operation function

TAUJ consists of four channels which operate both independently (independent channel operation function) and in combination with other channels (synchronous channel operation function).

- The independent channel operation function can use any channel independent of all other channels.
- The synchronous channel operation function is implemented by using a combination of channel groups (comprised of master and slave channels).

Several rules apply to the settings of channels.

Channel group

In the synchronous channel operation function, all channels that depend on each other are referred to as a “channel group”. A channel group has one master channel and one or more slave channels.

Operating mode

An operating mode can be selected for every channel m. The operating mode defines the basic operation and features of a channel.

In synchronous channel operation, every channel in the channel group can operate in a different operating mode, such as capture mode and interval timer mode.

Channel output mode

The channel output mode defines the operation of TAUJTTOUTm

- of a single channel (independent output operation) or
- of all channels in a channel group (synchronous output operation).

The channel output mode includes independent channel output mode 1.

Channel operation function

The channel operation function defines the complete function and all features

- of a single channel (independent channel operation) or
- of all channels in a channel group (synchronous channel operation).

Upper/lower channel

Depending on the channel number m, a channel with a smaller channel number or larger channel number can be referred to as “upper” or “lower” channel:

- Upper channel: Channel with a smaller channel number
- Lower channel: Channel with a larger channel number

For instance, as to channel 2, channel 1 is an upper channel and channel 3 is a lower channel. Channel 0 is the highest channel and channel 3 is the lowest channel.

19.2.1.2 Operation Functions

TAUJ provides the following functions by operating individual channels independently or in combination.

Table 19.8 TAUJ Operation Functions

Operation Function	Setting Example Reference Section
Independent Channel Operation Functions	**19.4.9**
Interval timer function	**19.4.9.1**
TAUJTTINm input interval timer function	**19.4.9.2**
TAUJTTINm input pulse interval measurement function	**19.4.9.3**
TAUJTTINm input signal width measurement function	**19.4.9.4**
TAUJTTINm input position detection function	**19.4.9.5**
TAUJTTINm input period count detection function	**19.4.9.6**
Synchronous Channel Operation Functions	**19.4.10**
PWM output function	**19.4.10.1**

19.2.1.3 Input/Output and Interrupt Request Signals

The following diagram illustrates TAUJ input/output and interrupt request signals.

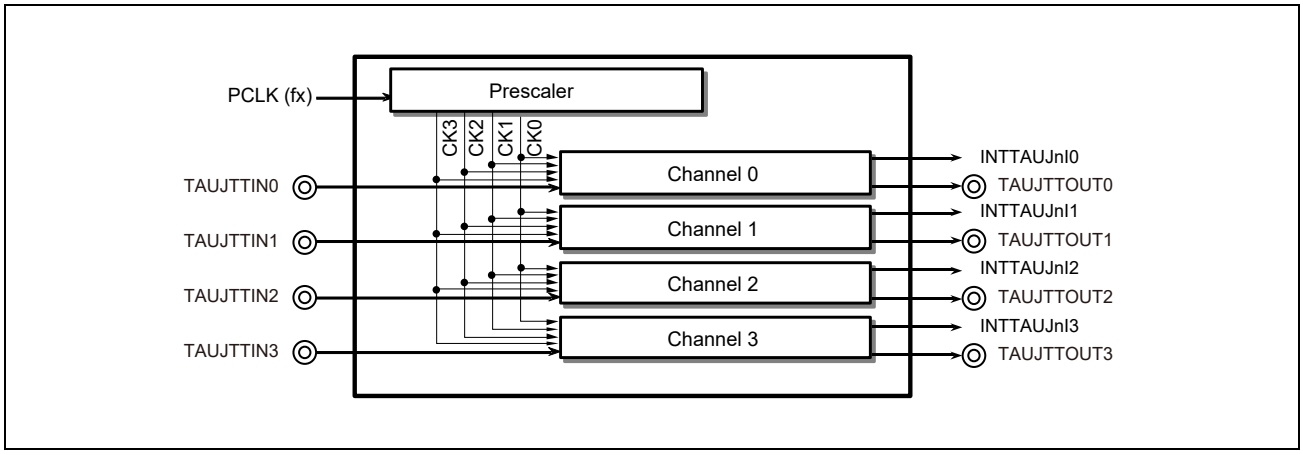


Figure 19.1 TAUJ Input/Output and Interrupt Request Signals

19.2.2 Block Diagram

The following figure shows the main components of the TAUJ:

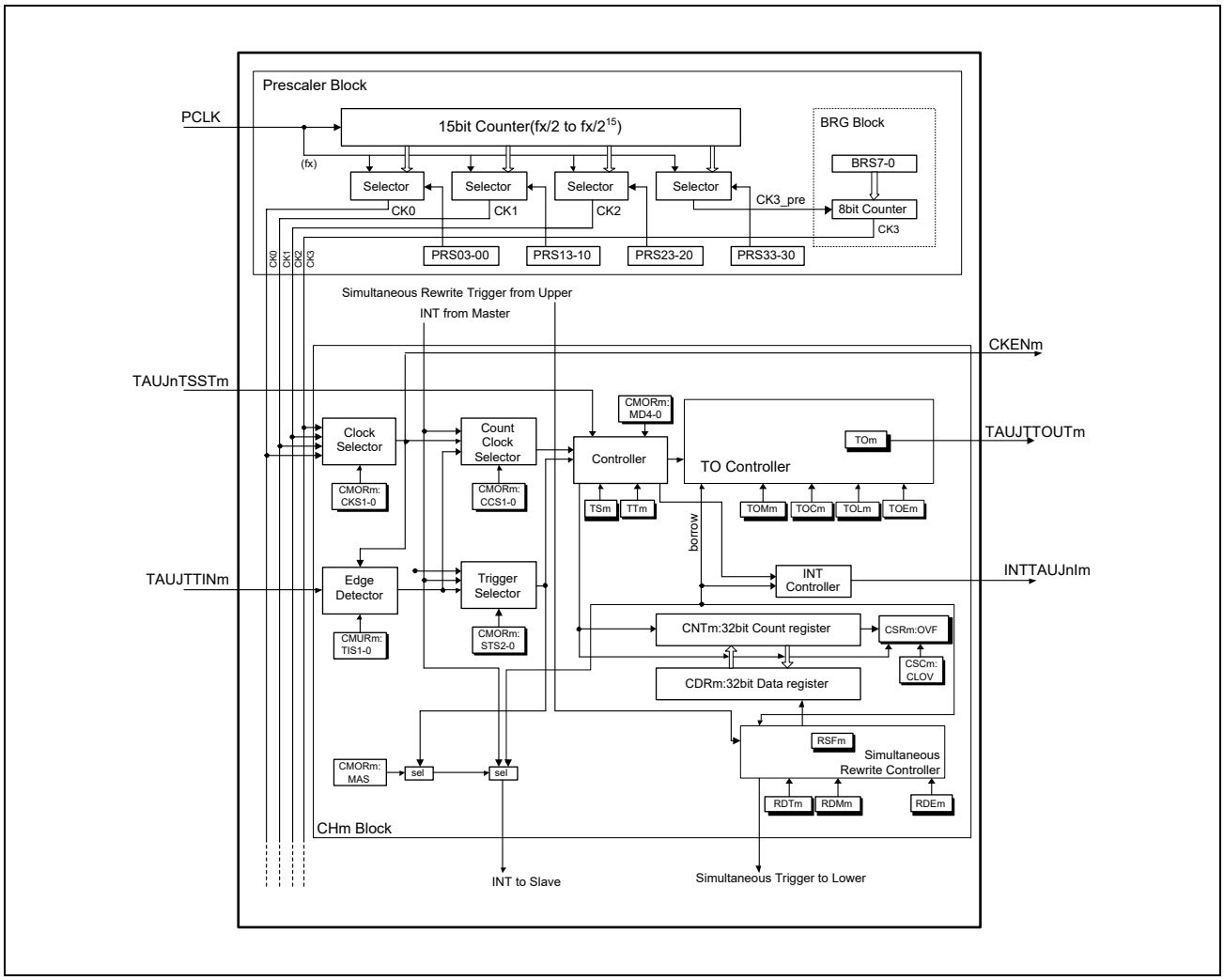


Figure 19.2 Block Diagram of the TAUJ

The prefix “TAUJn” has been omitted from the register names for the sake of clarity in the above figure.

- TAUJnTSSTm: Simultaneous start trigger (input from PIC1B)

19.2.2.1 Description of Blocks

The following describes the functional blocks:

Prescaler block

The prescaler block provides up to 4 clock signals (CK0 to CK3) that can be used as count clocks for all channels.

Count clocks CK0 to CK2 are derived from PCLK by a configurable prescaler division factor of 2^0 to 2^{15}. The fourth count clock CK3 can be adjusted more precisely by an additional division factor that is not a power of 2.

Clock and count clock selection

For every channel, the count clock selector selects which of the following is used as the clock source:

- One of the clocks CK0 to CK3 (selected by the clock selector)

Controller

The controller controls the main operations of the counter:

- Operating mode (selected with the TAUJnCMORm.TAUJnMD[4:0] bits)
- Counter start enable (TAUJnTS.TAUJnTSm) and counter stop (TAUJnTT.TAUJnTTm)

When counter start is enabled, status flag TAUJnTE.TAUJnTEm is set.

Trigger selector

Depending on the selected operation mode, the counter starts automatically when it is enabled (TAUJnTE.TAUJnTEm = 1), or it waits for an external start trigger signal. Any of the following signals can be used as the start trigger:

- Synchronous channel start trigger input TAUJnTSSTm
- Effective edge of TAUJTTINm input signal
- INTTAUJnIm from master channel

Simultaneous rewrite controller

Simultaneous rewrite control is a special function that can be used in synchronous operation functions. The data registers (TAUJnCDRm) of all channels in a channel group can be rewritten at any time. The simultaneous rewrite controller ensures that new data register values of all channels become effective at the same time.

TAUJnTO controller

The output control of every channel enables the generation of various output signals such as PWM signals.

19.3 Registers

19.3.1 List of Registers

TAUJ registers are listed in the following table.

For information on <TAUJn_base>, see **Section 19.1.2, Register Base Addresses.**

Table 19.9 TAUJn Registers Overview

Module name	Register name	Shortcut	Address
TAUJn prescaler registers			
TAUJn	TAUJn prescaler clock select register	TAUJnTPS	<TAUJn_base> + 90_H
TAUJn	TAUJn prescaler baud rate setting register	TAUJnBRS	<TAUJn_base> + 94_H
TAUJn control registers			
TAUJn	TAUJn channel data register m	TAUJnCDRm	<TAUJn_base> + m × 4_H
TAUJn	TAUJn channel counter register m	TAUJnCNTm	<TAUJn_base> + 10_H + m × 4_H
TAUJn	TAUJn channel mode OS register m	TAUJnCMORm	<TAUJn_base> + 80_H + m × 4_H
TAUJn	TAUJn channel mode user register m	TAUJnCMURm	<TAUJn_base> + 20_H + m × 4_H
TAUJn	TAUJn channel status register m	TAUJnCSRm	<TAUJn_base> + 30_H + m × 4_H
TAUJn	TAUJn channel status clear trigger register m TAUJnCSCm	TAUJnCSCm	<TAUJn_base> + 40_H + m × 4_H
TAUJn	TAUJn channel start trigger register	TAUJnTS	<TAUJn_base> + 54_H
TAUJn	TAUJn channel enable status register	TAUJnTE	<TAUJn_base> + 50_H
TAUJn	TAUJn channel stop trigger register	TAUJnTT	<TAUJn_base> + 58_H
TAUJn output registers			
TAUJn	TAUJn channel output enable register	TAUJnTOE	<TAUJn_base> + 60_H
TAUJn	TAUJn channel output register	TAUJnTO	<TAUJn_base> + $5C_H$
TAUJn	TAUJn channel output mode register	TAUJnTOM	<TAUJn_base> + 98_H
TAUJn	TAUJn channel output configuration register	TAUJnTOC	<TAUJn_base> + $9C_H$
TAUJn	TAUJn channel output active level register	TAUJnTOL	<TAUJn_base> + 64_H
TAUJn reload data registers			
TAUJn	TAUJn channel reload data enable register	TAUJnRDE	<TAUJn_base> + $A0_H$
TAUJn	TAUJn channel reload data mode register	TAUJnRDM	<TAUJn_base> + $A4_H$
TAUJn	TAUJn channel reload data trigger register	TAUJnRDT	<TAUJn_base> + 68_H
TAUJn	TAUJn channel reload status register	TAUJnRSF	<TAUJn_base> + $6C_H$

19.3.2 TAUJnTPS — TAUJn Prescaler Clock Select Register

This register specifies clocks CK0, CK1, CK2, and CK3_PRE for all channels of the PCLK prescalers. CK3 is generated by dividing CK3_PRE by the factor specified in TAUJnBRS.

Access: Readable/writable in 16-bit units.

Address: <TAUJn_base> + 90_H

Value after reset: $FFFF_H$. Any reset source triggers initialization.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnPRS3[3:0]				TAUJnPRS2[3:0]				TAUJnPRS1[3:0]				TAUJnPRS0[3:0]			
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.10 TAUJnTPS Register Contents (1/3)

Bit Position	Bit Name	Function
15 to 12	TAUJnPRS3[3:0]	These bits specify CK3_PRE clock. CK3_PRE clock is an input clock to BRG unit which supplies the CK3 operation clock to all channels.

TAUJnPRS3[3:0]	CK3_PRE Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK3 are stopped (TAUJnTE.TAUJnTEm = 0).

Table 19.10 TAUJnTPS Register Contents (2/3)

Bit Position	Bit Name	Function
11 to 8	TAUJnPRS2[3:0]	These bits specify the CK2 clock.

TAUJnPRS2[3:0]	CK2 Clock
0000_B	$PCLK/2^0$
0001_B	$PCLK/2^1$
0010_B	$PCLK/2^2$
0011_B	$PCLK/2^3$
0100_B	$PCLK/2^4$
0101_B	$PCLK/2^5$
0110_B	$PCLK/2^6$
0111_B	$PCLK/2^7$
1000_B	$PCLK/2^8$
1001_B	$PCLK/2^9$
1010_B	$PCLK/2^{10}$
1011_B	$PCLK/2^{11}$
1100_B	$PCLK/2^{12}$
1101_B	$PCLK/2^{13}$
1110_B	$PCLK/2^{14}$
1111_B	$PCLK/2^{15}$

The above bits are rewritable only when all the counters using CK2 are stopped (TAUJnTE.TAUJnTEm = 0).

7 to 4	TAUJnPRS1[3:0]	These bits specify the CK1 clock.

TAUJnPRS1[3:0]	CK1 Clock
0000_B	$PCLK/2^0$
0001_B	$PCLK/2^1$
0010_B	$PCLK/2^2$
0011_B	$PCLK/2^3$
0100_B	$PCLK/2^4$
0101_B	$PCLK/2^5$
0110_B	$PCLK/2^6$
0111_B	$PCLK/2^7$
1000_B	$PCLK/2^8$
1001_B	$PCLK/2^9$
1010_B	$PCLK/2^{10}$
1011_B	$PCLK/2^{11}$
1100_B	$PCLK/2^{12}$
1101_B	$PCLK/2^{13}$
1110_B	$PCLK/2^{14}$
1111_B	$PCLK/2^{15}$

The above bits are rewritable only when all the counters using CK1 are stopped (TAUJnTE.TAUJnTEm = 0).

Table 19.10 TAUJnTPS Register Contents (3/3)

Bit Position	Bit Name	Function
3 to 0	TAUJnPRS0[3:0]	These bits specify the CK0 clock.

TAUJnPRS0[3:0]	CK0 Clock
0000_B	$PCLK/2^0$
0001_B	$PCLK/2^1$
0010_B	$PCLK/2^2$
0011_B	$PCLK/2^3$
0100_B	$PCLK/2^4$
0101_B	$PCLK/2^5$
0110_B	$PCLK/2^6$
0111_B	$PCLK/2^7$
1000_B	$PCLK/2^8$
1001_B	$PCLK/2^9$
1010_B	$PCLK/2^{10}$
1011_B	$PCLK/2^{11}$
1100_B	$PCLK/2^{12}$
1101_B	$PCLK/2^{13}$
1110_B	$PCLK/2^{14}$
1111_B	$PCLK/2^{15}$

The above bits are rewritable only when all the counters using CK0 are stopped (TAUJnTE.TAUJnTEm = 0).

NOTE

TAUJn clock input PCLK is defined in the first part of this section, **Section 19.1.3, Clock Supply.**

19.3.3 TAUJnBRS — TAUJn Prescaler Baud Rate Setting Register

This register specifies the division factor of prescaler clock CK3.

CK3 is generated by dividing CK3_PRE by the factor specified in this register plus one. The PCLK prescaler for CK3_PRE is specified in TAUJnTPS. TAUJnPRS3[3:0].

Access: Readable/writable in 8-bit units.

Address: <TAUJn_base> + 94_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	TAUJnBRS[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.11 TAUJnBRS Register Contents

Bit Position	Bit Name	Function
7 to 0	TAUJnBRS [7:0]	These bits specify the division factor of CK3_PRE for generating CK3.

TAUJnBRS[7:0]	CK3 Clock
$0000\ 0000_B$	CK3_PRE / 1
$0000\ 0001_B$	CK3_PRE / 2
$0000\ 0010_B$	CK3_PRE / 3
$0000\ 0011_B$	CK3_PRE / 4
:	:
$1111\ 1110_B$	CK3_PRE / 255
$1111\ 1111_B$	CK3_PRE / 256

19.3.4 TAUJnCDRm — TAUJn Channel Data Register

This register functions either as a compare register or as a capture register, depending on the operating mode specified in TAUJnCMORm.TAUJnMD[4:1].

Access: Readable/writable in 32-bit units.
Readable in capture mode. Any write operation is ignored.
Readable/writable in compare mode.

Address: <TAUJn_base> + 0_H + m × 4_H

Value after reset: 0000 0000_H. Any reset source triggers initialization.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TAUJnCDR[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCDR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.12 TAUJnCDRm Register Contents

Bit Position	Bit Name	Function
31 to 0	TAUJnCDR[31:0]	Data register for capture/compare values

19.3.5 TAUJnCNTm — TAUJn Channel Counter Register

This is a channel m counter register.

Access: This register is read-only in 32-bit units.

Address: <TAUJn_base> + 10_H + m × 4_H

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TAUJnCNT[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCNT[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 19.13 TAUJnCNTm Register Contents

Bit Position	Bit Name	Function
31 to 0	TAUJnCNT[31:0]	32-bit counter value

The read value depends on a counter, an operating mode change, or TAUJnTS.TAUJnTSm/TAUJnTT.TAUJnTTm bit value.

The initial counter read value depends on the operating mode and how the counter is stopped.

- By a reset
- By a counter stop trigger (TAUJnTT.TAUJnTTm = 1)

The following table lists the initial counter read values after the counting is stopped (TAUJnTE.TAUJnTEm = 0) and re-enabled (TAUJnTS.TAUJnTSm = 1).

The table also contains the counter value which is read one count after the counting is enabled (TAUJnTS.TAUJnTSm = 1) while waiting for a start trigger.

NOTE

If operating mode is changed while the counter is stopped, the initial counter value after the counter is restarted becomes undefined. Operating mode can be changed by the TAUJnCMORm.TAUJnMD[4:1] register.

Table 19.14 TAUJnCNTm Read Values after Re-Enabling Counter

Mode Name	Count Method (Up/Down)	TAUJnCNTm		
		Start Value*1	After Stop Trigger	After One Count
Interval timer mode	Count down	FFFF FFFF$_H$	Stop value	—
Capture mode	Count up	0000 0000$_H$	Stop value	—
One-count mode	Count down	FFFF FFFF$_H$	Stop value	Stop value
Capture and one-count mode	Count up	0000 0000$_H$	Stop value	Capture value + 1 (TAUJnCDRm)
Count capture mode	Count up	0000 0000$_H$	Stop value	—
Capture and gate count mode	Count up	0000 0000$_H$	Stop value	Stop value

Note 1. The value set for TAUJnCNTm when the operation mode is changed after reset release.

19.3.6 TAUJnCMORm — TAUJn Channel Mode OS Register

This register controls channel m operation.

Access: Readable/writable in 16-bit units. Writable only when the counter is stopped (TAUJnTE.TAUJnTEm = 0).

Address: <TAUJn_base> + 80$_H$ + m × 4$_H$

Value after reset: 0000$_H$. Any reset source triggers initialization.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.15 TAUJnCMORm Register Contents (1/3)

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select an operation clock, which is used with the TAUJTTINm input edge detection circuit. Setting of TAUJnCMORm.TAUJnCCS[1:0] bit also allows the operation clock to serve as a count clock of TAUJnCNTm.

TAUJn CKS1	TAUJn CKS0	Selection of Operation Clock
0	0	CK0
0	1	CK1
1	0	CK2
1	1	CK3

Bit Position	Bit Name	Function
13, 12	TAUJnCCS[1:0]	These bits select a count clock for TAUJnCNTm counter.

TAUJn CCS1	TAUJn CCS0	Selection of Count Clock
0	0	Operation clock specified by TAUJnCMORm.TAUJnCKS[1:0].
0	1	Setting prohibited
1	0	
1	1	

Table 19.15 TAUJnCMORm Register Contents (2/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>11</td><td>TAUJnMAS</td><td>This bit specifies whether the channel is a master or slave channel during synchronous channel operation.
0: Slave
1: Master
This bit setting is valid only for even-numbered channels (CHm_even). Odd-numbered channels (CHm-odd) are fixed to 0.</td></tr>
<tr><td>10 to 8</td><td>TAUJnSTS[2:0]</td><td>These bits select a start trigger.
<table>
<tr><th>TAUJn STS2</th><th>TAUJn STS1</th><th>TAUJn STS0</th><th>Functional Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Software trigger</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Effective edge of TAUJTTINm input signal, which is specified by TAUJnCMURm.TAUJnTIS[1:0].</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Effective edge of TAUJTTINm input signal is used as a start trigger and the reverse edge as a stop trigger.</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Simultaneous rewrite trigger</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>INTTAUJnIm of the master channel</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td rowspan="3">Setting prohibited</td></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
</table></td></tr>
<tr><td>7, 6</td><td>TAUJnCOS[1:0]</td><td>These bits specify the timing for updating capture register TAUJnCDRm and overflow flag TAUJnCSRm.TAUJnOVF of channel m.
These bits are only valid if channel m is for capture function (capture mode and capture & one-count mode).
<table>
<tr><th>TAUJn COS1</th><th>TAUJn COS0</th><th>TAUJnCDRm</th><th>TAUJnCSRm.TAUJnOVF</th></tr>
<tr><td>0</td><td>0</td><td rowspan="2">Updated when effective edge of TAUJTTINm input is detected.</td><td>Updated (cleared or set) when effective edge of TAUJTTINm input is detected:
• Set TAUJnCSRm.TAUJnOVF if a counter overflow has occurred since the last effective edge was detected.
• Clear TAUJnCSRm.TAUJnOVF if no counter overflow has occurred since the last effective edge was detected.</td></tr>
<tr><td>0</td><td>1</td><td>Set when a counter overflow occurs and cleared by setting TAUJnCSCm.TAUJnCLOV to 1.</td></tr>
<tr><td>1</td><td>0</td><td rowspan="2">Updated when effective edge of TAUJTTINm input is detected and when a counter overflow occurs.
• Detection of effective edge of TAUJTTINm input:
The counter value is written into TAUJnCDRm.
• Occurrence of overflow:
FFFF FFFF$_H$ is loaded into TAUJnCDRm.
Detection of the next effective edge of TAUJTTINm is ignored.</td><td>No setting</td></tr>
<tr><td>1</td><td>1</td><td>Set when a counter overflow occurs and cleared by setting TAUJnCSCm.TAUJnCLOV to 1.</td></tr>
</table></td></tr>
</table>

Table 19.15 TAUJnCMORm Register Contents (3/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>5</td><td>Reserved</td><td>When read, the value after reset is read.
When writing, write the value after reset.</td></tr>
<tr><td>4 to 0</td><td>TAUJnMD[4:0]</td><td>These bits specify an operating mode.
<table>
<tr><th>TAUJn MD4</th><th>TAUJn MD3</th><th>TAUJn MD2</th><th>TAUJn MD1</th><th>TAUJn MD0</th><th>Functional Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1/0</td><td>Interval timer mode</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>1/0</td><td>Capture mode</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1/0</td><td>One-count mode</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Capture and one-count mode</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1/0</td><td>Count capture mode</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Capture and gate count mode</td></tr>
</table>
Settings other than the above are prohibited.
<table>
<tr><th>Mode</th><th>Role of TAUJnMD0 Bit</th></tr>
<tr><td>Interval timer mode
Capture mode
Count capture mode</td><td>Specifies whether INTTAUJnIm is generated at the beginning of count operation (when a start trigger is entered) or not.
0: INTTAUJnIm is not generated.
1: INTTAUJnIm is generated.</td></tr>
<tr><td>One-count mode</td><td>Enables or disables start trigger detection during counting.
0: Disables detection.
1: Enables detection.
CAUTION: In one-count mode, INTTAUJnIm signal is not output at the beginning of count operation.</td></tr>
<tr><td>Capture and one-count mode
Capture and gate count mode</td><td>This bit should be set to 0.
CAUTION: INTTAUJnIm signal is not output at the beginning of count operation. In addition, start trigger detected during counting is disabled.</td></tr>
</table>
</td></tr>
</table>

19.3.7 TAUJnCMURm — TAUJn Channel Mode User Register

This register specifies a type of effective edge detection used for TAUJTTINm input.

Access: Readable/writable in 8-bit units.

Address: <TAUJn_base> + 20_H + m × 4_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.16 TAUJnCMURm Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	These bits specify an effective edge of TAUJTTINm input signal.

TAUJn TIS1	TAUJn TIS0	Functional Description
0	0	Falling edge
0	1	Rising edge
1	0	Detection of falling and rising edges (selection of low width measurement) Start trigger: Falling edge Stop trigger (capture): Rising edge
1	1	Detection of falling and rising edges (selection of high width measurement) Start trigger: Rising edge Stop trigger (capture): Falling edge

Edge detection of TAUJTTINm input signal is based on the operation clock selected by TAUJnCMORm.TAUJnCKS[1:0].

19.3.8 TAUJnCSRm — TAUJn Channel Status Register

This register indicates the overflow status of channel m.

Access: This register is read-only in 8-bit units.

Address: <TAUJn_base> + 30_H + m × 4_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAUJnOVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 19.17 TAUJnCSRm Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read.
1	Reserved	When read, the value returned is undefined.
0	TAUJnOVF	This bit indicates the counter overflow status: 0: No overflow occurs 1: Overflow occurs This bit is used only in the following modes: • Capture mode • Capture and one-count mode The function of this bit depends on the setting of control bits TAUJnCMORm.TAUJnCOS[1:0].

19.3.9 TAUJnCSCm — TAUJn Channel Status Clear Register

This register is a trigger register for clearing the overflow flag TAUJnCSRm.TAUJnOVF of channel m.

Access: This register can only be written in 8-bit units. The read value is always 00_H.

Address: <TAUJn_base> + 40_H + m × 4_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAUJnCLOV
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 19.18 TAUJnCSCm Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TAUJnCLOV	0: No function 1: Clears the overflow flag TAUJnCSRm.TAUJnOVF

19.3.10 TAUJnTS — TAUJn Channel Start Trigger Register

This register enables the counter operation for each channel.

Access: This register can only be written in 8-bit units. The read value is always 00_H.

Address: <TAUJn_base> + 54_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTS03	TAUJnTS02	TAUJnTS01	TAUJnTS00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	W

Table 19.19 TAUJnTS Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When writing, write the value after reset.
3 to 0	TAUJnTSm	These bits enable the counter operation for channel m: 0: No function 1: Enables the counter operation and sets TAUJnTE.TAUJnTEm = 1 Only the counter operation is enabled even if TAUJnTE.TAUJnTEm = 1. Whether the counter is started or not depends on the selected of operating mode.

19.3.11 TAUJnTE — TAUJn Channel Enable Status Register

This register enables/disables a counter operation.

Access: This register is read-only in 8-bit units.

Address: <TAUJn_base> + 50_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTE03	TAUJnTE02	TAUJnTE01	TAUJnTE00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 19.20 TAUJnTE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read.
3 to 0	TAUJnTEm	These bits enable or disable channel m's counter operation. 0: Disables the counter operation 1: Enables the counter operation These bits are set to 1 when trigger input of TAUJnTSSTm (synchronous channel start trigger signal) is detected or when TAUJnTS.TAUJnTSm are set to 1. These bits are set to 0 when TAUJnTT.TAUJnTTm are set to 1.

19.3.12 TAUJnTT — TAUJn Channel Stop Trigger Register

This register stops the counter operation of each channel.

Access: This register can only be written in 8-bit units. The read value is always 00_H.

Address: <TAUJn_base> + 58_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTT03	TAUJnTT02	TAUJnTT01	TAUJnTT00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	W

Table 19.21 TAUJnTT Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When writing, write the value after reset.
3 to 0	TAUJnTTm	These bits stop channel m's counter operation. 0: No function 1: Stops the counter operation and resets TAUJnTE.TAUJnTEm TAUJnCNTm, TAUJnTO.TAUJnTOm, and TAUJTTOUTm retain the values provided before the counter is stopped.

19.3.13 TAUJnTOE — TAUJn Channel Output Enable Register

This register enables and disables independent channel output mode controlled by Software.

Access: Readable/writable in 8-bit units.

Address: <TAUJn_base> + 60_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTOE03	TAUJnTOE02	TAUJnTOE01	TAUJnTOE00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 19.22 TAUJnTOE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOEm	These bits enable or disable independent channel output mode: 0: Disables the independent timer output function (controlled by software) 1: Enables the independent timer output function

19.3.14 TAUJnTO — TAUJn Channel Output Register

This register specifies and reads the level of TAUJTTOUTm.

Access: Readable/writable in 8-bit units.

Address: <TAUJn_base> + $5C_H$

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTO03	TAUJnTO02	TAUJnTO01	TAUJnTO00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 19.23 TAUJnTO Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOm	These bits specify and read the level of TAUJTTOUTm: 0: Low 1: High Only TAUJnTOm bits for which independent channel output function is disabled (TAUJnTOEm = 0) can be written.

19.3.15 TAUJnTOM — TAUJn Channel Output Mode Register

This register specifies the output mode of each channel.

Access: Readable/writable in 8-bit units. It can only be written when the counter is stopped (TAUJnTE.TAUJnTEm = 0).

Address: <TAUJn_base> + 98_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTOM03	TAUJnTOM02	TAUJnTOM01	TAUJnTOM00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 19.24 TAUJnTOM Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOMm	These bits specify the channel output mode: 0: Independent channel output mode 1: Synchronous channel output mode The output mode depends on the setting of the output control bits (TAUJnTOE.TAUJnTOEm) of each channel.

19.3.16 TAUJnTOC — TAUJn Channel Output Configuration Register

This register specifies the output mode of each channel in combination with TAUJnTOMm.

Access: Readable/writable in 8-bit units. It can only be written when the counter is stopped (TAUJnTE.TAUJnTEm = 0).

Address: <TAUJn_base> + $9C_H$

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTOC03	TAUJnTOC02	TAUJnTOC01	TAUJnTOC00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 19.25 TAUJnTOC Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOCm	These bits specify the output mode: 0: Operation mode 1 (= Toggle mode) 1: No function These bits must be set to 0 for all output modes except Independent Channel Output Mode Controlled by Software. The output mode also depends on TAUJnTOM.TAUJnTOMm, as can be seen in the following table.

TAUJn TOMm	TAUJn TOCm	Functional Description
0	0	Toggle mode: Toggle operation is conducted when INTTAUJnIm occurs.
0	1	No function
1	0	Synchronous Channel Operation Mode 1: Set when INT occurs on the master channel and reset when INT occurs on the slave channel.
1	1	No function

19.3.17 TAUJnTOL — TAUJn Channel Output Level Register

This register specifies the output logic of the channel output bit (TAUJnTO.TAUJnTOm).

Access: Readable/writable in 8-bit units.

Address: <TAUJn_base> + 64_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTOL03	TAUJnTOL02	TAUJnTOL01	TAUJnTOL00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 19.26 TAUJnTOL Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOLm	These bits specify the output logic of the channel m output bit (TAUJnTO.TAUJnTOm): 0: Positive logic (active high) 1: Negative logic (active low) These bits apply in all channel output modes except independent channel output mode controlled by software and independent channel output mode 1.

19.3.18 TAUJnRDE — TAUJn Channel Reload Data Enable Register

This register enables and disables simultaneous rewrite of the data register TAUJnCDRm. It also enables and disables simultaneous rewrite of the data register TAUJnTOLm for the PWM output function.

Access: Readable/writable in 8-bit units. It can only be written when TAUJnTE.TAUJnTEm = 0.

Address: <TAUJn_base> + $A0_H$

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnRDE03	TAUJnRDE02	TAUJnRDE01	TAUJnRDE00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 19.27 TAUJnRDE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnRDEm	These bits enable or disable simultaneous rewrite of the data register of channel m: 0: Disables simultaneous rewrite 1: Enables simultaneous rewrite

19.3.19 TAUJnRDM — TAUJn Channel Reload Data Mode Register

This register selects when the signal that controls simultaneous rewrite is generated.

Access: Readable/writable in 8-bit units. It can only be written when TAUJnTE.TAUJnTEm=0.

Address: <TAUJn_base> + $A4_H$

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnRDM03	TAUJnRDM02	TAUJnRDM01	TAUJnRDM00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 19.28 TAUJnRDM Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnRDMm	These bits specify when the signal that triggers simultaneous rewrite is generated: 0: When the master channel counter starts counting 1: No function These bits only apply when TAUJnRDE.TAUJnRDEm = 1.

19.3.20 TAUJnRDT — TAUJn Channel Reload Data Trigger Register

This register triggers the simultaneous rewrite enabling state.

Access: This register can only be written in 8-bit units. The read value is always 00_H.

Address: <TAUJn_base> + 68_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnRDT3	TAUJnRDT2	TAUJnRDT1	TAUJnRDT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	W

Table 19.29 TAUJnRDT Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When writing, write the value after reset.
3 to 0	TAUJnRDTm	These bits trigger the simultaneous rewrite enabling state: 0: No function 1: Simultaneous rewrite enabling state is triggered. The simultaneous rewrite enabling flag (TAUJnRSFm) is set to 1. The system waits for the simultaneous rewrite trigger. The setting of these bits is applied only to the following case: • TAUJnRDE.TAUJnRDEm = 1

19.3.21 TAUJnRSF — TAUJn Channel Reload Status Register

This flag register indicates the simultaneous rewrite status.

Access: This register is read-only in 8-bit units.

Address: <TAUJn_base> + $6C_H$

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnRSF03	TAUJnRSF02	TAUJnRSF01	TAUJnRSF00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 19.30 TAUJnRSF Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read.
3 to 0	TAUJnRSFm	These bits indicate the simultaneous rewrite status: 0: Indicates that simultaneous rewrite has been completed due to the generation of simultaneous rewrite trigger. 1: Indicates that the system waits for a simultaneous rewrite trigger in the simultaneous rewrite enabling state (TAUJnRDTm = 1).

19.4 Function

19.4.1 General Operating Procedure

The following lists the general operation procedure for the TAUJn:

After reset release, the operation of each channel is stopped. Clock supply is started and writing to each register is enabled. All circuits and registers of all channels are initialized. The control register of TAUJTTOUTm is also initialized and outputs a low level.

(1) Set the TAUJnTPS and TAUJnBRS registers to specify the clock frequency of CK0 to CK3.

(2) Configure the desired TAUJn function:

- Set the operation mode
- Set any other control bits

(3) Enable the counter by setting the TAUJnTS.TAUJnTSm bit to 1.
The counter starts to count immediately, or when an appropriate trigger is detected, depending on the bit settings.

(4) Stop the counter or perform a forced restart operation during count operation. The counter can be stopped by setting the TAUJnTT.TAUJnTTm bit to 1. The counter can be forcibly restarted by setting the TAUJnTS.TAUJnTSm bit to 1.

(5) Stop the function by setting the TAUJnTT.TAUJnTTm bit to 1.

NOTE

A detailed description of the required control bits and the operation of the individual functions is given in **Section 19.4.9, Independent Channel Operation Functions**, **Section 19.4.10, Synchronous Channel Operation Functions**.

19.4.2 Concepts of Synchronous Channel Operation Function

The synchronous channel operation function is implemented using a combination of channel groups (comprised of master and slave channels).

Several rules apply to the settings of channels.

These rules are detailed in **Section 19.4.2.1, Rules of Synchronous Channel Operation Function**.

Two special features for the synchronous channel operation function are detailed in the following section:

- **Section 19.4.10, Synchronous Channel Operation Functions**

19.4.2.1 Rules of Synchronous Channel Operation Function

Number of master and slave channels

- Only even-numbered channels (CH0, CH2) can be set as master channels. Any channel apart from CH0 can be set as a slave channel.
- Only channels lower than the master channel can be set as slave channels, and several slave channels can be set for one master channel.
  Example: If CH2 is a master channel, CH3 can be set as slave channel.
- If two master channels are used, slave channels cannot cross the master.
  Example: If CH0 and CH2 are master channels, CH1 can be set as slave channels for CH0, but CH3 cannot.

Count clock

- The same count clock should be set for the master channel and the slave channels synchronizing to the master channel. This is achieved by setting the same value in the TAUJnCMORm.TAUJnCKS[1:0] bits of the master and slave channels.

Control trigger signal for master/salve channels

- Master channels can output control trigger signals to slave channels.
- Slave channels can use control trigger signals from master channels but cannot output control trigger signals for their own to lower channels.
- Master channels cannot use control trigger signals from upper master channels.

The basic concepts of master/slave usage and count clocks are illustrated in **Figure 19.3**.

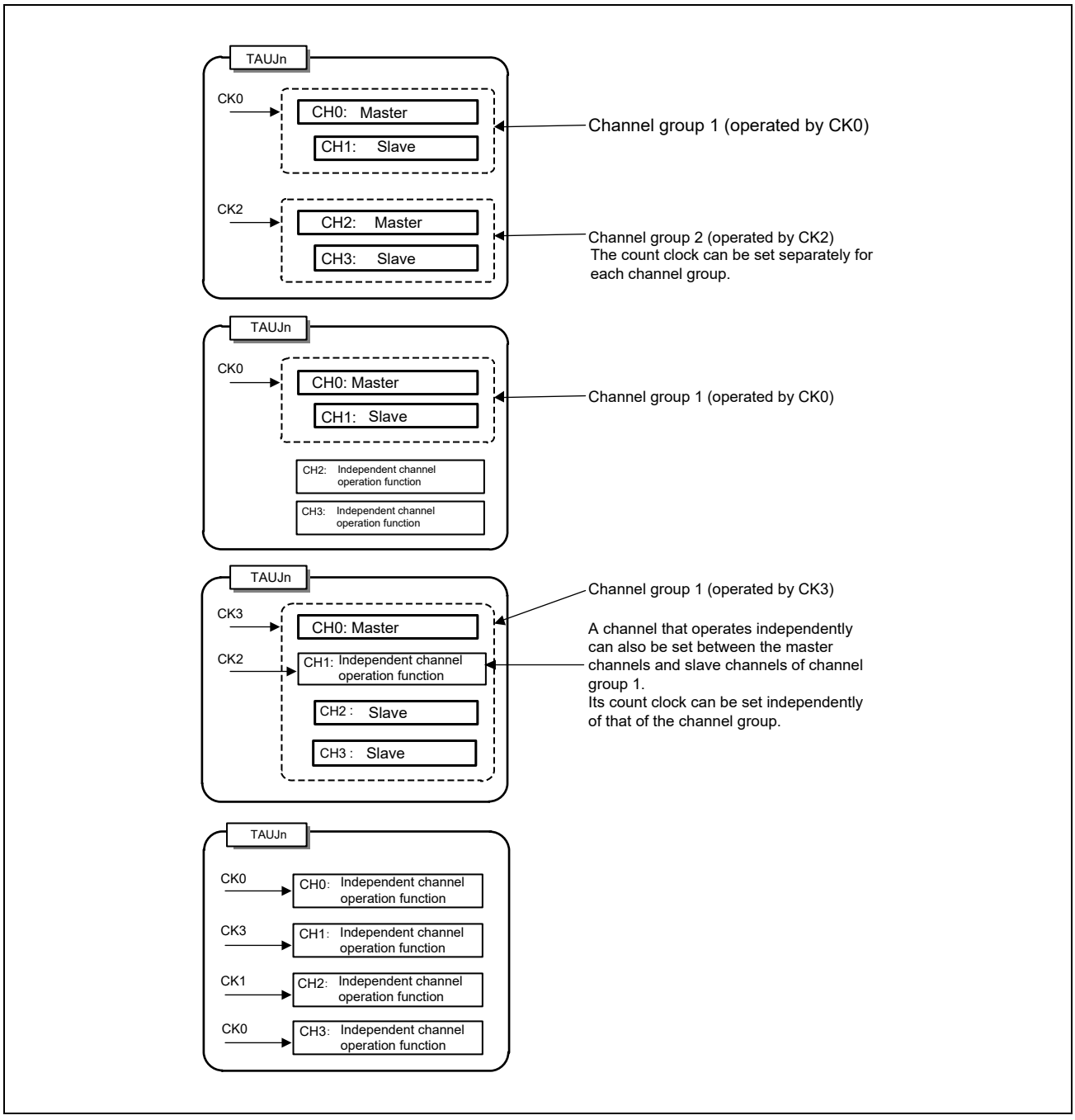


Figure 19.3 Grouping of Channels and Assignment of Count Clocks

19.4.2.2 Simultaneous Start and Stop of Synchronous Channel Counters

Channels that are operated synchronously can be started and stopped simultaneously, both within a TAUJ unit and between TAUJ units.

(1) Simultaneous start and stop within a TAUJ unit

- To simultaneously start synchronized channels, the TAUJnTS.TAUJnTSm bits of the channels should be set at the same time.
- To simultaneously stop synchronized channels, the TAUJnTT.TAUJnTTm bits of the channels should be set at the same time.

Setting 1 in the TAUJnTS.TAUJnTSm bits sets the corresponding TAUJnTE.TAUJnTEm bits to 1, enabling counting. The count start timing of the counter depends on the operating mode.

(2) Simultaneous start between TAUJ units

Counters in different TAUJ units can also be started simultaneously if the relevant counters are enabled before receiving the simultaneous trigger signal.

19.4.3 Simultaneous Rewrite

19.4.3.1 How to Control Simultaneous Rewrite

The following figure shows the general procedure for simultaneous rewrite. The three main blocks (Initial settings, Start counter & count operation, and Simultaneous rewrite) are explained afterwards.

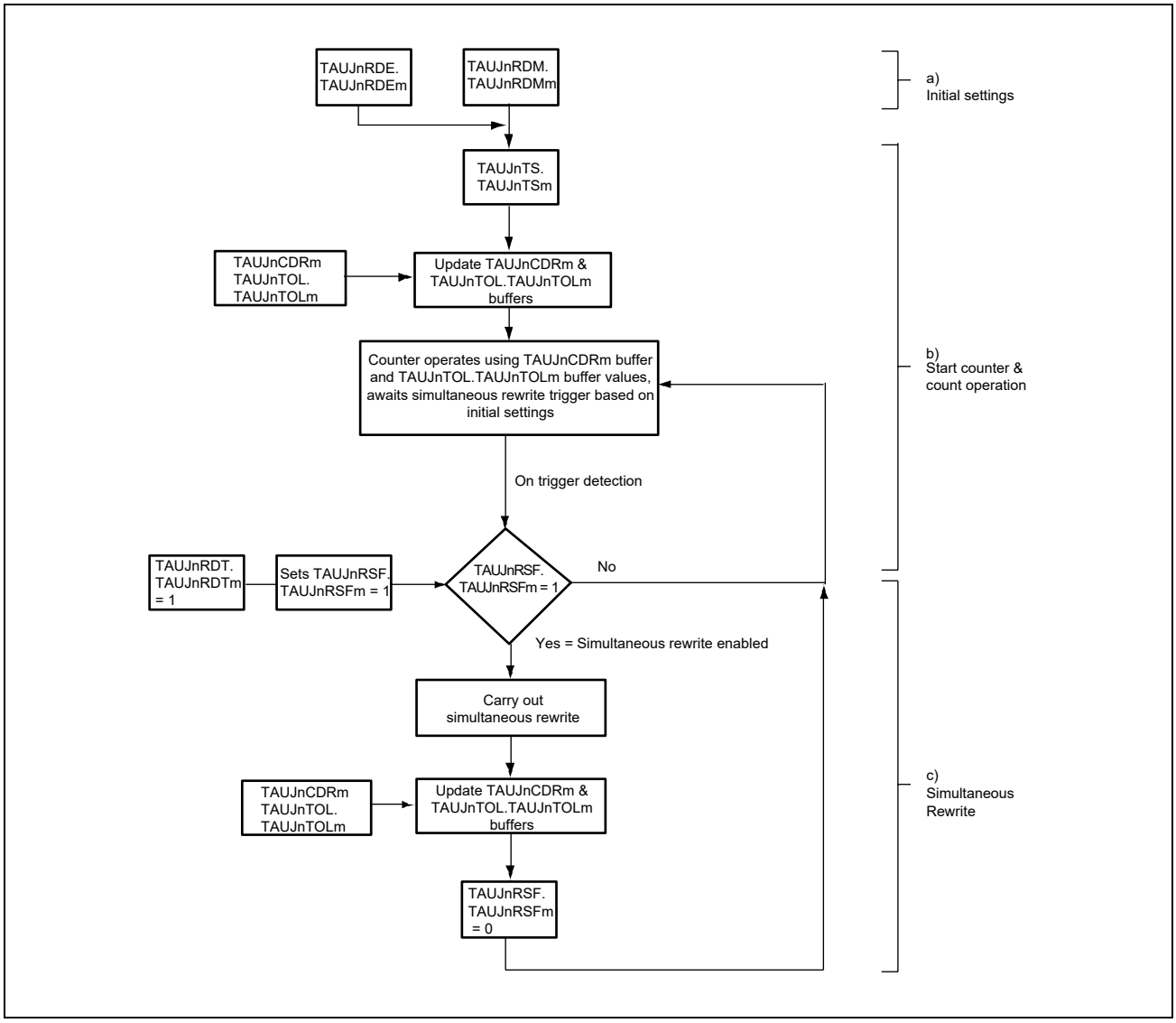


Figure 19.4 General Procedure for Simultaneous Rewrite

(1) Initial settings

- To enable simultaneous rewrite in channel m, set TAUJnRDE.TAUJnRDEm = 1
- To select simultaneous rewrite when the master channel starts counting, set TAUJnRDM.TAUJnRDMm

(2) Start counter and count operation

- To start all the TAUJnCNTm counters in the channel group, set the corresponding TAUJnTS.TAUJnTSm bits to 1. TAUJnTOL.TAUJnTOLm and the values in the data registers (TAUJnCDRm) are loaded into the corresponding TAUJnTOL.TAUJnTOLm buffer (TAUJnTOL.TAUJnTOLm buf) and data buffer registers (TAUJnCDRm buf) and the counters start.
- Setting the reload data trigger bit (TAUJnRDT.TAUJnRDTm) to 1 sets the reload flag (TAUJnRSF.TAUJnRSFm) to 1, enabling simultaneous rewrite. TAUJnRSF.TAUJnRSFm remains set to 1 until simultaneous rewrite is completed.
- When a specified trigger for simultaneous rewrite is detected, the TAUJnRSF.TAUJnRSFm bit is checked to see if simultaneous rewrite is enabled (TAUJnRSF.TAUJnRSFm = 1). If enabled, simultaneous rewrite is carried out. Otherwise simultaneous rewrite is not carried out and the system waits for detection of the next simultaneous rewrite trigger.

(3) Simultaneous rewrite

- When the simultaneous rewrite trigger is detected and simultaneous rewrite is enabled (TAUJnRSF.TAUJnRSFm = 1), the current values of the data registers are copied to their buffers. These values are then loaded into the corresponding counters and the values are applied the next time the counter starts or restarts.
- When the simultaneous rewrite is completed, the TAUJnRSF.TAUJnRSFm bit is set to 0, and the system awaits the next simultaneous rewrite trigger.

19.4.3.2 Other General Rules for Simultaneous Rewrite

The following rules also apply.

- TAUJnRDE.TAUJnRDEm and TAUJnRDM.TAUJnRDMm cannot be changed while the counter is in operation (TAUJnTE.TAUJnTEm = 1).
- TAUJnTOL.TAUJnTOLm can be rewritten only during operation using the PWM output function. For all other functions, TAUJnTOL.TAUJnTOLm should be written before the counter starts. If it is rewritten while any other function is used, TAUJTTOUTm outputs an invalid wave.

19.4.3.3 Simultaneous Rewrite Procedure

The simultaneous rewrite procedure in the PWM output function is described in the following figure.

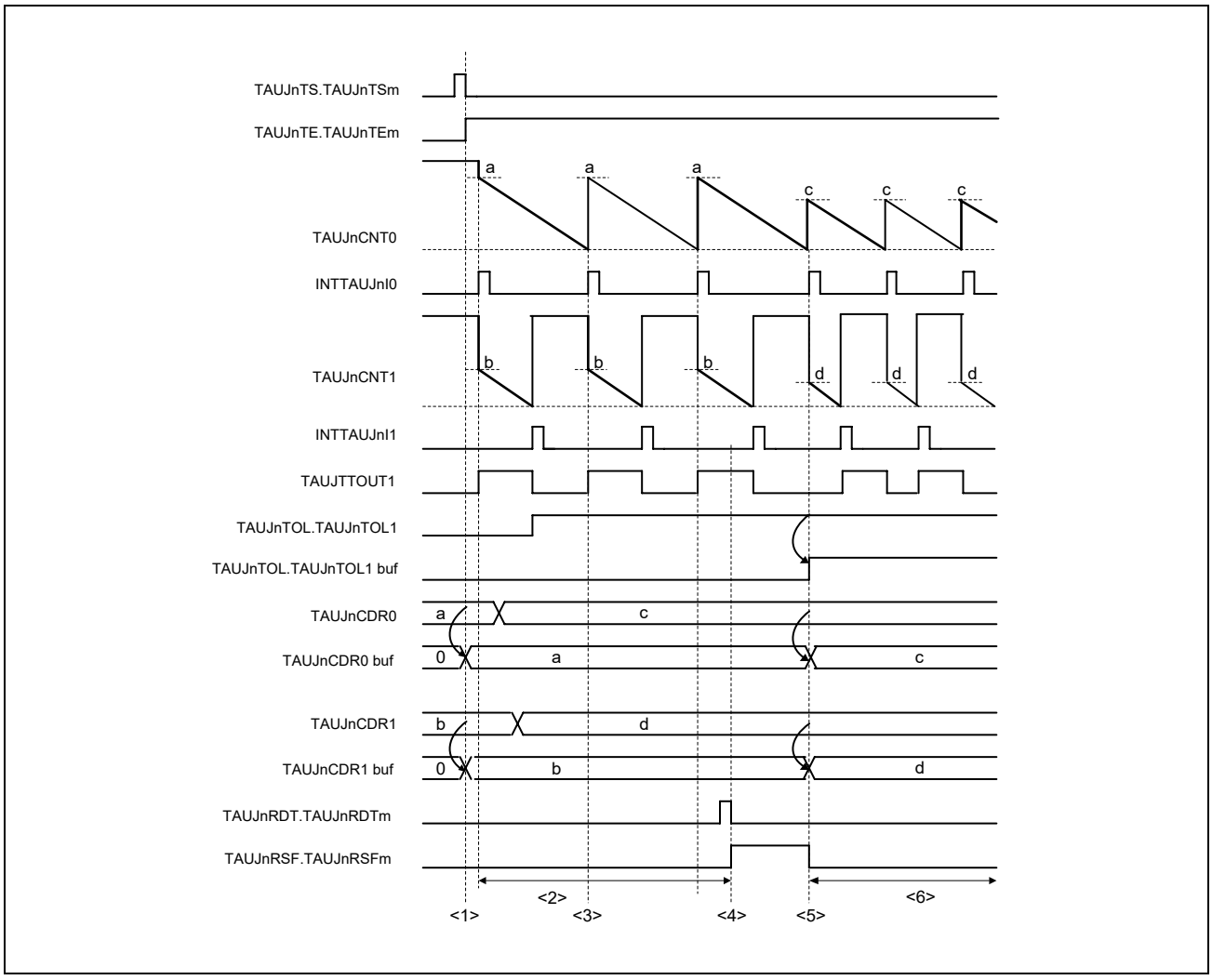


Figure 19.5 Simultaneous Rewrite when the Master Channel Starts/Restarts to Count

Setting

CH0 is a master channel of the PWM output function, and CH1 represents a slave channel of the PWM output function. Simultaneous rewrite is applied when the master channel starts counting.

Description:

(1) When TAUJnTS.TAUJnTSm = 1 is set, the value of TAUJnCDRm is copied to the TAUJnCDRm buffer and the value of TAUJnTOL.TAUJnTOLm is copied to the TAUJnTOL.TAUJnTOLm buffer.

(2) The TAUJnCDRm and TAUJnTOL.TAUJnTOLm registers can be written at any time.

(3) CH0 restarts counting, but simultaneous rewrite does not occur because it is disabled (TAUJnRSF.TAUJnRSFm = 0).

(4) The reload data trigger bit (TAUJnRDT.TAUJnRDTm) is set to 1 which sets the status flag (TAUJnRSF.TAUJnRSFm = 1), enabling simultaneous rewrite.

(5) Simultaneous rewrite is triggered when CH0 restarts counting, because simultaneous rewrite is enabled. The TAUJnCDRm value is loaded into the TAUJnCDRm buffer and the TAUJnTOL.TAUJnTOLm value is loaded into the TAUJnTOL.TAUJnTOLm buffer.

(6) The counters count down and await the next simultaneous rewrite trigger. The values of TAUJnCDRm and TAUJnTOL.TAUJnTOLm can be changed again.

19.4.4 Channel Output Modes

The output of the TAUJTTOUTm pin can be controlled in two ways, the latter of which can be further split into individual modes.

- By software (TAUJnTOE.TAUJnTOEm = 0)
  When controlled by software, the value written in the output register bit (TAUJnTO.TAUJnTOm) is sent out of the output pin (TAUJTTOUTm).
- By TAUJ signals (TAUJnTOE.TAUJnTOEm = 1)
  When controlled by TAUJ signals, the output level of TAUJTTOUTm is set or reset or toggled by internal signals. The value of TAUJnTO.TAUJnTOm is updated accordingly to reflect the value of TAUJTTOUTm.
  - Independently (TAUJnTOM.TAUJnTOMm = 0)
    In case of independent operation, the output of the TAUJTTOUTm pin is only affected by settings of channel m. Therefore, independent channel operation should be selected (TAUJnTOM.TAUJnTOMm = 0).
  - Synchronously (TAUJnTOM.TAUJnTOMm = 1)
    In case of synchronous operation, the output of the TAUJTTOUTm pin is affected by settings of channel m and those of other channels. Therefore, synchronous channel operation should be selected for all synchronized channels (TAUJnTOM.TAUJnTOMm = 1).

The TAUJnTO.TAUJnTOm bit can always be read to determine the current value of TAUJTTOUTm, regardless of whether the pin is controlled by software, operated independently, or operated synchronously.

Control bits

The settings of the control bits required to select a specific channel output mode are listed in **Table 19.31, Channel Output Modes**.

The channel output modes are described in details below.

- **Section 19.4.4.2, Channel Output Modes Controlled Independently by TAUJn Signals**
- **Section 19.4.4.3, Channel Output Modes Controlled Synchronously by TAUJn Signals**

Batch operation of TAUJnTOm bit

Whether a set value is reflected to the TAUJnTOm bit or not is controlled by the TAUJnTOE.TAUJnTOEm bit.

The TAUJnTOm setting is written only to the bit (channel) set with TAUJnTOE.TAUJnTOEm bit = 0 when a write to the TAUJnTO register is attempted. No TAUJnTOm setting is reflected to the bit (channel) set with TAUJnTOE.TAUJnTOEm bit = 1.

NOTE

TAUJnTO.TAUJnTOm bit is placed so that its bit number corresponds to a channel number.

Output logic

Positive or negative logic of the output is specified by control bit TAUJnTOL.TAUJnTOLm.

The value of TAUJnTOL.TAUJnTOLm bit should be set before the counter is started. It can only be changed during operation with PWM output function. Otherwise, changes to TAUJnTOL.TAUJnTOLm result in an invalid TAUJTTOUTm signal output.

See **Section 19.4.3, Simultaneous Rewrite**.

Table 19.31 lists the various channel output modes and the channel output control bits.

Table 19.31 Channel Output Modes

Channel Output Mode	TAUJnTOE.TAUJnTOEm	TAUJnTOM.TAUJnTOMm
By software		
Independent channel output mode controlled by software	0	X
By TAUJ signals, independently		
Independent channel output mode 1	1	0
By TAUJ signals, synchronously		
Synchronous channel output mode 1	1	1

CAUTIONS

1. All combinations not listed in this table are forbidden.
2. Bits marked with an x can be set to any value.
3. The following bits cannot be changed during counter operation (TAUJnTE.TAUJnTEm = 1):
   - TAUJnTOM.TAUJnTOMm
   - TAUJnTOC.TAUJnTOCm

19.4.4.1 General Procedures for Specifying a Channel Output Mode

This section describes the general procedures for specifying a TAUJTTOUTm channel output mode. The prerequisite is that timer output operation is disabled (TAUJnTOE.TAUJnTOEm = 0).

(1) Set TAUJnTO.TAUJnTOm to specify the initial level of the TAUJTTOUTm output.

(2) Set channel output mode according to **Table 19.31, Channel Output Modes**, and the output logic using the TAUJnTOL.TAUJnTOLm bit.

(3) Start the counter (TAUJnTS.TAUJnTSm = 1).

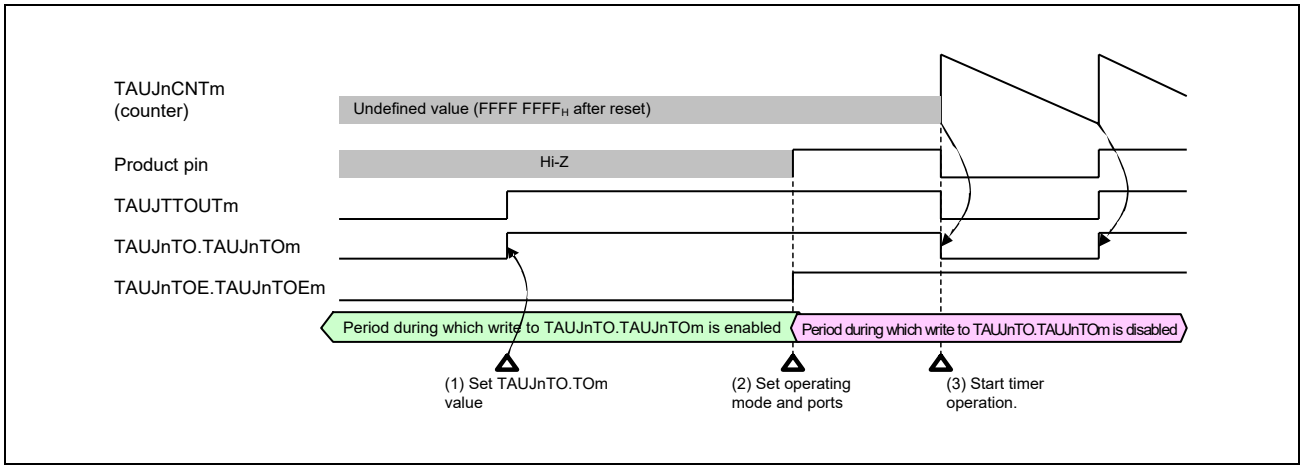


Figure 19.6 General Procedure for Specifying a TAUJTTOUTm Channel Output Mode

19.4.4.2 Channel Output Modes Controlled Independently by TAUJn Signals

This section lists the channel output modes that are controlled independently by TAUJn signals. The control bits used to specify a mode are listed in **Table 19.31, Channel Output Modes**.

(1) Independent channel output mode 1

Set/reset conditions

In this output mode, TAUJTTOUTm toggles when INTTAUJnIm is detected. The value of TAUJnTOL.TAUJnTOLm is ignored.

Prerequisites

There are no prerequisites other than those shown in **Table 19.31, Channel Output Modes**.

19.4.4.3 Channel Output Modes Controlled Synchronously by TAUJn Signals

This section lists the channel output modes that are controlled synchronously by TAUJn signals. The control bits used to specify a mode are listed in **Table 19.31, Channel Output Modes**.

(1) Synchronous channel output mode 1

Set/reset conditions

In this output mode, INTTAUJnIm of master channel serves as a set signal and INTTAUJnIm of the slave channel as a reset signal. If INTTAUJnIm of master channel and INTTAUJnIm of the slave channel are generated at the same time, INTTAUJnIm of the slave channel (reset signal) has priority over INTTAUJnIm (set signal) of master channel, i.e., the master channel is ignored.

Prerequisites

There are no prerequisites other than those shown in **Table 19.31, Channel Output Modes**.

19.4.5 Start Timing in Each Operating Mode

This section describes the timing at which the counter starts after TAUJnTS.TAUJnTSm is set to 1 in each operating mode.

In all modes, the value of data register (TAUJnCDRm) and whether or not an interrupt occurs depends on mode and register settings.

CAUTION

The count start timing described in this section is for your reference. Actually, the count start timing depends on the count clock timing.

19.4.5.1 Interval Timer Mode, Capture Mode, and Count Capture Mode

The counter starts operating with the next count clock after TAUJnTS.TAUJnTSm is set to 1. The value of data register is also loaded when the counter starts.

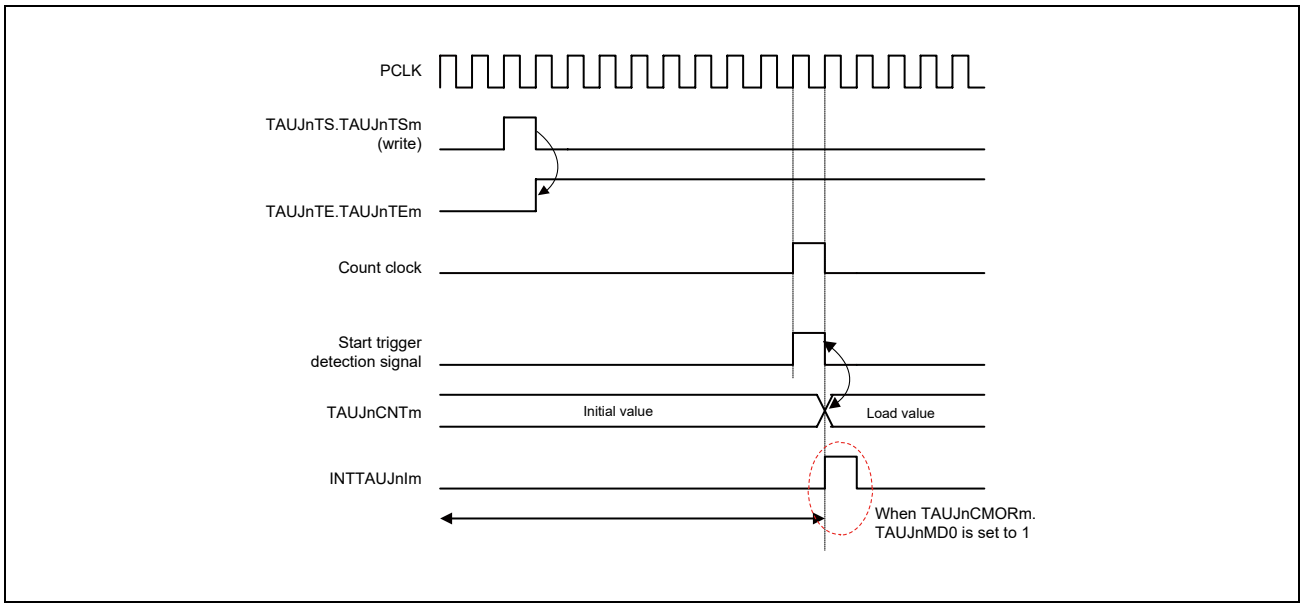


Figure 19.7 Start Timing in Interval Timer Mode, Capture Mode, and Count Capture Mode

19.4.5.2 Other Operating Modes

In other operating modes, the counter operation start timing is triggered only upon detection of an effective edge of TAUJTTINm. Once the counter starts, the value of data register is also loaded. The count clock cycles, which is irrelevant to start of counter operation, determine the frequency with which all operations take place.

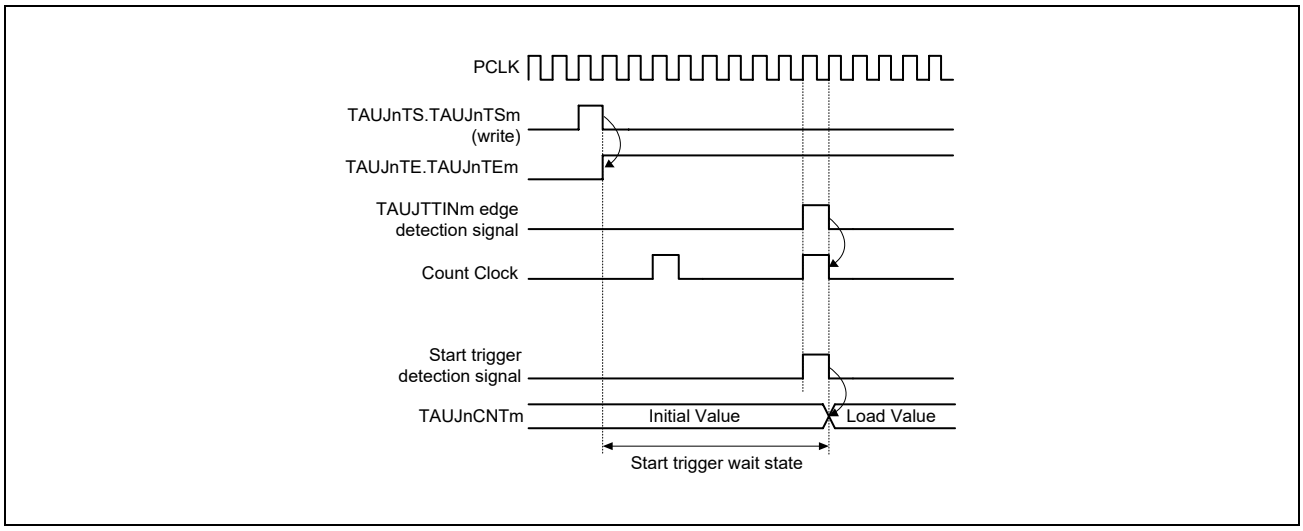


Figure 19.8 Start Timing in Other Operating Modes

19.4.6 TAUJTTOUTm Output and INTTAUJnIm Generation when Counter Starts or Restarts

When the counter starts, it is possible to specify whether an INTTAUJnIm is generated using the TAUJnCMORm.TAUJnMD0 bit. The effect of the bit depends on the selected mode, as shown in the table below. The effect of INTTAUJnIm on TAUJTTOUTm depends on the selected channel operation function.

Table 19.32 Effect of TAUJnCMORm.TAUJnMD0 Bit on Generation of INTTAUJnIm when Counter is Triggered

Mode	TAUJnCMORm.TAUJnMD0 Bit	INTTAUJnIm Generated when Counter Starts or Restarts or when TAUJTTINm Input Signal Trigger is Detected
Interval timer mode Capture mode Count capture mode	0	No
	1	Yes
Capture and one-count mode Capture and gate count mode	0	No
One-count mode	0/1	No, regardless of setting of TAUJnCMORm.TAUJnMD0 bit.

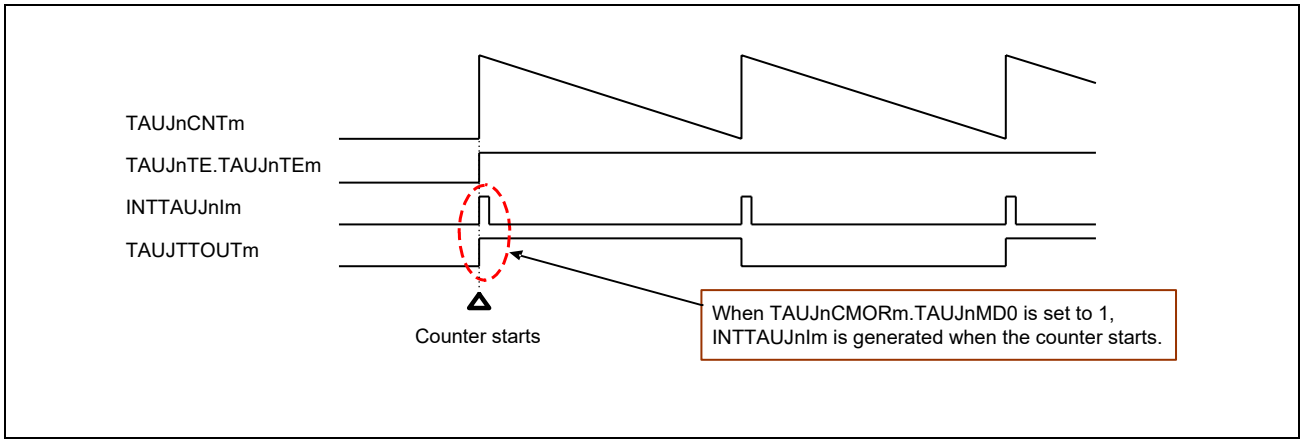


Figure 19.9 INTTAUJnIm Generated when Counter Starts

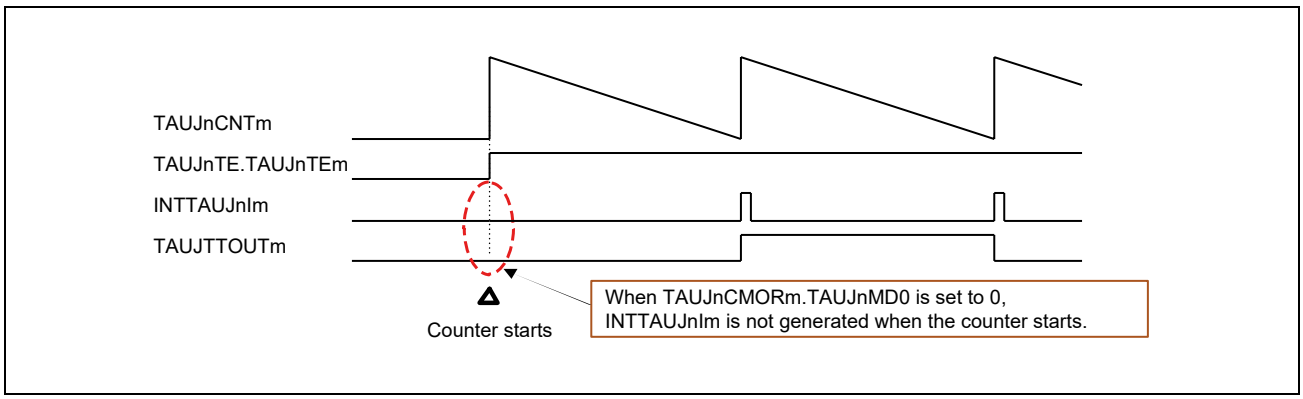


Figure 19.10 INTTAUJnIm Not Generated when Counter Starts

19.4.7 Interrupt Generation upon Overflow

With some independent functions, an interrupt will not be generated when the up counter overflows by reaching FFFF FFFF$_H$. This section describes how it is possible to generate an interrupt, by combining a channel operating in one of these modes with a channel in a different operation mode which counts down.

The appropriate operation mode for the second channel depends on the operation mode of the first channel. Nevertheless, the principle is the same for all combinations:

- Find an operating mode for the second channel that counts down in such a manner, that it reaches 0000 0000$_H$ at the same time as the first channel overflows (TAUJnCNTm = FFFF FFFF$_H$).
- Set TAUJnCDRm of the second channel to FFFF FFFF$_H$.
- The two channels must count at the same speed (i.e. they must have the same count clock).
- Both channels are triggered by the same TAUJTTINm input.
- The trigger detection settings (TAUJnCMORm.TAUJnSTS[2:0] and TAUJnCMURm.TAUJnTIS[1:0]) must be identical for both channels.

Result:

The down-counter of the second channel reaches 0000 0000$_H$ at exactly the same time as the up-counter of the first channel overflows (TAUJnCNTm = FFFF FFFF$_H$). Thus the second channel generates the desired interrupt.

The following sections list the operating modes that count down that are required to match specific operating modes that count up, as well as example timing diagrams.

19.4.7.1 Combination of the TAUJTTINm Input Position Detection Function and the Interval Timer Function

When the capture trigger is input simultaneously to TAUJTTINm of both channels, INTTAUJnIm of the interval timer function can detect the overflow when TAUDJnCNTm of the TAUJTTINm input position detection function exceeds FFFF FFFF$_H$.

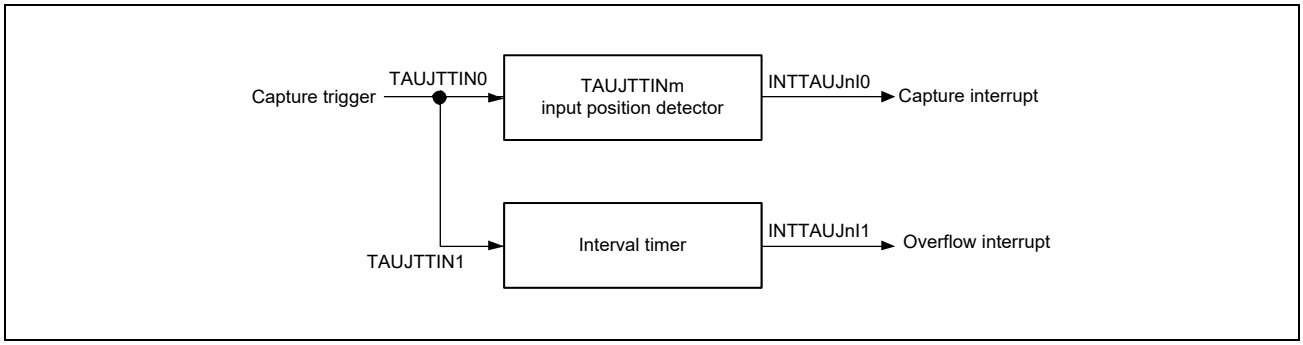


Figure 19.11 Combination of the TAUJTTINm Input Position Detection Function and the Interval Timer Function

Timing diagram

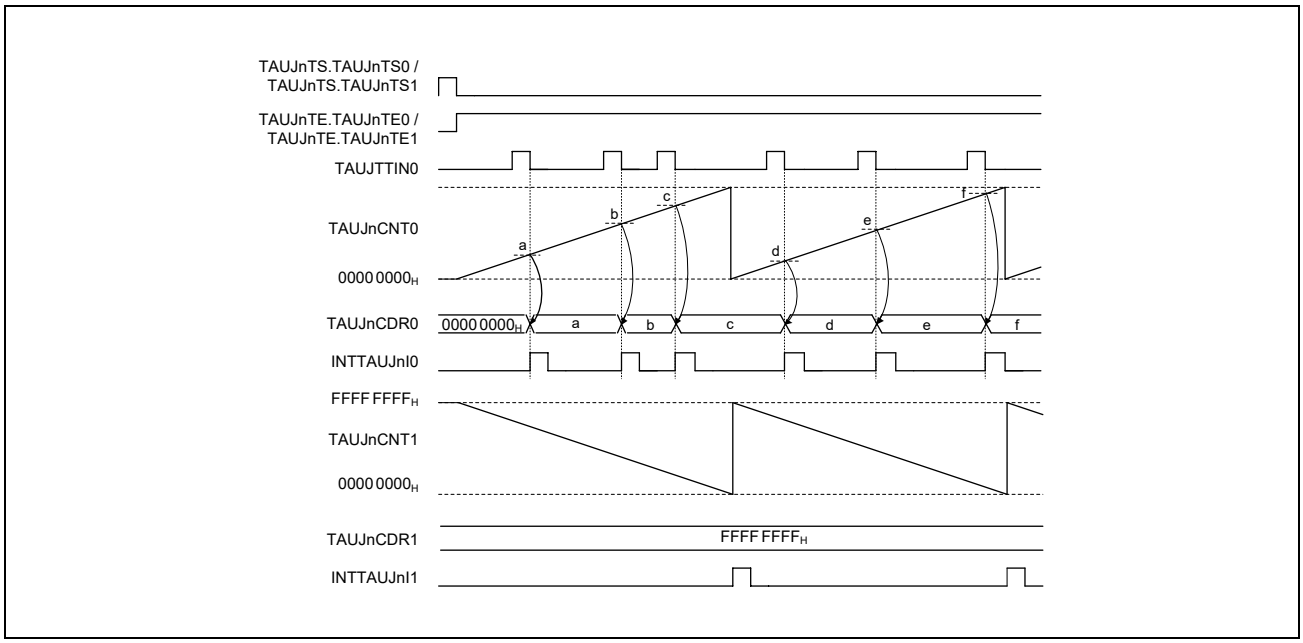


Figure 19.12 Interrupt Generation via Combination of the TAUJTTINm Input Position Detection Function and the Interval Timer Function

19.4.8 TAUJTTINm Edge Detection

Edge detection is based on the operation clock. This means that an edge can only be detected at the next rising edge of the operation clock. This can lead to a maximum delay of one operation clock cycle.

The following figure shows when edge detection takes place.

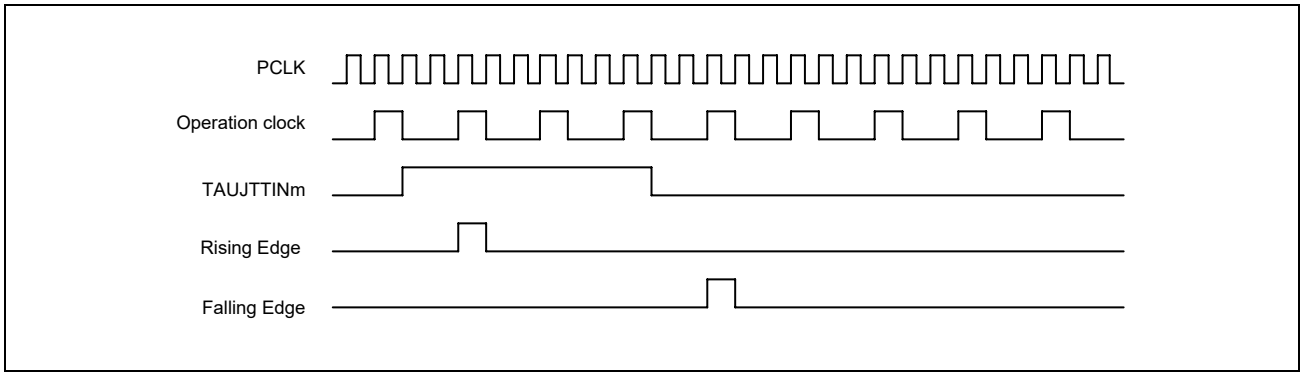


Figure 19.13 Basic Edge Detection Timing

CAUTION

Figure 19.13 shows an image of operation timing. In the actual operation, delay time occurs due to noise filter and synchronization circuit between the TAUJnIm pin and TAUJn.

19.4.9 Independent Channel Operation Functions

- **Section 19.4.9.1, Interval Timer Function**
- **Section 19.4.9.2, TAUJTTINm Input Interval Timer Function**
- **Section 19.4.9.3, TAUJTTINm Input Pulse Interval Measurement Function**
- **Section 19.4.9.4, TAUJTTINm Input Signal Width Measurement Function**
- **Section 19.4.9.5, TAUJTTINm Input Position Detection Function**
- **Section 19.4.9.6, TAUJTTINm Input Period Count Detection Function**

19.4.9.1 Interval Timer Function

(1) Overview

Summary

This function is used as a reference timer for generating timer interrupts (INTTAUJnIm) at regular intervals. When an interrupt is generated, the TAUJTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operating mode must be set to interval timer mode (see **Table 19.33, Contents of TAUJnCMORm Register for Interval Timer Function**.
- The channel output mode must be set to independent channel output mode 1. See **Section 19.4.4, Channel Output Modes**.

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The current value of TAUJnCDRm is written to TAUJnCNTm and the counter starts to count down from this value.

When the counter reaches 0000 0000_H, INTTAUJnIm is generated and the TAUJTTOUTm signal toggles. TAUJnCNTm then loads the TAUJnCDRm value and subsequently continues to operate.

The value of TAUJnCDRm can be rewritten at any time, and the changed value of TAUJnCDRm is applied the next time the counter starts to count down.

The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1, which in turn sets TAUJnTE.TAUJnTEm to 0. TAUJnCNTm and TAUJTTOUTm stop but retain their values. The counter can be restarted by setting TAUJnTS.TAUJnTSm to 1. The counter can also be forcibly restarted (without stopping it first) by setting TAUJnTS.TAUJnTSm to 1 during operation.

Conditions

If the TAUJnCMORm.TAUJnMD0 bit is set to 0, the first interrupt after a start or restart is not generated, and therefore TAUJTTOUTm does not toggle. This results in an inverted TAUJTTOUTm signal compared to when TAUJnCMORm.TAUJnMD0 is set to 1. For details refer to **Section 19.4.6, TAUJTTOUTm Output and INTTAUJnIm Generation when Counter Starts or Restarts**.

(2) Equations

INTTAUJnIm cycle = count clock cycle × (TAUJnCDRm + 1)

TAUJTTOUTm square wave cycle = count clock cycle × (TAUJnCDRm + 1) × 2

(3) Block diagram and general timing diagram

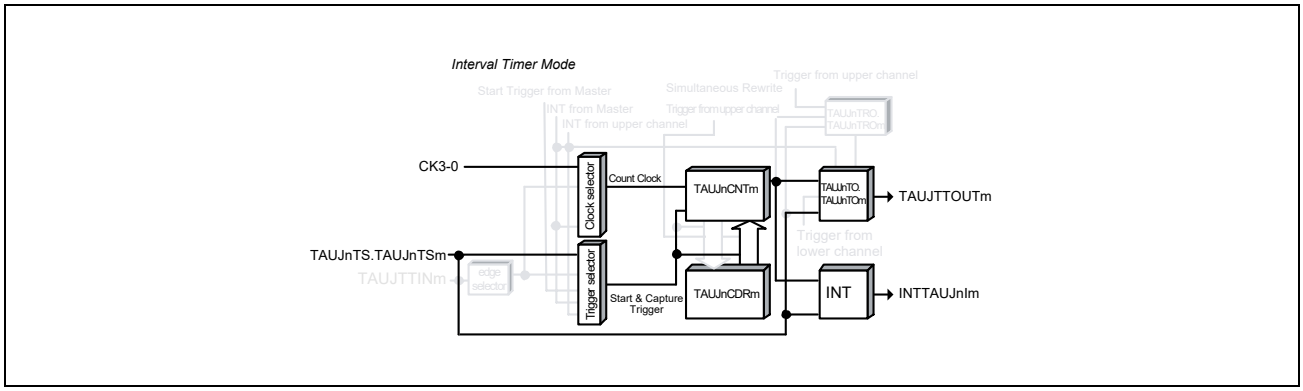


Figure 19.14 Block Diagram for Interval Timer Function

The following settings apply to the general timing diagram:

- INTTAUJnIm is generated at operation start (TAUJnCMORm.(TAUJnMD0 = 1)

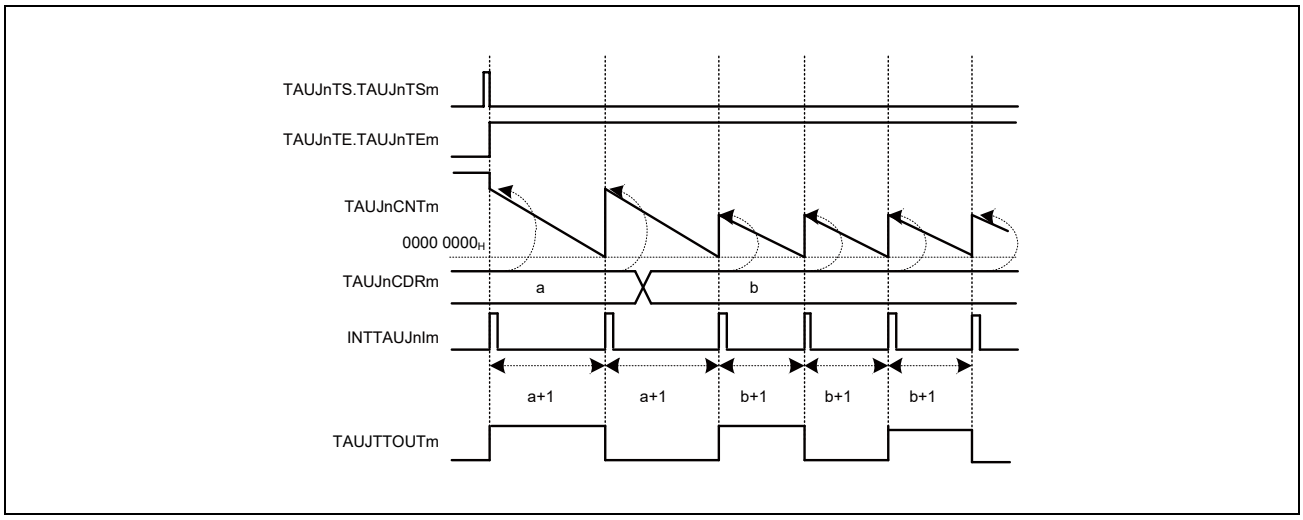


Figure 19.15 General Timing Diagram for Interval Timer Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.33 Contents of TAUJnCMORm Register for Interval Timer Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	0: Not used, so set to 0
10 to 8	TAUJnSTS[2:0]	000: A software trigger is used as a start trigger.
7, 6	TAUJnCOS[1:0]	00: Not used, so set to 00
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0000: Interval timer mode
0	TAUJnMD0	0: INTTAUJnIm is not generated to toggle TAUJTTOUTm at the start of operation. 1: INTTAUJnIm is generated to toggle TAUJTTOUTm at the start or restart of operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.34 Contents of TAUJnCMURm Register for Interval Timer Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Not used, so set to 00

(c) Channel output mode

Table 19.35 Control Bit Settings for Independent Channel Output Mode 1

Bit Name	Setting
TAUJnTOE.TAUJnTOEm	1: Enables independent channel output mode
TAUJnTO.TAUJnTOm	0: Low level 1: High level
TAUJnTOM.TAUJnTOMm	0: Independent channel output
TAUJnTOC.TAUJnTOCm	0: Toggle mode
TAUJnTOL.TAUJnTOLm	0: Positive logic

NOTE

The channel output mode can also be set to Channel Output Mode Controlled by Software by setting TAUJnTOE.TAUJnTOEm = 0. TAUJTTOUTm can then be controlled independently of the interrupts. For details refer to **Section 19.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the Interval Timer Function. Therefore, these registers must be set to 0.

Table 19.36 Simultaneous Rewrite Settings for Interval Timer Function

Bit Name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm = 0), set these bits to 0

(5) Operating procedure for interval timer function

Table 19.37 Operating Procedure for Interval Timer Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 19.33, Contents of TAUJnCMORm Register for Interval Timer Function** and **Table 19.34, Contents of TAUJnCMURm Register for Interval Timer Function**. Set the value of the TAUJnCDRm register. Set the channel output mode by setting the control bits as described in **Table 19.35, Control Bit Settings for Independent Channel Output Mode 1**.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. • TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and the counter starts. TAUJnCNTm loads the TAUJnCDRm value. • When TAUJnCMORm.TAUJnMD0 = 1, INTTAUJnIm is generated and TAUJTTOUTm toggles.
During operation	The TAUJnCDRm register value can be changed at any time. The TAUJnCNTm register can be read at all times.	TAUJnCNTm counts down. When the counter reaches 0000 0000_H: • TAUJnCNTm reloads the TAUJnCDRm value and continues count operation • INTTAUJnIm is generated and TAUJTTOUTm toggles.
Stop operation	Set TAUJnTT.TAUJnTTm to 1. • TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. • TAUJnCNTm and TAUJTTOUTm stop and retain their current values.

Restart

(6) Specific timing diagrams

(a) TAUJnCDRm = 0000 0000H, count clock = PCLK/2

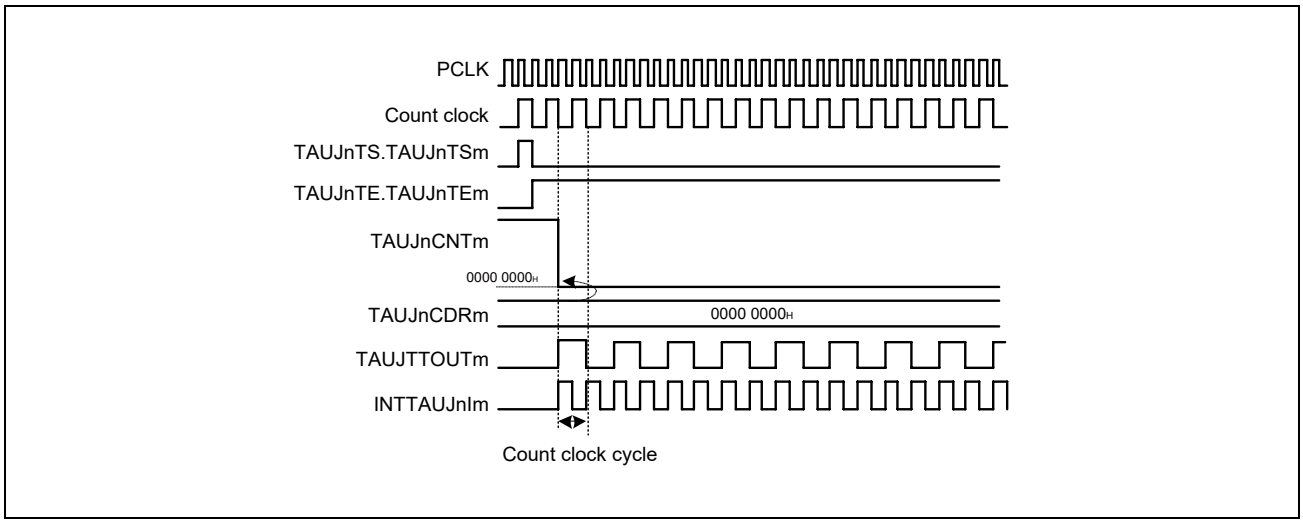


Figure 19.16 TAUJnCDRm = 0000 0000H, Count Clock = PCLK/2

- If TAUJnCDRm = 0000 0000H and the count clock = PCLK/2, the TAUJnCDRm value is written to TAUJnCNTm every count clock, meaning that TAUJnCNTm is always 0000 0000H.
- INTTAUJnIm is generated every count clock, resulting in TAUJTTOUTm toggling every count clock.

(b) TAUJnCDRm = 0000 0000H, count clock = PCLK

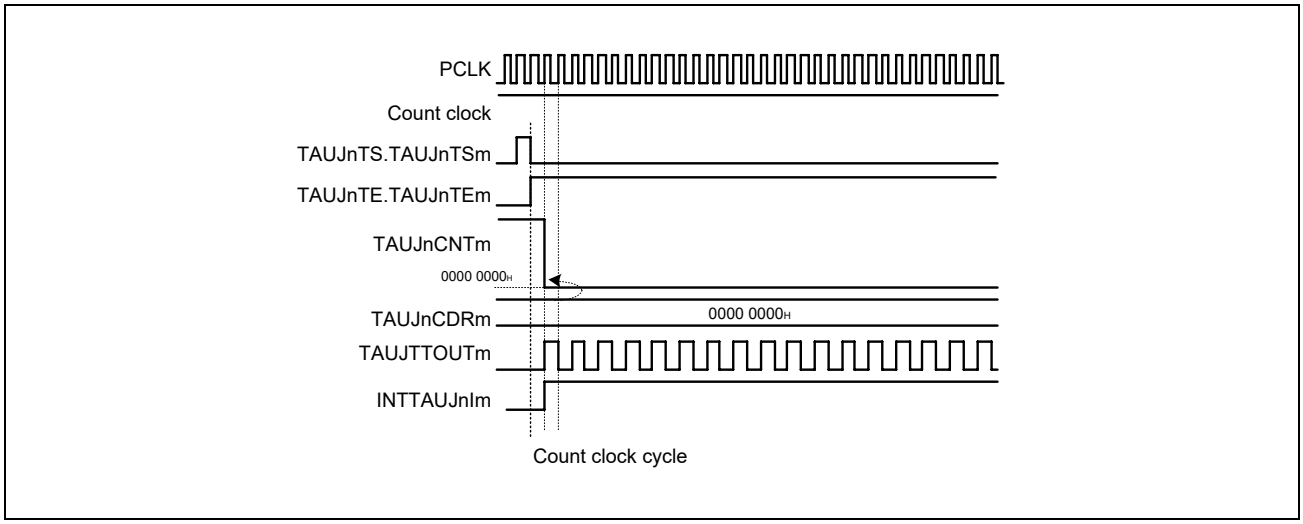


Figure 19.17 TAUJnCDRm = 0000 0000H, Count Clock = PCLK

- If TAUJnCDRm = 0000 0000H and the count clock = PCLK, the TAUJnCDRm value is written to TAUJnCNTm every PCLK clock, meaning that TAUJnCNTm is always 0000 0000H.
- INTTAUJnIm is generated continuously, resulting in TAUJTTOUTm toggling every PCLK clock.

(c) Operation stop and restart

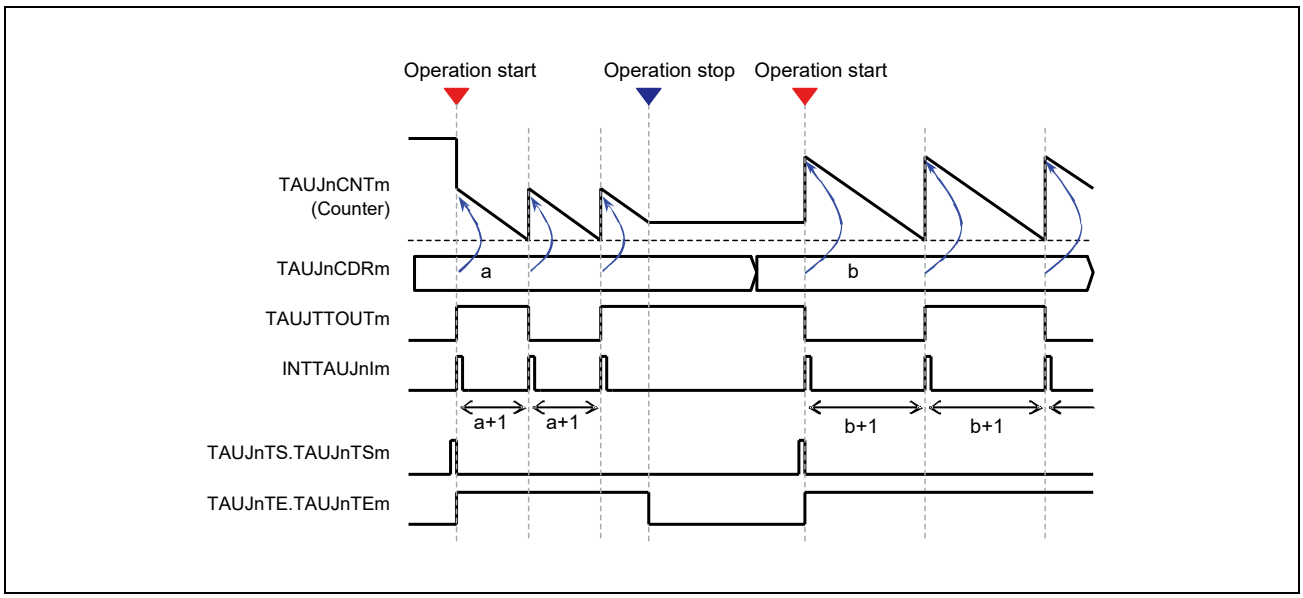


Figure 19.18 Operation Stop and Restart, TAUJnCMORm.TAUJnMD0 = 1

- The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1, which in turn sets TAUJnTE.TAUJnTEm to 0.
- TAUJnCNTm and TAUJTTOUTm stop but retain their values.
- The counter can be restarted by setting TAUJnTS.TAUJnTSm to 1.

(d) Forced restart

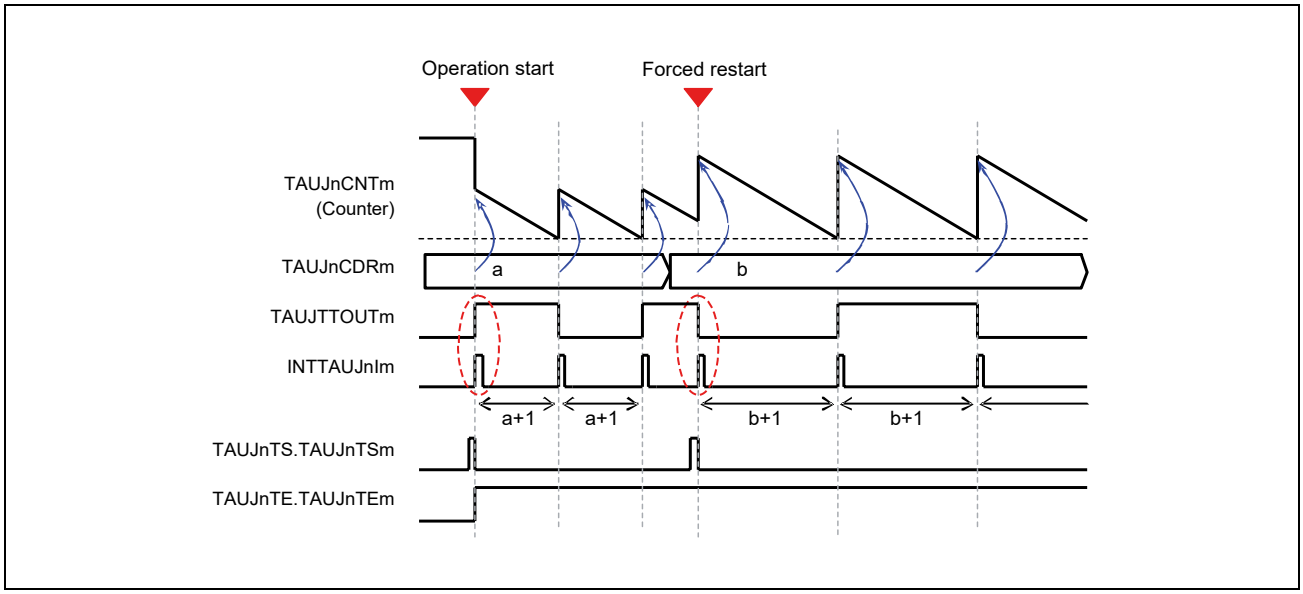


Figure 19.19 Forced Restart Operation, TAUJnCMORm.TAUJnMD0 = 1

- The counter can be forcibly restarted (without stopping it first) by setting TAUJnTS.TAUJnTSm to 1 during operation.
- If the TAUJnCMORm.TAUJnMD0 bit is set to 1, the first interrupt after a start or restart is generated.

19.4.9.2 TAUJTTINm Input Interval Timer Function

(1) Overview

Summary

This function is used as a reference timer for generating timer interrupts (INTTAUJnIm) at regular intervals or when an effective TAUJTTINm input edge is detected. When an interrupt is generated, the TAUJTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operating mode must be set to interval timer mode (see **Table 19.15, TAUJnCMORm Register Contents**).
- The channel output mode must be set to independent channel output mode 1. See **Table 19.31, Channel Output Modes**.

Descriptions

This function operates in an identical manner to the Interval Timer Function (see **Section 19.4.9.1, Interval Timer Function**), except that this function is restarted by an effective TAUJTTINm input edge. The type of edge used as the trigger is specified using the TAUJnCMURm.TAUJnTIS[1:0] bits. Either rising edge, falling edge, or rising and falling edge can be selected.

(2) Equations

INTTAUJnIm cycle = count clock cycle × (TAUJnCDRm + 1)

TAUJTTOUTm square wave cycle = count clock cycle × (TAUJnCDRm + 1) × 2

(3) Block diagram and general timing diagram

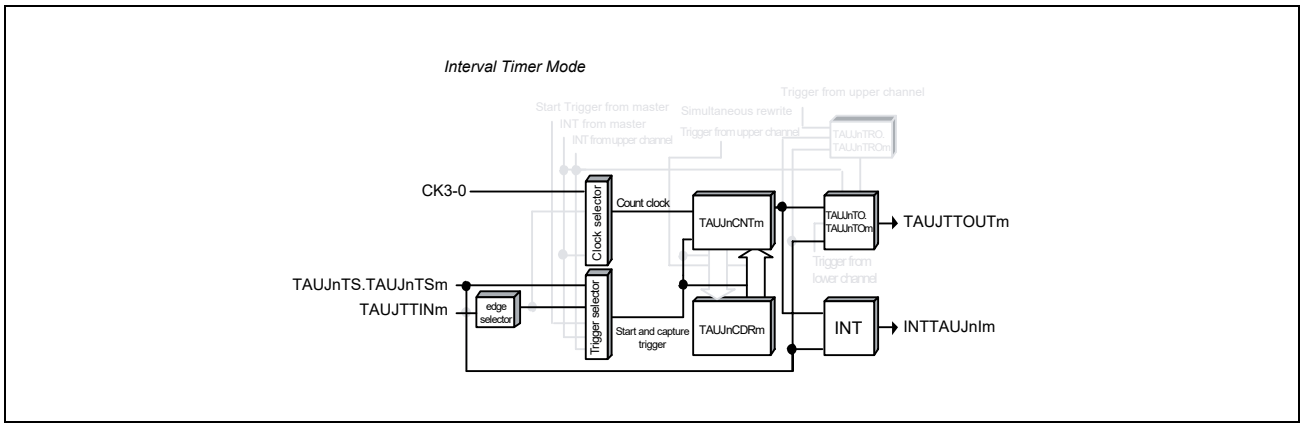


Figure 19.20 Block Diagram for TAUJTTINm Input Interval Timer Function

The following settings apply to the general timing diagram:

- INTTAUJnIm is generated at operation start (TAUJnCMORm.TAUJnMD0 = 1)
- Rising edge detection (TAUJnCMURm.TAUJnTIS[1:0] = 01_B)

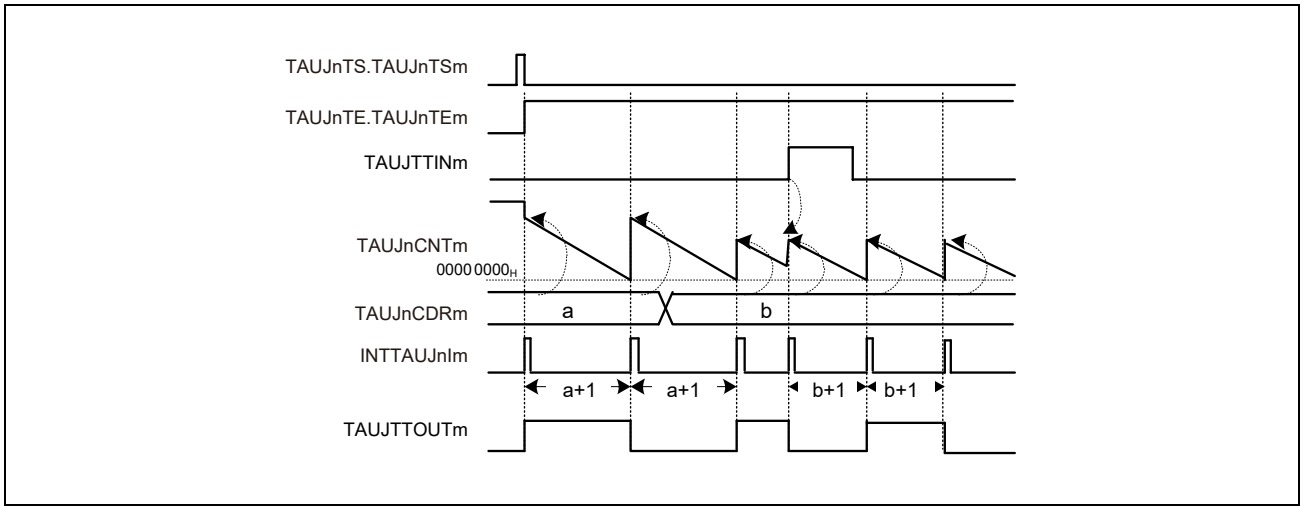


Figure 19.21 General Timing Diagram for TAUJTTINm Input Interval Timer Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.38 Contents of TAUJnCMORm Register for TAUJTTINm Input Interval Timer Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	0: Not used, so set to 0
10 to 8	TAUJnSTS[2:0]	001: Effective TAUJTTINm input edge signal is used as an external start trigger.
7, 6	TAUJnCOS[1:0]	00: Not used, so set to 00
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0000: Interval timer mode
0	TAUJnMD0	0: INTTAUJnIm is not generated to toggle TAUJTTOUTm at the start of operation. 1: INTTAUJnIm is generated to toggle TAUJTTOUTm at the start of operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.39 Contents of TAUJnCMURm Register for TAUJTTINm Input Interval Timer Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Falling edge detection 01: Rising edge detection 10: Rising and falling edge detection

(c) Channel output mode

Table 19.40 Control Bit Settings for Independent Channel Output Mode 1

Bit Name	Setting
TAUJnTOE.TAUJnTOEm	1: Enables independent channel output mode
TAUJnTO.TAUJnTOm	0: Low level 1: High level
TAUJnTOM.TAUJnTOMm	0: Independent channel output
TAUJnTOC.TAUJnTOCm	0: Operating mode 1 (toggle mode if TAUJnTOM.TAUJnTOMm = 0)
TAUJnTOL.TAUJnTOLm	0: Positive logic

NOTE

The channel output mode can also be set to Channel Output Mode Controlled by Software by setting TAUJnTOE.TAUJnTOEm = 0. TAUJTTOUTm can then be controlled independently of the interrupts. For details refer to **Section 19.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Interval Timer Function. Therefore, these registers must be set to 0.

Table 19.41 Simultaneous Rewrite Settings for TAUJTTINm Input Interval Timer Function

Bit Name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm = 0), set these bits to 0

(5) Operating procedure for TAUJTTINm input interval timer function

Table 19.42 Operating Procedure for TAUJTTINm Input Interval Timer Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 19.38, Contents of TAUJnCMORm Register for TAUJTTINm Input Interval Timer Function** and **Table 19.39, Contents of TAUJnCMURm Register for TAUJTTINm Input Interval Timer Function**. Set the value of the TAUJnCDRm register Set the channel output mode by setting the control bits as described in **Table 19.40, Control Bit Settings for Independent Channel Output Mode 1**.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. • TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and the counter starts. TAUJnCNTm loads the TAUJnCDRm value. • When TAUJnCMORm.TAUJnMD0 = 1, INTTAUJnIm is generated and TAUJTTOUTm toggles.
During operation	The values of the TAUJnCMURm.TAUJnTIS[1:0] and TAUJnCDRm registers can be changed at any time. The TAUJnCNTm register can be read at all times. Detection of TAUJTTINm edge	TAUJnCNTm counts down. When the counter reaches 0000 0000_H: • TAUJnCNTm reloads the TAUJnCDRm value and continues count operation • INTTAUJnIm is generated and TAUJTTOUTm toggles When an effective TAUJTTINm input edge is detected during count operation, TAUJnCNTm reloads the TAUJnCDRm value and continues count operation. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TAUJnTTm to 1. • TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. • TAUJnCNTm and TAUJTTOUTm stop and retain their current values.

Restart

(6) Specific timing diagrams

The timing diagrams in **Section 19.4.9.1, Interval Timer Function** apply, and in addition the counter can also be restarted by an effective TAUJTTINm input edge.

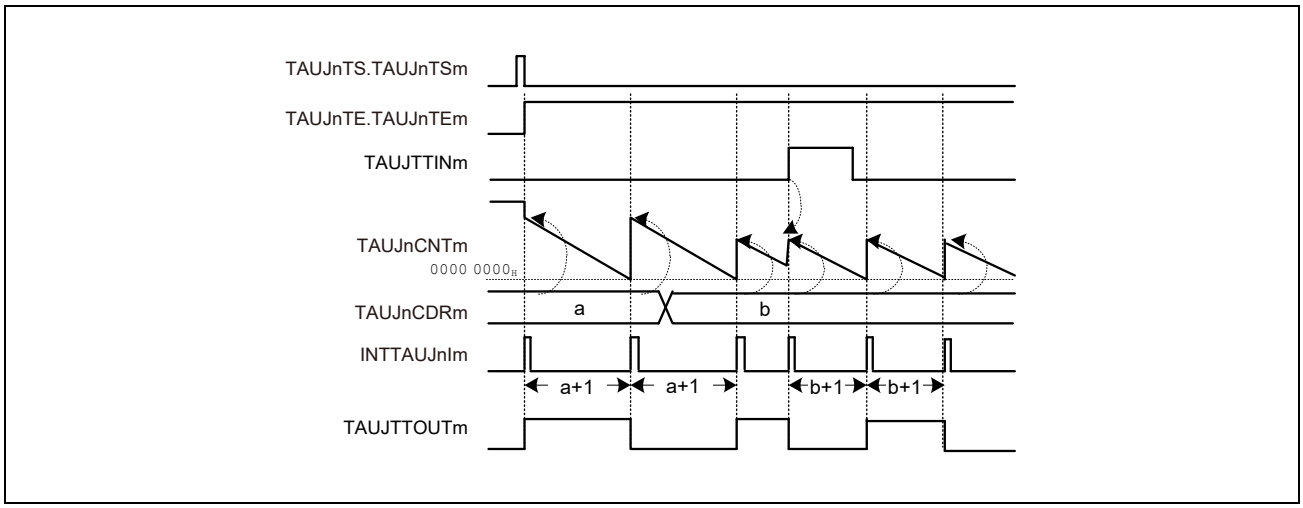


Figure 19.22 Counter Triggered by Rising TAUJTTINm Input Edge (TAUJnCMURm.TAUJnTIS[1:0] = 01_B), TAUJnCMORm.TAUJnMD0 = 1

If an effective TAUJTTINm input edge is detected, an interrupt is generated which causes TAUJTTOUTm to toggle. In this example, the effective edge is a rising edge (TAUJnCMURm.TAUJnTIS[1:0] = 01_B).

19.4.9.3 TAUJTTINm Input Pulse Interval Measurement Function

(1) Overview

Summary

This function captures the count value and uses this value and the overflow bit TAUJnCSRm.TAUJnOVF to measure the interval of the TAUJTTINm input signal.

Prerequisites

- The operating mode must be set to capture mode (see **Table 19.44, Contents of TAUJnCMORm Register for TAUJTTINm Input Pulse Interval Measurement Function**).
- TAUJTTOUTm is not used with this function.

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The counter TAUJnCNTm starts counting up from 0000 0000_H. When an effective TAUJTTINm edge is detected, the value of TAUJnCNTm is captured, transferred to TAUJnCDRm, and an interrupt INTTAUJnIm is generated. The counter resets to 0000 0000_H and subsequently continues to operate.

If the counter reaches FFFF $FFFF_H$ before an effective TAUJTTINm edge is detected, it overflows to 0000 0000_H. The counter is reset to 0000 0000_H and subsequently continues to operate. The values transferred to TAUJnCDRm and TAUJnCSRm.TAUJnOVF respectively depend on the values of bits TAUJnCMORm.TAUJnCOS[1:0]:

Table 19.43 Effects of an Overflow

TAUJnCMORm. TAUJnCOS[1:0]	When Overflow Occurs		When an Effective TAUJTTINm Input is then Detected	
	TAUJnCDRm	TAUJnCSRm.TAUJnOVF	TAUJnCDRm, TAUJnCNTm	TAUJnCSRm.TAUJnOVF
00	Unchanged	0	TAUJnCNTm written to TAUJnCDRm	1
01		1		
10	Set to FFFF $FFFF_H$	0	TAUJnCNTm set to 0, TAUJnCDRm unchanged	Unchanged
11		1		

If an overflow is set (TAUJnCSRm.TAUJnOVF = 1) when TAUJnCMORm.TAUJnCOS[0] = 1, it can only be cleared by setting TAUJnCSCm.TAUJnCLOV = 1.

The combination of the value of TAUJnCDRm and TAUJnCSRm.TAUJnOVF can be used to deduce the interval of the TAUJTTINm signal. However, if an overflow occurs multiple times before an effective TAUJTTINm input is detected, the overflow bit TAUJnCSRm.TAUJnOVF cannot indicate this.

The function can be stopped by setting TAUJnTT.TAUJnTTm = 1, which in turn sets TAUJnTE.TAUJnTEm = 0. TAUJnCNTm stops but retains its value. While the function is stopped, effective TAUJTTINm input edge detection and TAUJnCNTm capture are not performed.

The counter is reset to 0000 0000_H and subsequently continues to operate.

Conditions

If the TAUJnCMORm.TAUJnMD0 bit is set to 0, the first interrupt after a start or restart is not generated. For details refer to **Section 19.4.6, TAUJTTOUTm Output and INTTAUJnIm Generation when Counter Starts or Restarts**.

NOTE

When TAUJnCMORm.TAUJnCOS[1:0] = 1, the value of TAUJnCNTm is not written to TAUJnCDRm when the first effective TAUJTTINm input edge occurs after an overflow. However, an interrupt is generated.

(2) Equations

TAUJTTINm input pulse interval = count clock cycle ×
[(TAUJnCSRm.TAUJnOVF × (FFFF FFFF$_H$ + 1)) + TAUJnCDRm capture value + 1]

(3) Block diagram and general timing diagram

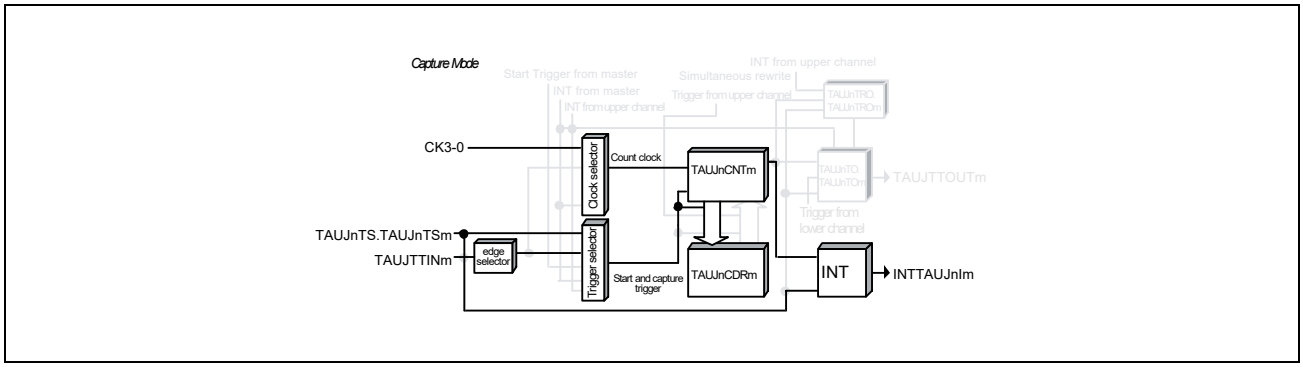


Figure 19.23 Block Diagram for TAUJTTINm Input Pulse Interval Measurement Function

The following settings apply to the general timing diagram:

- INTTAUJnIm not generated at operation start (TAUJnCMORm.TAUJnMD0 = 0)
- Falling edge detection (TAUJnCMURm.TAUJnTIS[1:0] = 00$_B$)
- When an effective TAUJTTINm input is detected after an overflow, TAUJnCDRm is changed and TAUJnCSRm.TAUJnOVF is set to 1 (TAUJnCMORm.TAUJnCOS[1:0] = 00$_B$)

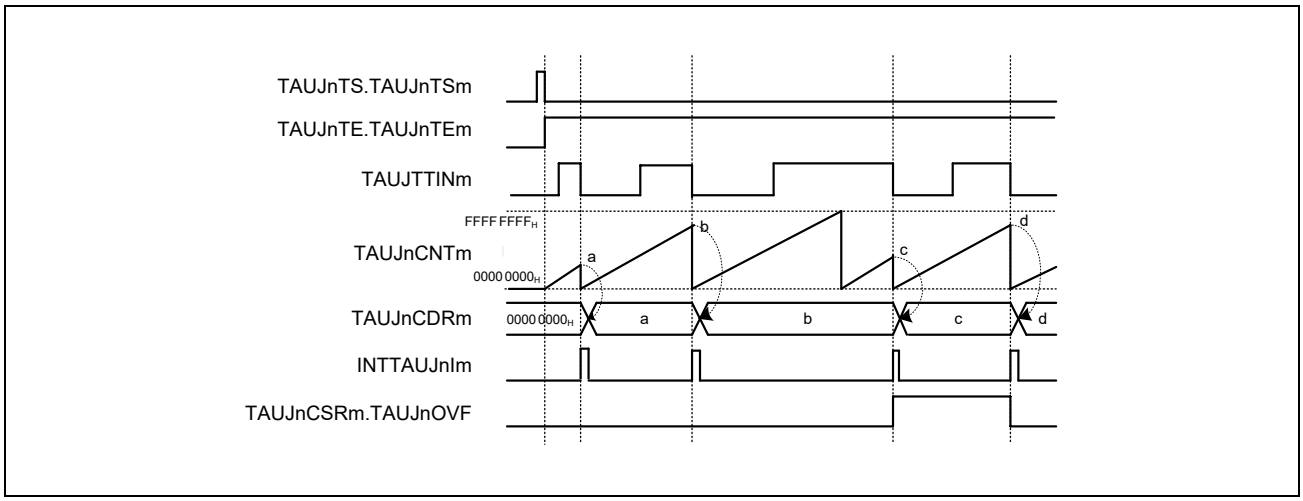


Figure 19.24 General Timing Diagram for TAUJTTINm Input Pulse Interval Measurement Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.44 Contents of TAUJnCMORm Register for TAUJTTINm Input Pulse Interval Measurement Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	0: Not used, so set to 0
10 to 8	TAUJnSTS[2:0]	001: Effective edge of the TAUJTTINm input signal is used as an external capture trigger.
7, 6	TAUJnCOS[1:0]	See **Table 19.43, Effects of an Overflow**.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0010: Capture mode
0	TAUJnMD0	0: INTTAUJnIm is not generated at the start operation. 1: INTTAUJnIm is generated at the start operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.45 Contents of TAUJnCMURm Register for TAUJTTINm Input Pulse Interval Measurement Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Falling edge detection 01: Rising edge detection 10: Rising and falling edge detection

(c) Channel output mode

TAUJnTOE.TAUJnTOEm is set to 0 because the channel output mode is not used by this function.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Pulse Interval Measurement Function. Therefore, these registers must be set to 0.

Table 19.46 Simultaneous Rewrite Settings for TAUJTTINm Input Pulse Interval Measurement Function

Bit Name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm = 0), set these bits to 0

(5) Operating procedure for TAUJTTINm input pulse interval measurement function

Table 19.47 Operating Procedure for TAUJTTINm Input Pulse Interval Measurement Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 19.44, Contents of TAUJnCMORm Register for TAUJTTINm Input Pulse Interval Measurement Function** and **Table 19.45, Contents of TAUJnCMURm Register for TAUJTTINm Input Pulse Interval Measurement Function**. The TAUJnCDRm register functions as a capture register.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. • TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and the counter starts. • TAUJnCNTm is cleared to 0000 0000_H. INTTAUJnIm is generated when TAUJnCMORm.TAUJnMD0 is set to 1.
During operation	Detection of TAUJTTINm edges. The TAUJnCMURm. TAUJnTIS[1:0] bits can be changed at any time. The TAUJnCDRm and TAUJnCSRm registers can be read at any time. TAUJnCSCm.TAUJnCLOV bit can be written to 1 (TAUJnCSRm.TAUJnOVF bit is cleared to 0).	TAUJnCNTm starts to count up from 0000 0000_H. When an effective TAUJTTINm edge is detected: • TAUJnCNTm transfers (captures) its value to TAUJnCDRm, and returns to 0000 0000_H • INTTAUJnIm is then generated. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TAUJnTTm to 1. • TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. • TAUJnCNTm stops and both it and TAUJnCSRm.TAUJnOVF retain their current values.

Restart

(6) Specific timing diagrams: overflow behavior

(a) TAUJnCMORm.TAUJnCOS[1:0] = 00_B

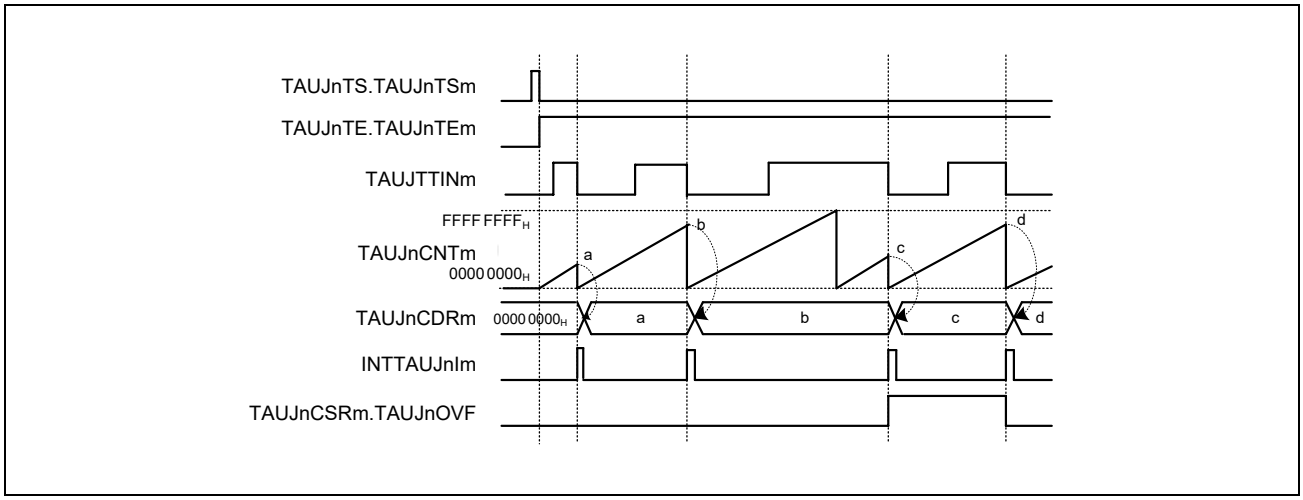


Figure 19.25 TAUJnCMORm.TAUJnCOS[1:0] = 00_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- When an overflow occurs, the value of TAUJnCDRm remains unchanged and TAUJnCSRm.TAUJnOVF remains = 0.
- Upon detection of the next effective TAUJTTINm input edge, the value of TAUJnCNTm is written to TAUJnCDRm and TAUJnCSRm.TAUJnOVF is set to 1.
- If the next effective TAUJTTINm input edge is detected when no overflow occurs, TAUJnCSRm.TAUJnOVF is cleared to 0.

(b) TAUJnCMORm.TAUJnCOS[1:0] = 01_B

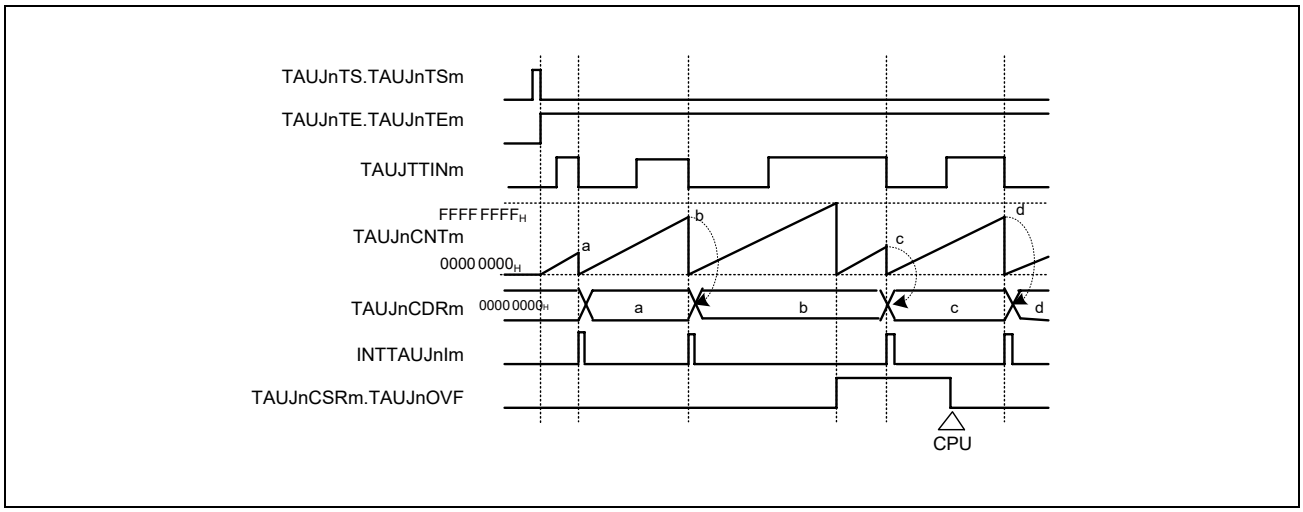


Figure 19.26 TAUJnCMORm.TAUJnCOS[1:0] = 01_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- When an overflow occurs, the value of TAUJnCDRm remains unchanged and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next effective TAUJTTINm input edge, the value of TAUJnCNTm is written to TAUJnCDRm.
- TAUJnCSRm.TAUJnOVF is only cleared by a CPU command (by setting the TAUJnCSCm.TAUJnCLOV bit to 1).

(c) TAUJnCMORm.TAUJnCOS[1:0] = 10_B

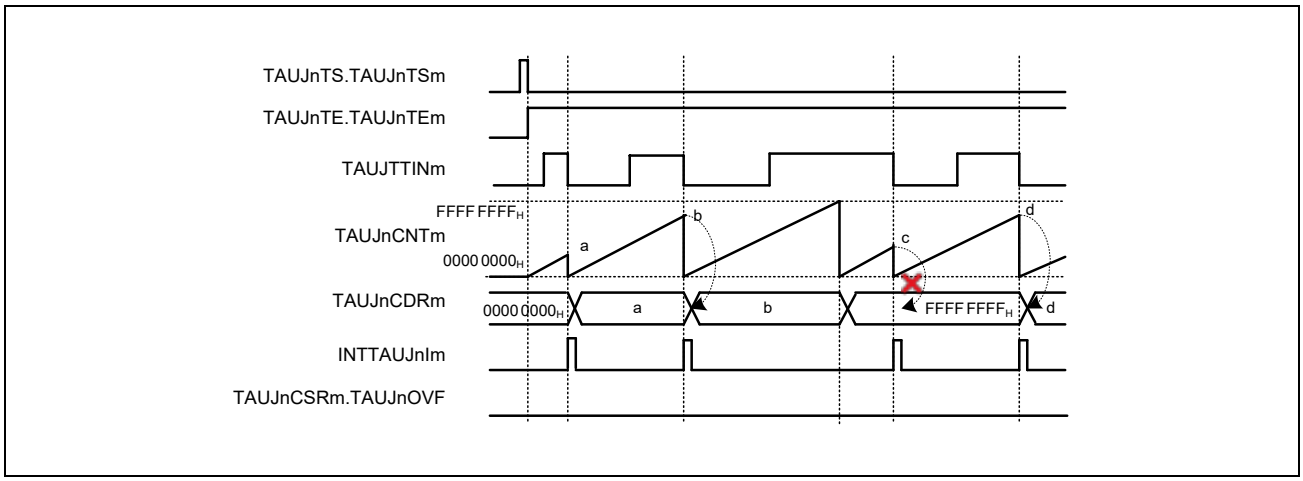


Figure 19.27 TAUJnCMORm.TAUJnCOS[1:0] = 10_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- When an overflow occurs, TAUJnCDRm is set to FFFF FFFF$_H$ and TAUJnCSRm.TAUJnOVF remains = 0.
- Upon detection of the next effective TAUJTTINm input edge, TAUJnCNTm is reset to 0, but TAUJnCDRm and TAUJnCSRm.TAUJnOVF remain unchanged.
- Thus, the next effective TAUJTTINm input edge after the overflow is ignored.

(d) TAUJnCMORm.TAUJnCOS[1:0] = 11_B

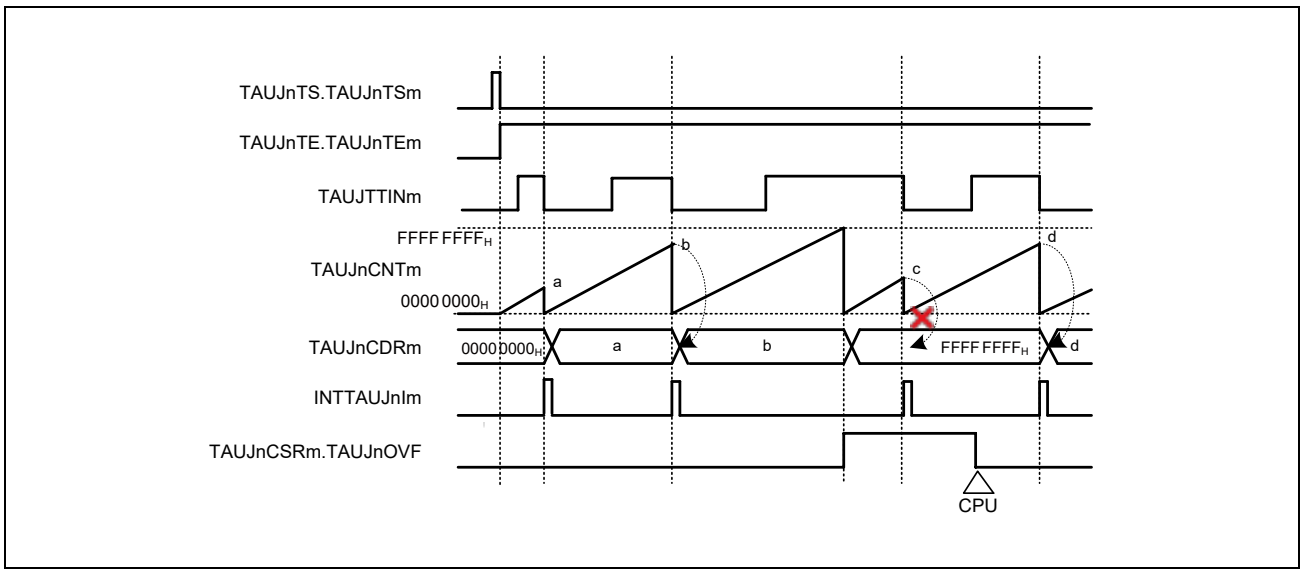


Figure 19.28 TAUJnCMORm.TAUJnCOS[1:0] = 11_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- When an overflow occurs, TAUJnCDRm is set to FFFF $FFFF_H$, and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next effective TAUJTTINm input edge, TAUJnCNTm is reset to 0, but TAUJnCDRm and TAUJnCSRm.TAUJnOVF remain unchanged.
- Thus, the next effective TAUJTTINm input edge after the overflow is ignored.
- TAUJnCSRm.TAUJnOVF is cleared by setting the TAUJnCSCm.TAUJnCLOV bit to 1.

19.4.9.4 TAUJTTINm Input Signal Width Measurement Function

(1) Overview

Summary

This function measures the width of a TAUJTTINm signal by starting counting on one edge of the TAUJTTINm signal and capturing the counter value on the opposite edge.

Prerequisites

- The operating mode should be set to capture and one-count mode (see **Table 19.49, Contents of TAUJnCMORm Register for TAUJTTINm Input Signal Width Measurement Function**).
- TAUJTTOUTm is not used with this function.
- TAUJnCMORm.TAUJnMD0 should be set to 0.

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. When an effective TAUJTTINm start edge is detected, the counter TAUJnCNTm starts counting up from 0000 0000_H. When an effective TAUJTTINm stop edge is detected, the value of TAUJnCNTm is captured, transferred to TAUJnCDRm, and an interrupt INTTAUJnIm is generated. The counter retains its value and awaits the next effective TAUJTTINm input start edge.

If the counter reaches FFFF $FFFF_H$ before an effective TAUJTTINm stop edge is detected, it overflows. The counter is reset to 0000 0000_H and subsequently continues to operate. The values transferred to TAUJnCDRm and TAUJnCSRm.TAUJnOVF respectively depend on the values of bits TAUJnCMORm.TAUJnCOS[1:0]:

Table 19.48 Effects of an Overflow

TAUJnCMORm.TAUJnCOS[1:0]	When Overflow Occurs		When an Effective TAUJTTINm Input Stop Edge is Detected	
	TAUJnCDRm	TAUJnCSRm.TAUJnOVF	TAUJnCDRm, TAUJnCNTm	TAUJnCSRm.TAUJnOVF
00	Unchanged	0	TAUJnCNTm written to TAUJnCDRm	1
01		1		
10	Set to FFFF $FFFF_H$	0	TAUJnCNTm stops counting TAUJnCDRm unchanged	Unchanged
11		1		

If an overflow is set (TAUJnCSRm.OVF = 1) when TAUJnCMORm.TAUJnCOS[0] = 1, it can only be cleared by setting TAUJnCSCm.TAUJnCLOV = 1.

The combination of the value of TAUJnCDRm and TAUJnCSRm.TAUJnOVF can be used to deduce the width of the TAUJTTINm signal. However, if an overflow occurs multiple times before an effective TAUJTTINm input is detected, the overflow bit TAUJnCSRm.TAUJnOVF cannot indicate this.

This function cannot be forcibly restarted.

NOTE

When TAUJnCMORm.TAUJnCOS[1] = 1, the value of TAUJnCNTm is not written to TAUJnCDRm when the first effective TAUJTTINm input edge occurs after an overflow. However, an interrupt is generated.

(2) Equations

TAUJTTINm input signal width = count clock cycle ×
[(TAUJnCSRm.TAUJnOVF × (FFFF FFFF$_H$ + 1)) + TAUJnCDRm capture value + 1]

(3) Block diagram and general timing diagram

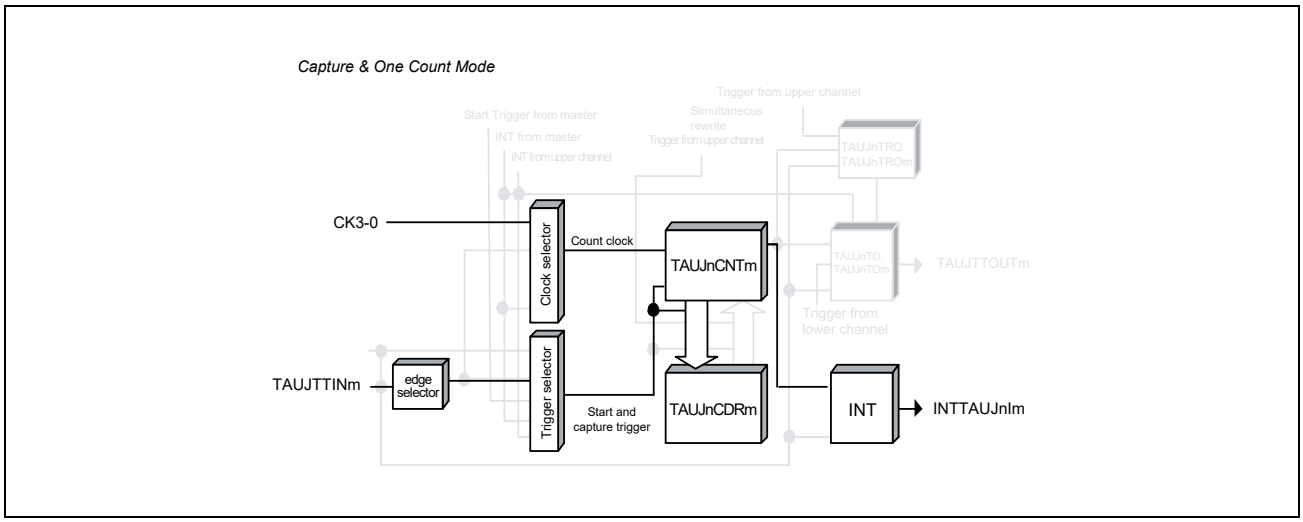


Figure 19.29 Block Diagram for TAUJTTINm Input Signal Width Measurement Function

The following settings apply to the general timing diagram:

- Rising and falling edge detection = high width measurement (TAUJnCMURm.TAUJnTIS[1:0] = 11_B)
- When an effective TAUJTTINm input is detected after an overflow, TAUJnCDRm is changed and TAUJnCSRm.TAUJnOVF is set to 1 (TAUJnCMORm.TAUJnCOS[1:0] = 00_B)

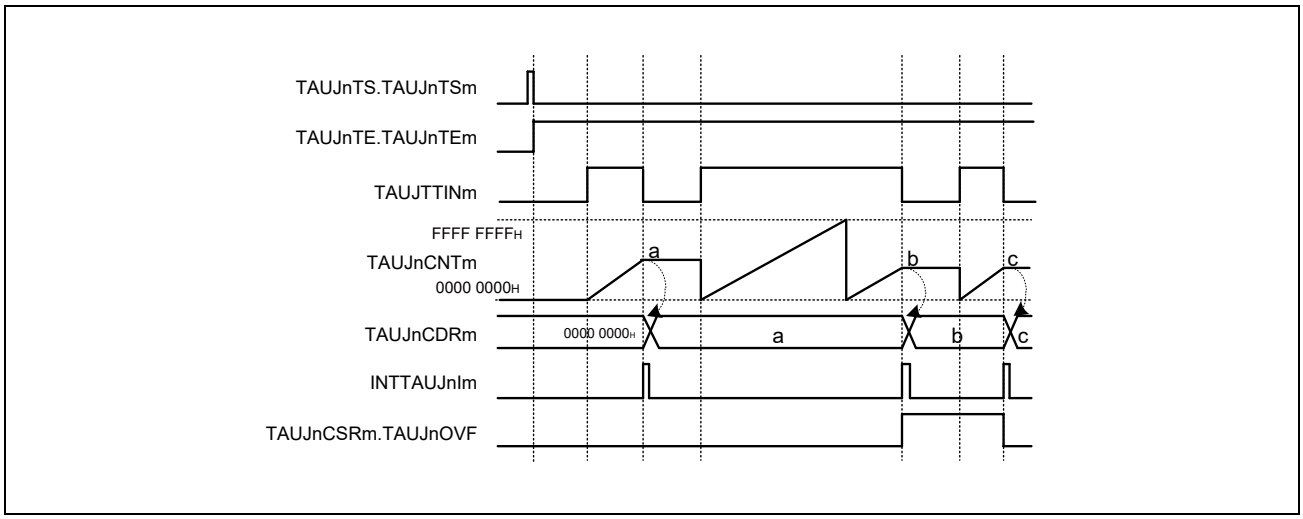


Figure 19.30 General Timing Diagram for TAUJTTINm Input Signal Width Measurement Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.49 Contents of TAUJnCMORm Register for TAUJTTINm Input Signal Width Measurement Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	0: Not used, so set to 0
10 to 8	TAUJnSTS[2:0]	010: Effective edge of the TAUJTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUJnCOS[1:0]	See **Table 19.48, Effects of an Overflow.**
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0110: Capture and one-count mode
0	TAUJnMD0	0: Disables the start trigger during operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.50 Contents of TAUJnCMURm Register for TAUJTTINm Input Signal Width Measurement Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	10: Rising and falling edge detection (low width measurement) 11: Rising and falling edge detection (high width measurement)

(c) Channel output mode

TAUJnTOE.TAUJnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Signal Width Measurement Function. Therefore, these registers must be set to 0.

Table 19.51 Simultaneous Rewrite Settings for TAUJTTINm Input Signal Width Measurement Function

Bit Name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm = 0), set these bits to 0

(5) Operating procedure for TAUJTTINm input signal width measurement function

Table 19.52 Operating Procedure for TAUJTTINm Input Signal Width Measurement Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 19.49, Contents of TAUJnCMORm Register for TAUJTTINm Input Signal Width Measurement Function** and **Table 19.50, Contents of TAUJnCMURm Register for TAUJTTINm Input Signal Width Measurement Function**. The TAUJnCDRm register functions as a capture register.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. • TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and TAUJnCNTm waits for detection of the TAUJTTINm start edge. When a TAUJTTINm start edge is detected, TAUJnCNTm starts to count up.
During operation	Detection of TAUJTTINm edges. The TAUJnCDRm, TAUJnCNTm, and TAUJnCSRm registers can be read at any time. The TAUJnCSCm.TAUJnCLOV bit can be set to 1.	TAUJnCNTm starts to count up from 0000 0000_H. When an effective TAUJTTINm edge is detected: • AUJnCNTm transfers (captures) its value to TAUJnCDRm, and retains its value • INTTAUJnIm is then generated. • Counting stops at the value that transferred to TAUJnCDRm + 1 and TAUJnCNTm waits for detection of the TAUJTTINm start edge. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TAUJnTTm to 1. • TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. • TAUJnCNTm stops and both it and TAUJnCSRm.TAUJnOVF retain their current values.

Restart

(6) Specific timing diagrams: overflow behavior

(a) TAUJnCMORm.TAUJnCOS[1:0] = 00_B

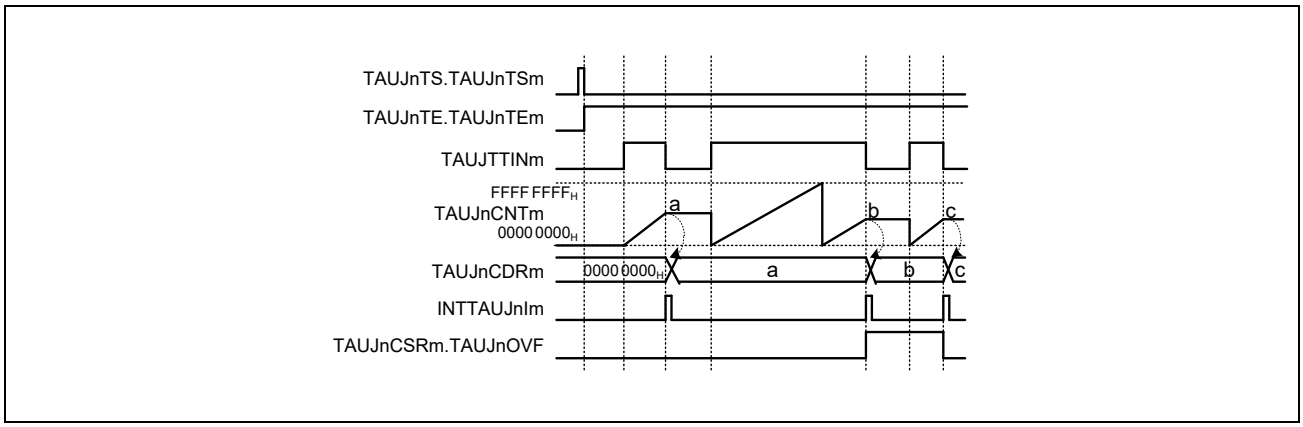


Figure 19.31 TAUJnCMORm.TAUJnCOS[1:0] = 00_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- When an overflow occurs, the value of TAUJnCDRm remains unchanged and TAUJnCSRm.TAUJnOVF remains = 0.
- Upon detection of the next effective TAUJTTINm input edge, the value of TAUJnCNTm is written to TAUJnCDRm and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next effective TAUJTTINm input edge with no overflow occurring, TAUJnCSRm.TAUJnOVF is cleared to 0.

(b) TAUJnCMORm.TAUJnCOS[1:0] = 01_B

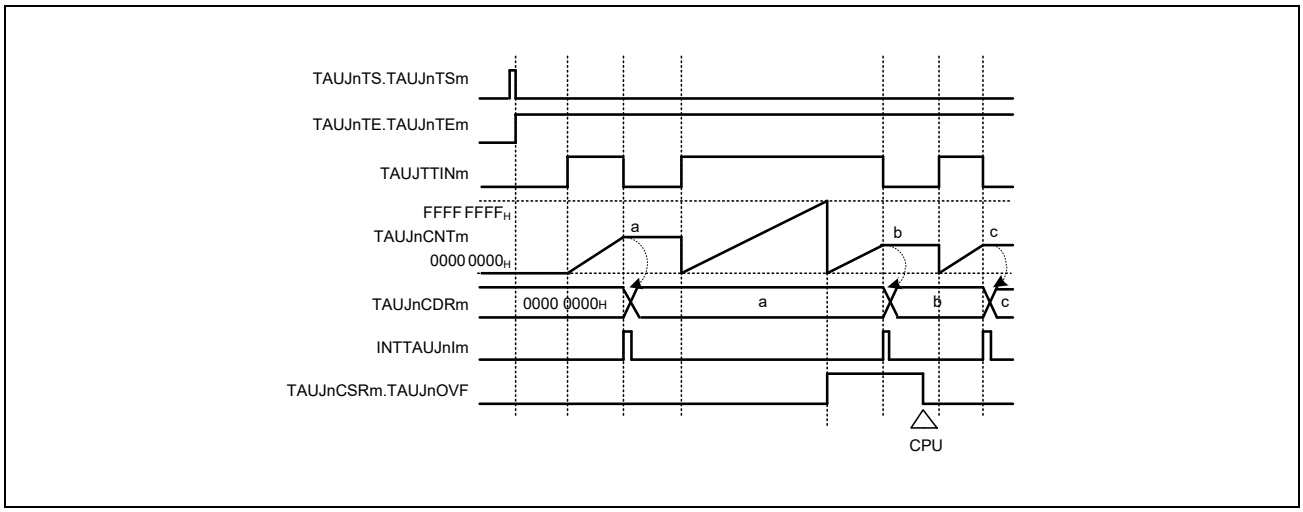


Figure 19.32 TAUJnCMORm.TAUJnCOS[1:0] = 01_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- When an overflow occurs, the value of TAUJnCDRm remains unchanged and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next effective TAUJTTINm input edge, the value of TAUJnCNTm is written to TAUJnCDRm.
- TAUJnCSRm.TAUJnOVF is only cleared by a CPU command (by setting TAUJnCSCm.TAUJnCLOV bit = 1).

(c) TAUJnCMORm.TAUJnCOS[1:0] = 10_B

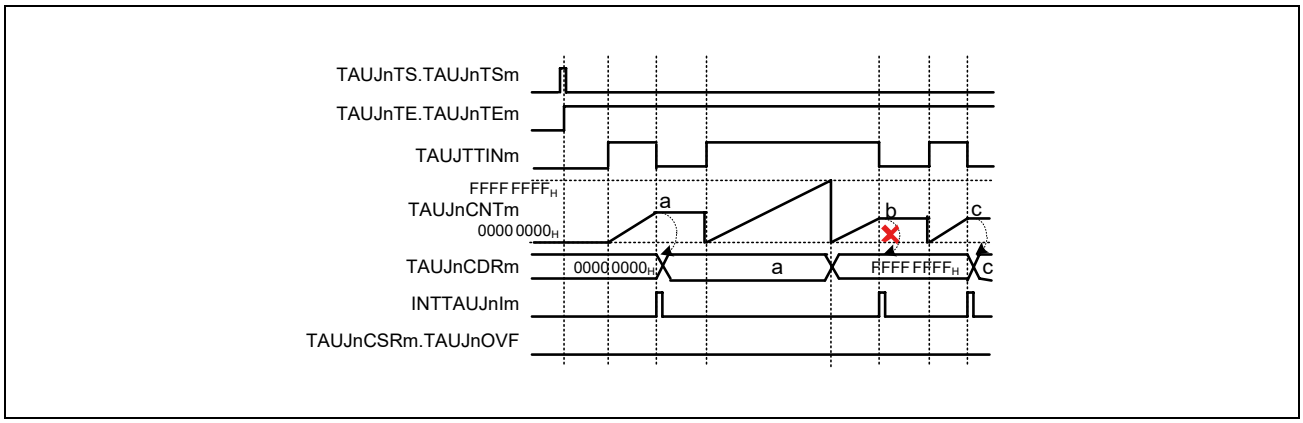


Figure 19.33 TAUJnCMORm.TAUJnCOS[1:0] = 10_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- When an overflow occurs, TAUJnCDRm is set to FFFF $FFFF_H$ and TAUJnCSRm.TAUJnOVF remains = 0.
- Upon detection of the next effective TAUJTTINm input edge, the counter of TAUJnCNTm is stopped, and TAUJnCDRm and TAUJnCSRm.TAUJnOVF remain unchanged.
- Thus, the next effective TAUJTTINm input edge after the overflow is ignored.

(d) TAUJnCMORm.TAUJnCOS[1:0] = 11_B

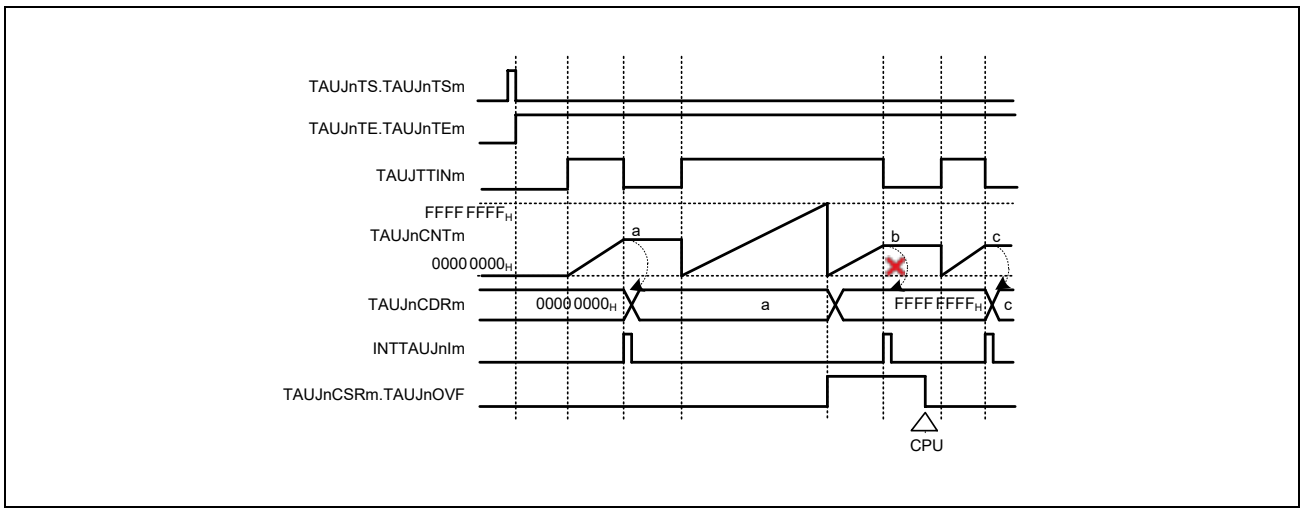


Figure 19.34 TAUJnCMORm.TAUJnCOS[1:0] = 11_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- When an overflow occurs, TAUJnCDRm is set to FFFF FFFF$_H$, and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next effective TAUJTTINm input edge, the counter of TAUJnCNTm is stopped, and TAUJnCDRm and TAUJnCSRm.TAUJnOVF remain unchanged.
- Thus, the next effective TAUJTTINm input edge after the overflow is ignored.
- TAUJnCSRm.TAUJnOVF is cleared by setting the TAUJnCSCm.TAUJnCLOV bit to 1.

19.4.9.5 TAUJTTINm Input Position Detection Function

(1) Overview

Summary

This function measures the interval of an input signal by capturing the counter value on an effective edge of the TAUJTTINm signal.

NOTE

The input TAUJTTINm is sampled at the frequency of the operation clock, specified by the TAUJnCMORm.TAUJnCKS[1:0] bits. As a result, the output cycle of TAUJTTOUTm has an error of ± 1 operation clock cycle.

Prerequisites

- The operating mode should be set to count capture mode (see **Table 19.53, Contents of TAUJnCMORm Register for TAUJTTINm Input Position Detection Function**).
- TAUJTTOUTm is not used for this function

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The counter starts to count from 0000 0000_H. When an effective TAUJTTINm input stop edge is detected, the current TAUJnCNTm value is written to TAUJnCDRm and an interrupt (INTTAUJnIm) is generated. The counter continues to count.

When the counter reaches FFFF $FFFF_H$, the counter restarts from 0000 0000_H.

Conditions

If the TAUJnCMORm.TAUJnMD0 bit is set to 0, the first interrupt after a start or restart is not generated. For details, see **Section 19.4.6, TAUJTTOUTm Output and INTTAUJnIm Generation when Counter Starts or Restarts**.

(2) Equations

Function duration at a TAUJTTINm input pulse = count clock cycle × (TAUJnCDRm capture value + 1)

(3) Block diagram and general timing diagram

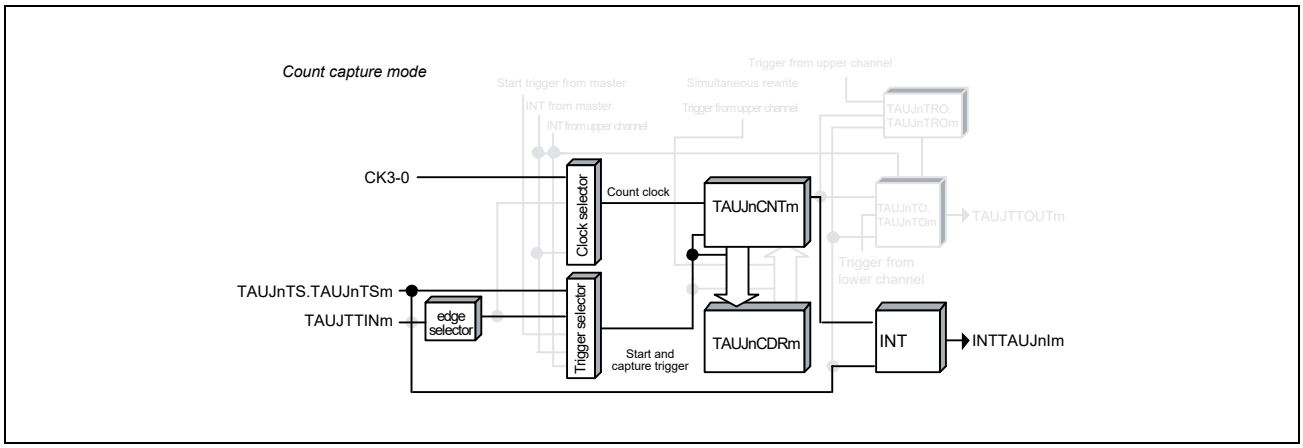


Figure 19.35 Block Diagram of TAUJTTINm Input Period Count Detection Function

The following settings apply to the general timing diagram:

- INTTAUJnIm not generated at operation start (TAUJnCMORm.TAUJnMD0 = 0)
- Falling edge detection (TAUJnCMURm.TAUJnTIS[1:0] = 00_B)

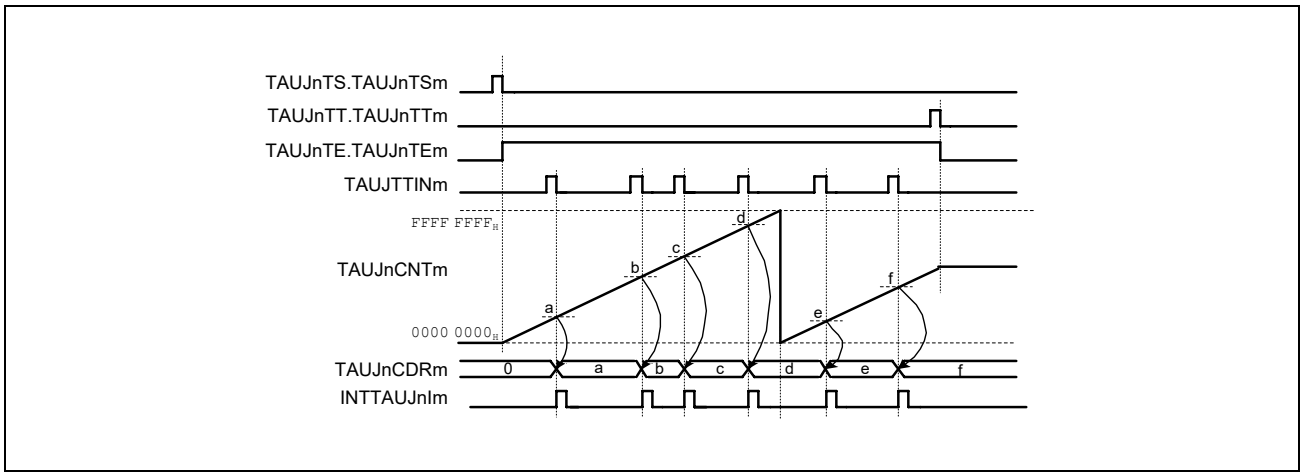


Figure 19.36 General Timing Diagram for TAUJTTINm Input Position Detection Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.53 Contents of TAUJnCMORm Register for TAUJTTINm Input Position Detection Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	0: Not used, so set to 0
10 to 8	TAUJnSTS[2:0]	001: Effective TAUJTTINm input edge signal is used as an external capture trigger.
7, 6	TAUJnCOS[1:0]	01: This value must be set.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	1011: Count capture mode
0	TAUJnMD0	0: INTTAUJnIm is not generated at the start of operation. 1: INTTAUJnIm is generated at the start of operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.54 Contents of TAUJnCMURm Register for TAUJTTINm Input Position Detection Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Falling edge detection 01: Rising edge detection 10: Rising and falling edge detection

(c) Channel output mode

The channel output mode is not used by this function.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Position Detection Function. Therefore, these registers must be set to 0.

Table 19.55 Simultaneous Rewrite Settings for TAUJTTINm Input Position Detection Function

Bit Name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm=0), set these bits to 0

(5) Operating procedure for TAUJTTINm input position detection function

Table 19.56 Operating Procedure for TAUJTTINm Input Position Detection Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 19.53, Contents of TAUJnCMORm Register for TAUJTTINm Input Position Detection Function** and **Table 19.54, Contents of TAUJnCMURm Register for TAUJTTINm Input Position Detection Function**. The TAUJnCDRm register functions as a capture register.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. • TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and the counter starts. • INTTAUJnIm is generated when TAUJnCMORm.TAUJnMD0 is set to 1.
During operation	The TAUJnCMURm.TAUJnTIS[1:0] bits can be changed at any time. The TAUJnCDRm and TAUJnCSRm registers can be read at any time.	TAUJnCNTm starts to count up from 0000 0000_H. When an effective TAUJTTINm edge is detected: • TAUJnCNTm transfers (captures) its value to TAUJnCDRm • INTTAUJnIm is output. • The counter value is not cleared to 0000 0000_H and TAUJnCNTm continues count operation. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TTm to 1. • TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. • TAUJnCNTm stops. TAUJnCNTm retains its current value.

Restart

(6) Specific timing diagrams

(a) Operation stop and restart

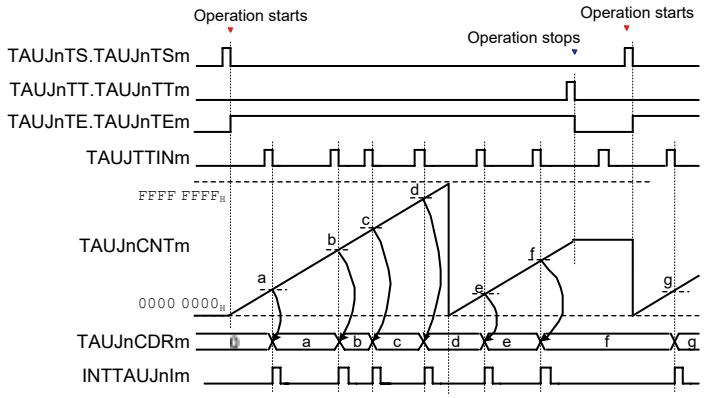


Figure 19.37 Operation Stop and Restart, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1, which in turn sets TAUJnTE.TAUJnTEm to 0.
- TAUJnCNTm stops and the current value is retained.
- If the counter is stopped, effective TAUJTTINm input edges are ignored.
- The counter can be restarted by setting TAUJnTS.TAUJnTSm to 1. TAUJnCNTm restarts to count from $0000\ 0000_H$.

19.4.9.6 TAUJTTINm Input Period Count Detection Function

(1) Overview

Summary

This function measures the cumulative width of a TAUJTTINm input signal.

Prerequisites

- The operating mode should be set to capture and gate count mode (see **Table 19.57, Contents of TAUJnCMORm Register for TAUJTTINm Input Period Count Detection Function**).
- TAUJTTOUTm is not used for this function.

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The counter awaits an effective TAUJTTINm input edge.

When an effective TAUJTTINm input start edge is detected, the counter starts to count from 0000 0000_H.

When an effective TAUJTTINm input stop edge is detected, the current TAUJnCNTm value is written to TAUJnCDRm and an interrupt (INTTAUJnIm) is generated. The counter stops and retains its value until the next effective TAUJTTINm input start edge is detected.

When the next effective TAUJTTINm input start edge is detected, the counter restarts counting from the stop value.

If the counter reaches FFFF $FFFF_H$, the counter restarts from 0000 0000_H.

NOTE

The input TAUJTTINm signal is sampled at the frequency of the operation clock, specified by the TAUJnCMORm.TAUJnCKS[1:0] bits.

Conditions

The effective start and stop edges are specified by the TAUJnCMURm.TAUJnTIS[1:0] bits:

- If TAUJnCMURm.TAUJnTIS[1:0] = 10_B, the TAUJTTINm input low period is counted. The start trigger is a falling edge and the stop trigger is a rising edge.
- If TAUJnCMURm.TAUJnTIS[1:0] = 11_B, the TAUJTTINm input high period is counted. The start trigger is a rising edge and the stop trigger is a falling edge.

(2) Equations

Cumulative TAUJTTINm input width = count clock cycle × (TAUJnCDRm capture value + 1)

(3) Block diagram and general timing diagram

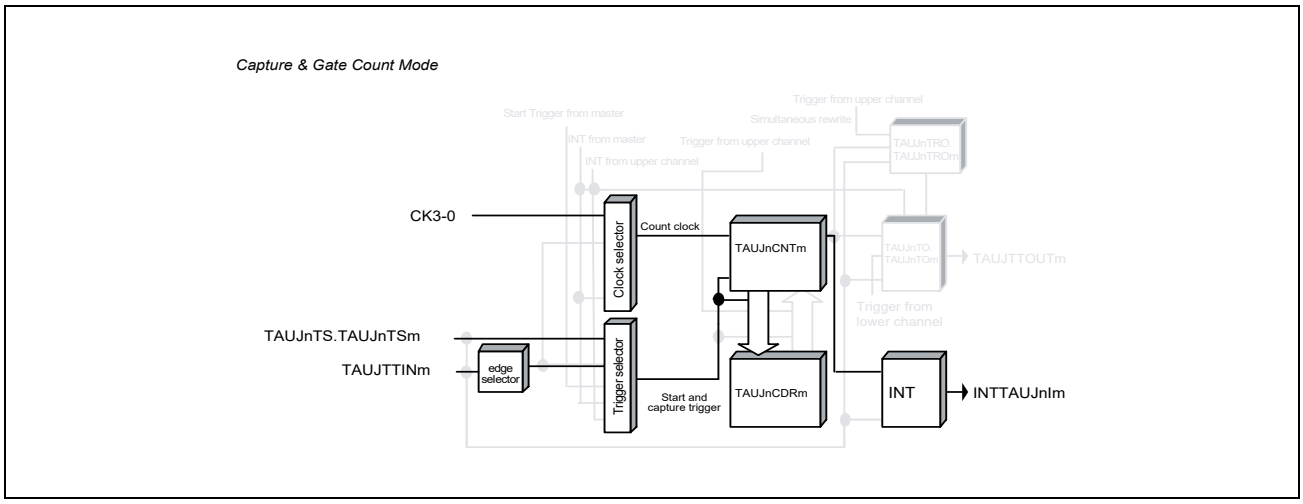


Figure 19.38 Block Diagram for TAUJTTINm Input Period Count Detection Function

The following settings apply to the general timing diagram:

- Rising and falling edge detection = high width measurement (TAUJnCMURm.TAUJnTIS[1:0] = 11_B)

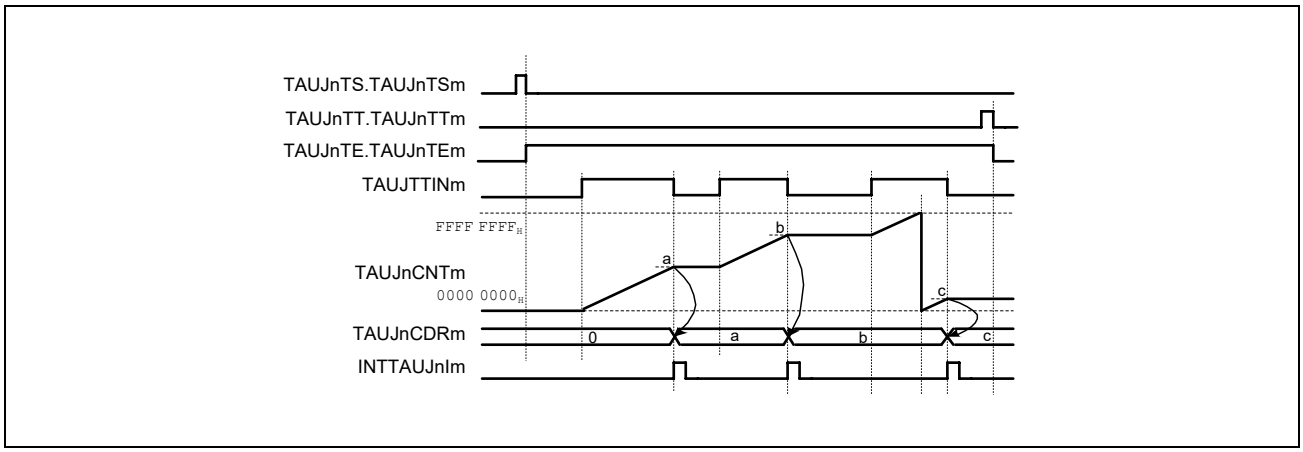


Figure 19.39 General Timing Diagram for TAUJTTINm Input Period Count Detection Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.57 Contents of TAUJnCMORm Register for TAUJTTINm Input Period Count Detection Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	0: Not used, so set to 0
10 to 8	TAUJnSTS[2:0]	010: Effective edge of the TAUJTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUJnCOS[1:0]	01: This value must be set.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	1101: Capture and gate count mode
0	TAUJnMD0	0: Disables the start trigger during operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.58 Contents of TAUJnCMURm Register for TAUJTTINm Input Period Count Detection Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	10: Rising and falling edge detection (Low width measurement) 11: Rising and falling edge detection (High width measurement)

(c) Channel output mode

TAUJnTOE.TAUJnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Period Count Detection Function. Therefore, these registers must be set to 0.

Table 19.59 Simultaneous Rewrite Settings for TAUJTTINm Input Period Count Detection Function

Bit Name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm=0), set these bits to 0

(5) Operating procedure for TAUJTTINm input period count detection function

Table 19.60 Operating Procedure for TAUJTTINm Input Period Count Detection Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 19.57, Contents of TAUJnCMORm Register for TAUJTTINm Input Period Count Detection Function** and **Table 19.58, Contents of TAUJnCMURm Register for TAUJTTINm Input Period Count Detection Function**. The TAUJnCDRm register functions as a capture register.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. • TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0. Detection of TAUJTTINm start edge.	TAUJnTE.TAUJnTEm is set to 1 and TAUJnCNTm waits for detection of the TAUJTTINm start edge. When a start edge is detected, TAUJnCNTm is cleared to 0000 0000_H and TAUJnCNTm starts to count up.
During operation	Detection of TAUJTTINm edges. The TAUJnCDRm, TAUJnCNTm, and TAUJnCSRm registers can be read at any time.	When a TAUJTTINm start edge (rising edge for high level width measurement, falling edge for low level width measurement) is detected, TAUJnCNTm starts to count up from the stop value. • When TAUJnCNTm detects a start edge (falling edge for high level width measurement, rising edge for low level width measurement), it transfers the value to TAUJnCDRm and INTTAUJnIm is generated. • Counting stops at the “value transferred to TAUJnCDRm + 1” and TAUJnCNTm waits for detection of the TAUJTTINm start edge. • If the TAUJnCNTm reaches FFFF $FFFF_H$, the counter restarts count operation from 0000 0000_H. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TTm to 1. • TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. • TAUJnCNTm stops. TAUJnCNTm retains its current value.

Restart

(6) Specific timing diagrams

(a) Operation stop and restart

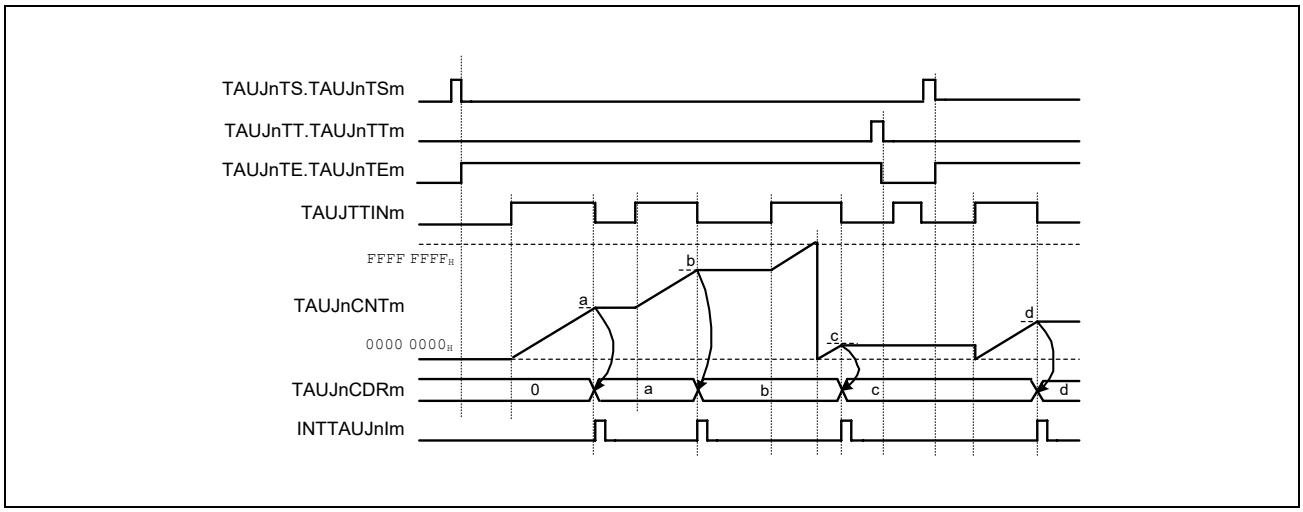


Figure 19.40 Operation Stop and Restart, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1, which in turn sets TAUJnTE.TAUJnTEm to 0.
- TAUJnCNTm stops and the current value is retained.
- If the counter is stopped, effective TAUJTTINm input edges are ignored.
- The counter can be restarted by setting TAUJnTS. TAUJnTSm to 1. TAUJnCNTm restarts to count from $0000\ 0000_H$.

19.4.10 Synchronous Channel Operation Functions

- **Section 19.4.10.1, PWM Output Function**

19.4.10.1 PWM Output Function

(1) Overview

Summary

This function generates multiple PWM outputs by using a master and multiple slave channels. It enables the pulse cycle (frequency) and the duty of the TAUJTTOUTm to be set. The pulse cycle is set in the master channel. The duty is set in the slave channel.

Prerequisites

- Two channels
- The operating mode for the master channel should be set to interval timer mode (see **Table 19.61, Contents of TAUJnCMORm Register of Master Channel for PWM Output Function**).
- The operating mode for the slave channels should be set to one-count mode (see **Table 19.64, Contents of TAUJnCMORm Register of Slave Channel for PWM Output Function**).
- TAUJTTOUTm is not used with the master channel of this function.
- The channel output mode for the slave channels should be set to synchronous channel output mode 1 (see **Section 19.4.4, Channel Output Modes**).

Description

The counters are enabled by setting the channel trigger bits (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The current value of TAUJnCDRm is written to TAUJnCNTm and the counters start to count down from these values. INTTAUJnIm is generated on the master channel and TAUJTTOUTm (slave) is set or reset to realize the PWM output.

- Master channel:
  When the counter of the master channel reaches 0000 0000_H, pulse cycle time has elapsed and INTTAUJnIm is generated. The TAUJnCDRm value is realized to TAUJnCNTm, and the counter counts down.
- Slave channel(s):
  The INTTAUJnIm of the master channel triggers the counter of the slave channel(s). The current value of TAUJnCDRm (slave) is written to TAUJnCNTm (slave) and the counter starts to count down from this value. The TAUJTTOUTm signal is set, to the active level.
  When the counter reaches 0000 0000_H, i.e. duty time has elapsed, INTTAUJnIm is generated and the TAUJTTOUTm signal is set to the inactive level. The counter returns to FFFF $FFFF_H$ and awaits the next INTTAUJnIm of the master channel, and thus the start of the next pulse cycle.

The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1 for the master and slave channel(s), which in turn sets TAUJnTE.TAUJnTEm to 0. TAUJnCNTm and TAUJTTOUTm of master and slave channel(s) stop but retain their values. The counters can be restarted by setting TAUJnTS.TAUJnTSm to 1.

Conditions

Simultaneous rewrite can be used with this function. Please refer to **Section 19.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUJnCDRm (master) + 1) × count clock cycle

Duty cycle [%] = (TAUJnCDRm (slave) / (TAUJnCDRm (master) +1)) × 100

- Duty cycle = 0%
  TAUJnCDRm (slave) = $0000\ 0000_H$
- Duty cycle = 100%
  TAUJnCDRm (slave) ≥ TAUJnCDRm (master) + 1

(3) Block diagram and general timing diagram

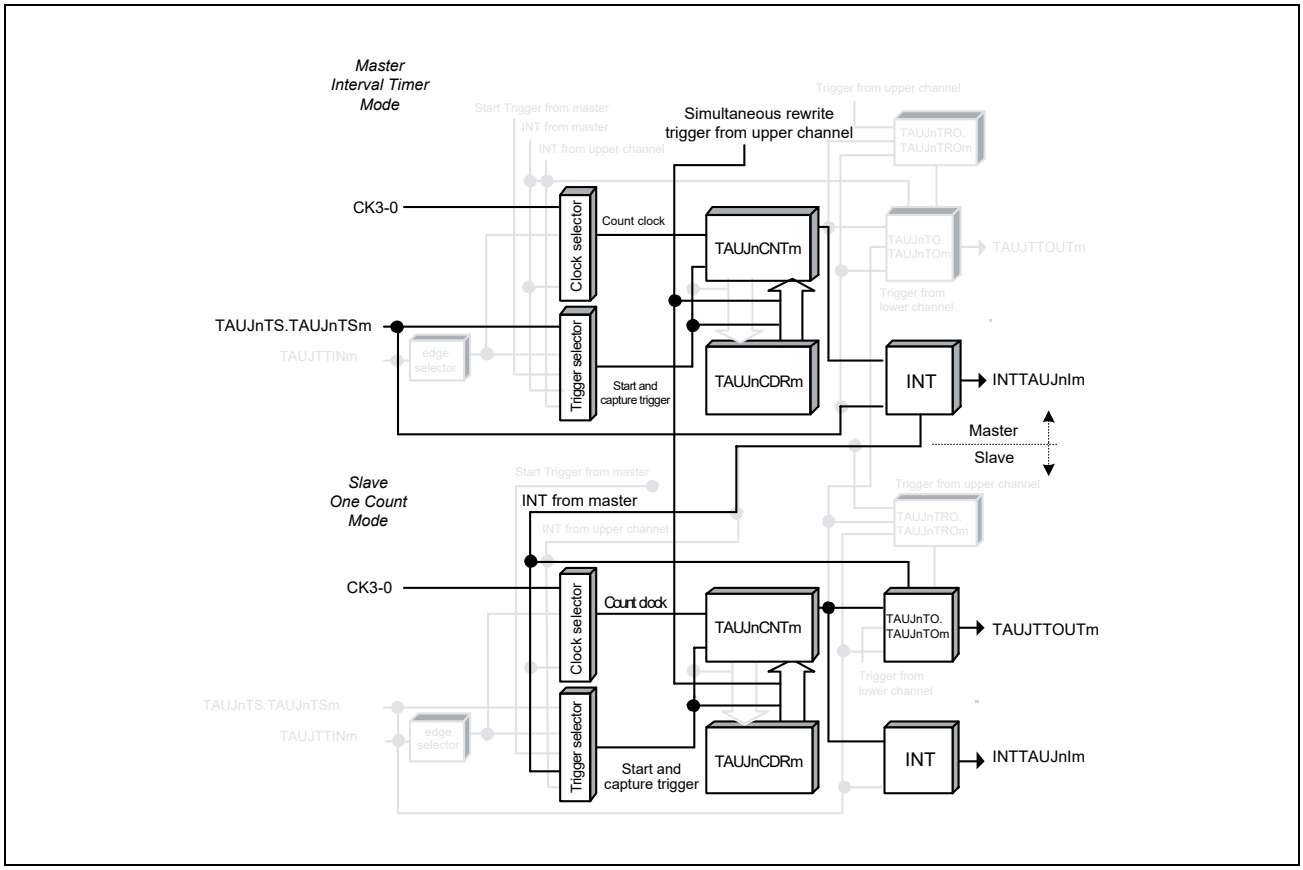


Figure 19.41 Block Diagram for PWM Output Function

The following settings apply to the general timing diagram:

- Slave channel: Positive logic (TAUJnTOL.TAUJnTOLm = 0)

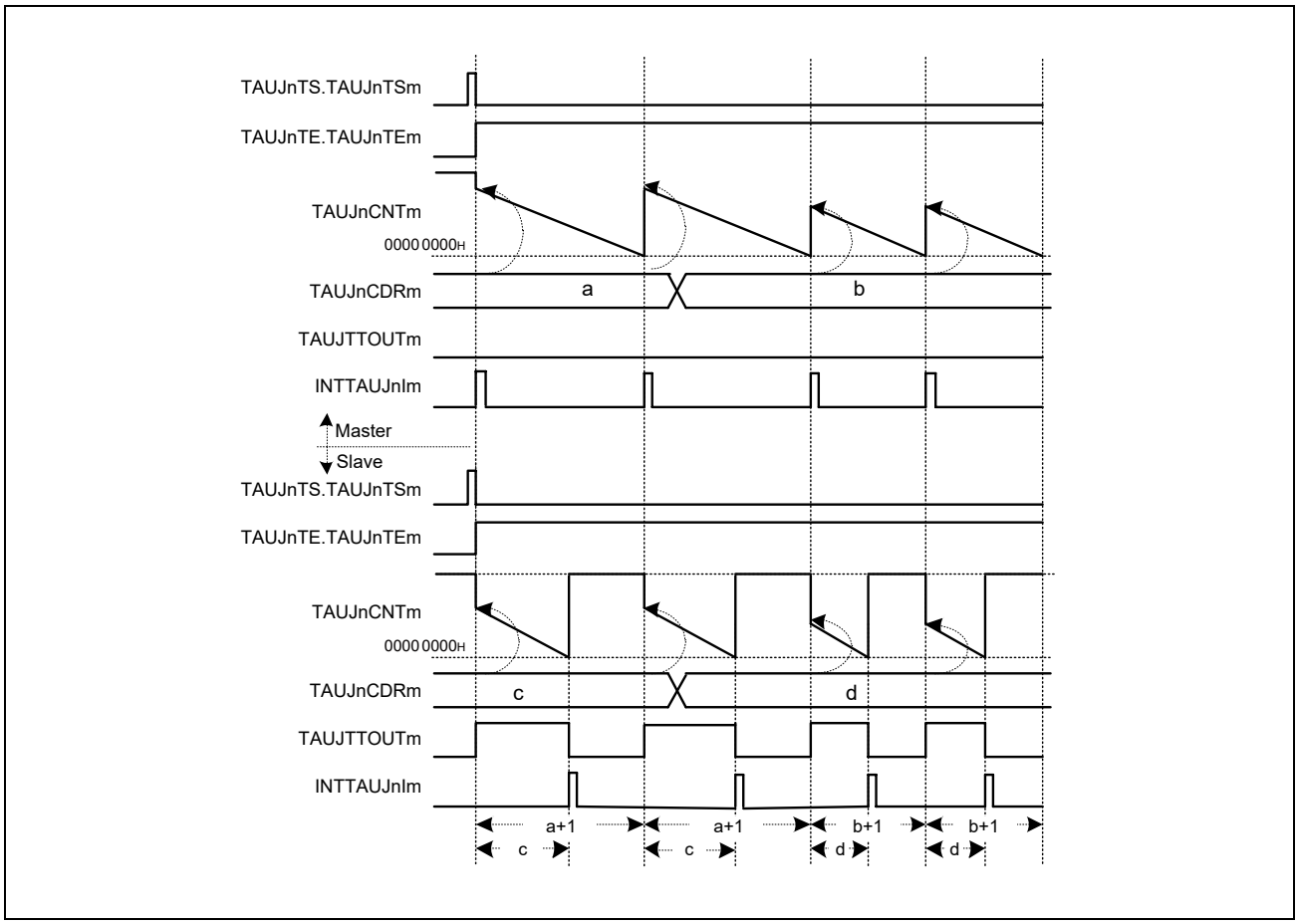


Figure 19.42 General Timing Diagram for PWM Output Function

NOTES

- The interval between the slave channel starting to count and an interrupt being generated is the value of corresponding TAUJnCDRm, whereas for the master channel the interval is the corresponding TAUJnCDRm + 1.
- TAUJTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUJnIm of the master channel.

(4) Register settings for the master channel

(a) TAUJnCMORm for the master channel

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.61 Contents of TAUJnCMORm Register of Master Channel for PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUJnCKS[1:0] bit of the master and slave channel(s) must be identical.
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	1: Channel is master channel
10 to 8	TAUJnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUJnCOS[1:0]	00: Not used, so set to 00
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0000: Interval timer mode
0	TAUJnMD0	1: INTTAUJnIm is generated at the start of operation.

(b) TAUJnCMURm for the master channel

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.62 Contents of TAUJnCMURm Register of Master Channel for PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS [1:0]	00: Not used so set to 00

(c) Channel output mode for the master channel

The channel output mode is not used by this function.

(d) Simultaneous rewrite for the master channel

The simultaneous rewrite settings of the master and slave channel must be identical.

Table 19.63 Simultaneous Rewrite Settings for the Master Channel of the PWM Output Function

Bit Name	Setting
TAUJnRDE.TAUJnRDEm	1: Enables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: The simultaneous rewrite trigger signal is generated when the master channel starts counting

(5) Register settings for the slave channel(s)

(a) TAUJnCMORm for the slave channel(s)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS [1:0]		TAUJnCCS [1:0]		TAUJn MAS	TAUJnSTS [2:0]			TAUJnCOS [1:0]		—	TAUJnMD [4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 19.64 Contents of TAUJnCMORm Register of Slave Channel for PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUJnCKS[1:0] bit of the master and slave channel(s) must be identical.
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	0: Channel is slave channel
10 to 8	TAUJnSTS[2:0]	100: INTTAUJnIm of master channel is a start trigger.
7, 6	TAUJnCOS[1:0]	00: Not used, so set to 00
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0100: One-count mode
0	TAUJnMD0	1: INTTAUJnIm is generated at the start of operation.

(b) TAUJnCMURm for the slave channel(s)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.65 Contents of TAUJnCMURm Register of Slave Channel for PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Not used so set to 00

(c) Channel output mode for the slave channel(s)

Table 19.66 Control Bit Settings for Synchronous Channel Output Mode 1

Bit Name	Setting
TAUJnTOE.TAUJnTOEm	1: Enables independent channel output mode
TAUJnTO.TAUJnTOm	0: Low level 1: High level
TAUJnTOM.TAUJnTOMm	1: Synchronous channel operation
TAUJnTOC.TAUJnTOCm	0: Operating mode 1
TAUJnTOL.TAUJnTOLm	0: Positive logic 1: Negative logic

(d) Simultaneous rewrite for the slave channel(s)

The simultaneous rewrite settings of the master and slave channel must be identical.

Table 19.67 Simultaneous Rewrite Settings for the Slave Channel of the PWM Output Function

Bit Name	Setting
TAUJnRDE.TAUJnRDEm	1: Enables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: The simultaneous rewrite trigger signal is generated when the master channel starts counting

(6) Operating procedure for PWM output function

Table 19.68 Operating Procedure for PWM Output Function

	Operation	Status of TAUJn
Initial channel setting	Master channel: set the TAUJnCMORm and TAUJnCMURm registers and the channel output mode as described in **Section (4) Register settings for the master channel**. Slave channel: set the TAUJnCMORm and TAUJnCMURm registers and the channel output mode as described in **Section (5) TAUJnCMURm for the slave channel(s)**. Set the values of the TAUJnCDRm registers of all channels	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm of the master and slave channels to 1 simultaneously. • TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm (master and slave channels) is set to 1 and the counters of the master and slave channels start. INTTAUJnIm is generated on the master channel and TAUJTTOUTm (slave) is set.
During operation	TAUJnCDRm can be changed at any time. TAUJnCNTm and TAUJnRSF.TAUJnRSFm can be read at any time. TAUJnRDT.TAUJnRDTm can be changed during operation.	TAUJnCNTm of the master channel loads TAUJnCDRm and counts down. When the counter reaches 0000 0000_H: • INTTAUJnIm (master) is generated • TAUJnCNTm (master) loads the TAUJnCDRm value and continues count operation • TAUJnCNTm (slave) loads the TAUJnCDRm value and counts down • TAUJTTOUTm (slave) is set to the active level. When TAUJnCNTm (slave) reaches 0000 0000_H: • INTTAUJnIm (slave) is generated • TAUJTTOUTm (slave) is set to the inactive level.
Stop operation	Set TAUJnTT.TAUJnTTm of the master and slave channels to 1 simultaneously. • TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. • TAUJnCNTm and TAUJTTOUTm stop and retain their current values.

Restart

(7) Specific timing diagrams

(a) Duty cycle = 0%

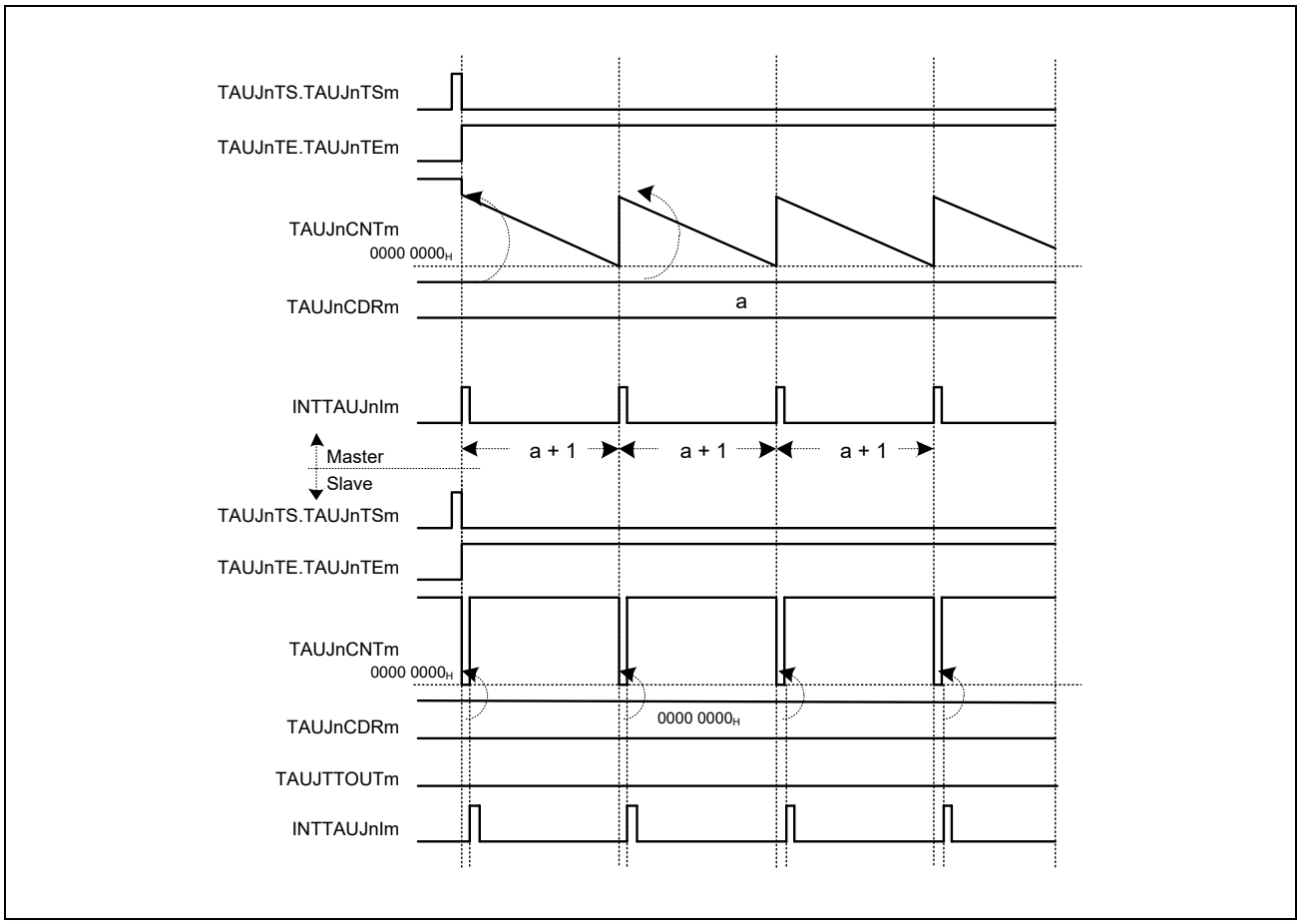


Figure 19.43 TAUJnCDRm (slave) = 0000 0000$_H$, Positive Logic (TAUJnTOL.TAUJnTOLm (slave) = 0)

- Every time the master channel generates an interrupt (INTTAUJnIm), 0000 0000$_H$ is written to TAUJnCNTm (slave). Therefore, TAUJnCNTm (slave) cannot start to count and TAUJTTOUTm remains at not active state.
- The value of TAUJnCDRm is loaded into TAUJnCNTm (slave) to generate an interrupt.

(b) Duty cycle = 100%

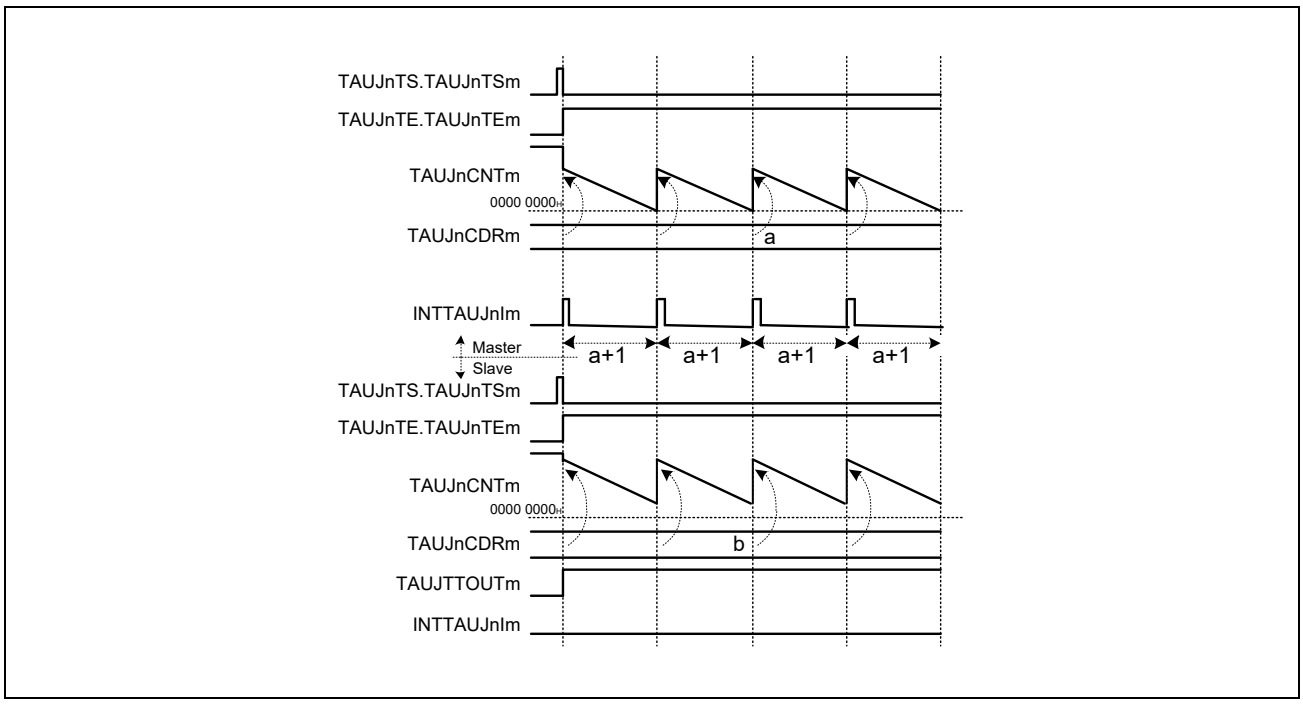


Figure 19.44 TAUJnCDRm (Slave) ≥ TAUJnCDRm (Master) + 1, Positive Logic (TAUJnTOL.TAUJnTOLm (slave) = 0)

If the TAUJnCDRm (slave) value is greater than the TAUJnCDRm (master) value, no interrupt occurs because the counter of the slave channel does not reach 0000 0000_H. TAUJTTOUTm remains active.

(c) Operation stop and restart

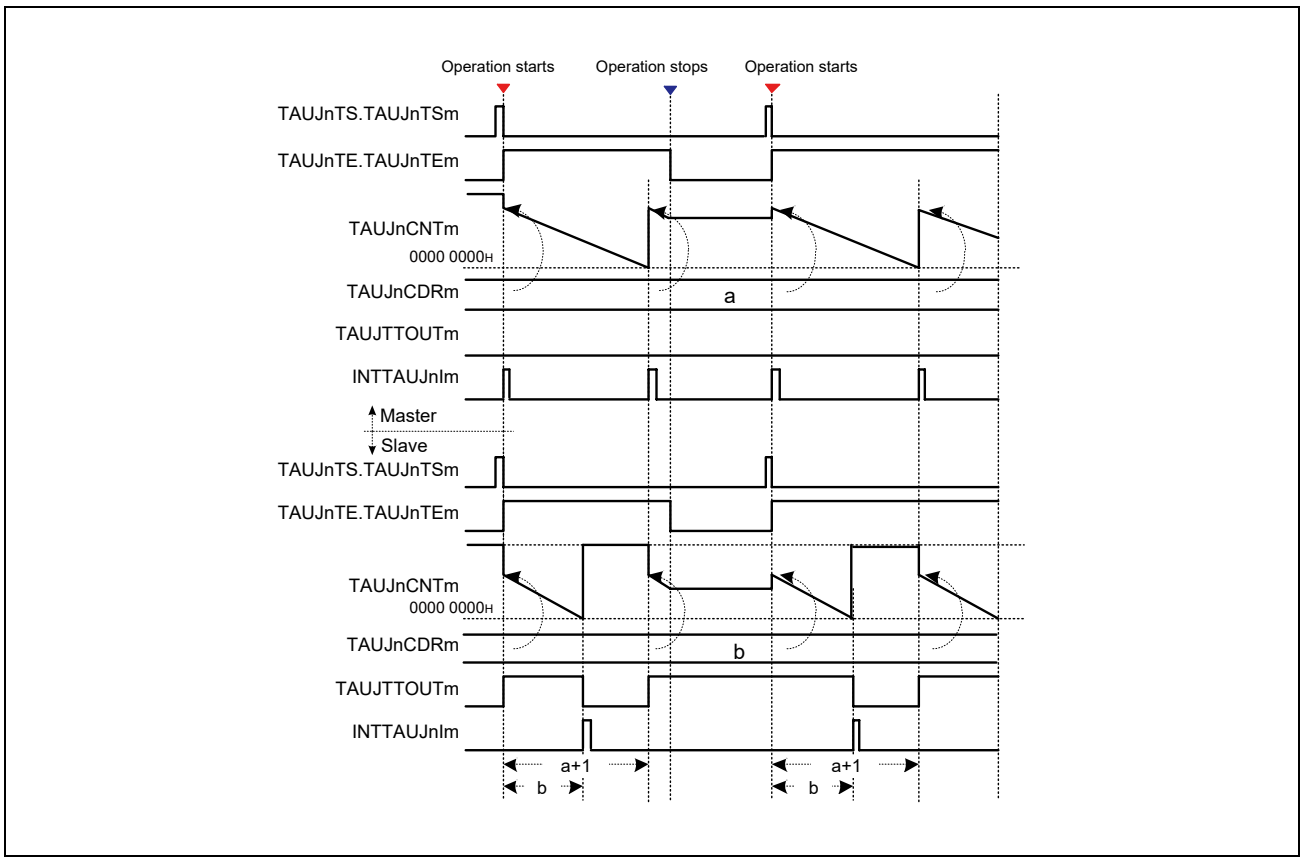


Figure 19.45 Stop and Restart Operation, Positive Logic (TAUJnTOL.TAUJnTOLm (slave) = 0)

- The counter can be stopped by setting TAUJnTT.TAUJnTTm of master and slave channels to 1. This sets TAUJnTE.TAUJnTEm to 0.
- TAUJnCNTm and TAUJTTOUTm of every channel stop and retain their current values. No interrupt occurs.
- The counter can be restarted by setting TAUJnTS.TAUJnTSm of master and slave channels to 1. TAUJnCDRm value of master and slave channels is loaded into TAUJnCNTm. The counter starts to count down from this value.

Section 20 Motor Control Timer (TSG3)

This section contains a generic description of the motor control timer (TSG3).

The first part of the section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of the TSG3.

20.1 Features of RH850/C1M-A TSG3

20.1.1 Number of Units

This LSI has the following number of units of TSG3.

Table 20.1 Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of Units	3	2
Name	TSG3n (n = 0 to 2)	TSG3n (n = 0, 1)

Note: TSG32 does not support the functions of synchronous operation with EMU3, linkage with ENCA, and external pattern input.

Table 20.2 Index

Index	Meaning
n	Throughout this section, the individual TSG3 units are identified by the index “n” (n = 0 to 2), for example, TSG3nCTL0 for the TSG3n control register 0.
m, k	Throughout this section, the variables used for description are indicated by the letter “m” or “k”, for example, TSG3nCMPmE is a non-specified compare register.

20.1.2 Register Base Address

TSG3 base addresses are listed in the following table.

TSG3 register addresses are given as offsets from the base addresses in general.

Table 20.3 Register Base Address

Base Address Name	Base Address
<TSG30_base>	FFE7 0000_H
<TSG31_base>	FF87 1000_H
<TSG32_base>	FFE7 2000_H*1

Note 1. Not supported for RH850/C1M-A1.

20.1.3 Clock Supply

Clock supply by and to TSG3 is listed in the following table.

Table 20.4 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
TSG3n	PCLK	CLKC_HSB (Non-modulated high-speed peripheral clock)

20.1.4 Interrupt Requests

TSG3 interrupt requests are listed in the following table.

Table 20.5 Interrupt Requests (1/2)

Unit Interrupt Signal	Outline	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
TSG30						
INTTSG30I0	TSG30 compare match interrupt 0	232	—	—	—	—
INTTSG30I1	TSG30 compare match interrupt 1	190	—	—	—	—
INTTSG30I2	TSG30 compare match interrupt 2	191	—	—	—	—
INTTSG30I3	TSG30 compare match interrupt 3	192	—	—	—	—
INTTSG30I4	TSG30 compare match interrupt 4	193	—	—	—	—
INTTSG30I5	TSG30 compare match interrupt 5	194	—	—	—	—
INTTSG30I6	TSG30 compare match interrupt 6	195	—	—	—	—
INTTSG30I7	TSG30 compare match interrupt 7	196	—	—	—	—
INTTSG30I8	TSG30 compare match interrupt 8	197	—	—	—	—
INTTSG30I9	TSG30 compare match interrupt 9	198	—	—	—	—
INTTSG30I10	TSG30 compare match interrupt 10	199	—	—	—	—
INTTSG30I11	TSG30 compare match interrupt 11	200	87	—	87	—
INTTSG30I12	TSG30 compare match interrupt 12	201	88	—	88	—
INTTSG30IPEK	TSG30 peak interrupt	233	89	—	89	—
INTTSG30IVLY	TSG30 trough interrupt	234	90	—	90	—
INTTSG30IER	TSG30 error interrupt	202	—	—	—	—
INTTSG30IWN	TSG30 warning interrupt	203	—	—	—	—

Table 20.5 Interrupt Requests (2/2)

Unit Interrupt Signal	Outline	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
TSG31						
INTTSG31I0	TSG31 compare match interrupt 0	235	—	—	—	—
INTTSG31I1	TSG31 compare match interrupt 1	204	—	—	—	—
INTTSG31I2	TSG31 compare match interrupt 2	205	—	—	—	—
INTTSG31I3	TSG31 compare match interrupt 3	206	—	—	—	—
INTTSG31I4	TSG31 compare match interrupt 4	207	—	—	—	—
INTTSG31I5	TSG31 compare match interrupt 5	208	—	—	—	—
INTTSG31I6	TSG31 compare match interrupt 6	209	—	—	—	—
INTTSG31I7	TSG31 compare match interrupt 7	210	—	—	—	—
INTTSG31I8	TSG31 compare match interrupt 8	211	—	—	—	—
INTTSG31I9	TSG31 compare match interrupt 9	212	—	—	—	—
INTTSG31I10	TSG31 compare match interrupt 10	213	—	—	—	—
INTTSG31I11	TSG31 compare match interrupt 11	214	91	—	91	—
INTTSG31I12	TSG31 compare match interrupt 12	215	92	—	92	—
INTTSG31IPEK	TSG31 peak interrupt	236	93	—	93	—
INTTSG31IVLY	TSG31 trough interrupt	237	94	—	94	—
INTTSG31IER	TSG31 error interrupt	216	—	—	—	—
INTTSG31IWN	TSG31 warning interrupt	217	—	—	—	—
TSG32*2						
INTTSG32I0	TSG32 compare match interrupt 0	238	—	—	—	—
INTTSG32I1	TSG32 compare match interrupt 1	218	—	—	—	—
INTTSG32I2	TSG32 compare match interrupt 2	219	—	—	—	—
INTTSG32I3	TSG32 compare match interrupt 3	220	—	—	—	—
INTTSG32I4	TSG32 compare match interrupt 4	221	—	—	—	—
INTTSG32I5	TSG32 compare match interrupt 5	222	—	—	—	—
INTTSG32I6	TSG32 compare match interrupt 6	223	—	—	—	—
INTTSG32I7	TSG32 compare match interrupt 7	224	—	—	—	—
INTTSG32I8	TSG32 compare match interrupt 8	225	—	—	—	—
INTTSG32I9	TSG32 compare match interrupt 9	226	—	—	—	—
INTTSG32I10	TSG32 compare match interrupt 10	227	—	—	—	—
INTTSG32I11	TSG32 compare match interrupt 11	228	95	—	95	—
INTTSG32I12	TSG32 compare match interrupt 12	229	96	—	96	—
INTTSG32IPEK	TSG32 peak interrupt	239	97	—	97	—
INTTSG32IVLY	TSG32 trough interrupt	240	98	—	98	—
INTTSG32IER	TSG32 error interrupt	230	—	—	—	—
INTTSG32IWN	TSG32 warning interrupt	231	—	—	—	—

Note 1. 1st: Primary channel, 2nd: Secondary channel

Note 2. Not supported for RH850/C1M-A1.

"—": No number is assigned

20.1.5 Reset Sources

TSG3 reset sources are listed in the following table.

TSG3 is initialized by this reset source.

Table 20.6 Reset Source

Unit Name	Reset Source
TSG3n	All reset sources

20.1.6 External Input/Output Signals

External input/output signals of TSG3 are listed in the following table.

Table 20.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal Name
TSG30		
TSG30PTSI0-TSG30PTSI2	External pattern input	ENCA0E0, ENCA0E1, ENCA0EC*1
TSG30O0-TSG30O7	Timer output	TSG30O0-TSG30O7
TSG31		
TSG31PTSI0-TSG31PTSI2	External pattern input	ENCA1E0, ENCA1E1, ENCA1EC*1
TSG31O0-TSG31O7	Timer output	TSG31O0-TSG31O7
TSG32*2		
TSG32O0-TSG32O7	Timer output	TSG32O0-TSG32O7

Note 1. For the RH850/C1M-A products, external pattern input is shared with ENCAn input. For the detailed specification, see **Section 24.2.2.21, PIC1BTSGHALLSEL – Hall Sensor Input Select Register**.

Note 2. Not supported for RH850/C1M-A1.

20.2 Overview

20.2.1 Functional Overview

The TSG3n is an 18-bit timer counter with various motor control functions.

- Count clock resolution: Minimum 12.5 ns (count clock = 80 MHz)
- Operating mode corresponding to various motor control methods
- Compare registers with reload buffer
- 10-bit dead time counter
  - Dead time counter with reload buffer.
  - Independent dead time can be set for positive to inverse phase change and inverse to positive phase change.
- A/D conversion trigger signal generation
  - Three A/D conversion trigger signals can be generated by the compare registers TSG3nDCMP0E, TSG3nDCMP1E, and TSG3nDCMP2E.
  - Skipping function of A/D conversion trigger signals TSG3nADTRG0 and TSG3nADTRG1 can be set independently. The skipping ratio can be selected among 1/1, 1/2, 1/4, and 1/8.
  - The dedicated pin (TSG3nO7) can be used to output the toggle or diagnostic signal set by the TSG3nADTRG0 signal and reset by the TSG3nADTRG1 signal.
- Interrupt skipping
  - Skipping rate: 1/1 to 1/32
- Forced output stop function
  - Using the timer option (TAPA) function allows the high impedance control of the TSG3nO1 to TSG3nO6 pin outputs.
- Compare value setting
  - Reload (simultaneous rewrite) or anytime rewrite can be selected.
- Reload mode
  - Writing to TSG3nCMP1E enables reload (the reload request flag (TSG3nRSF) is set), and allows simultaneous transfer of the values of multiple registers.
  - Data can be transferred at peak/trough/peak or trough reload timing
  - Reload request flag (TSG3nRSF)
  - Register address assignment allowing DMA transfer

Reload skipping

- HT-PWM mode
  - 0 to 100% PWM duty cycle output is possible (with possible dead time reduction).
  - The LSB in the compare register can be used to append an additional pulse to the PWM output during count up, thus improving the output resolution without extra load on software.
- 120-DC control*[1]
  - Semi-automatic cruise function (trigger signal can be generated by an offset in conjunction with two-phase encoder, three-phase encoder, or ENCA).
- Three-phase encoder function (hall sensor signals can be input).*[1]
- Active level of the output pins TSG3nO1 to TSG3nO6 can be set individually.
- Fail-safe function (warning interrupt or error interrupt can be generated)
  - Simultaneous active output detect function for positive and inverse phases.
  - Abnormal input detection function of the three-phase encoder
- Direct transfer of carrier-cycle setting and PWM duty setting from EMU3.*[1]
- Output selectable between rectangular waveform output from EMU3 and PWM generate by TSG3.*[1]

Note 1. TSG32 does not support the functions of synchronous operation with EMU3, linkage with ENCA, and external pattern input.

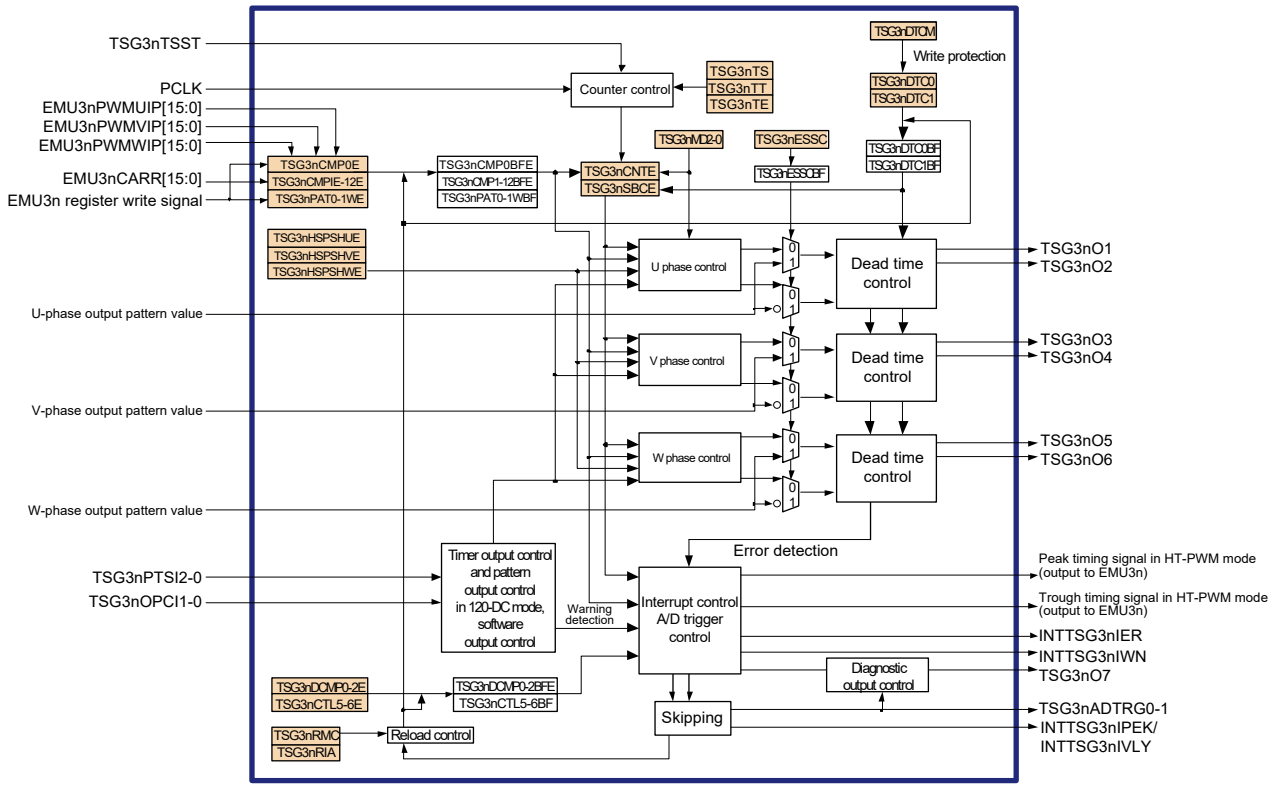
NOTE

In this section, the active level is assumed to be high.

20.2.2 Block Diagram

The block diagram below shows the major components of TSG3 (EMU30, EMU31*1).

Note 1. RH850/C1M-A1 does not have EMU31.



Note: EMU3nPWMUIP[15:0]: EMU3n U-phase compare value register (input from EMU3n)
EMU3nPWMVIP[15:0]: EMU3n V-phase compare value register (input from EMU3n)
EMU3nPWMWIP[15:0]: EMU3n W-phase compare value register (input from EMU3n)
EMU3nCARR[15:0]: EMU3n carrier cycle register (input from EMU3n)
EMU3n register write signal: Write timing signal of the above register value (input from EMU3n)
U-phase output pattern value: Input from EMU3n rectangular waveform generator
V-phase output pattern value: Input from EMU3n rectangular waveform generator
W-phase output pattern value: Input from EMU3n rectangular waveform generator

Figure 20.1 TSG3n Block Diagram (n = 0, 1)

- TSG3nTSST: Simultaneous start trigger (input from PIC1B)

The block diagram below shows the major components of TSG3.

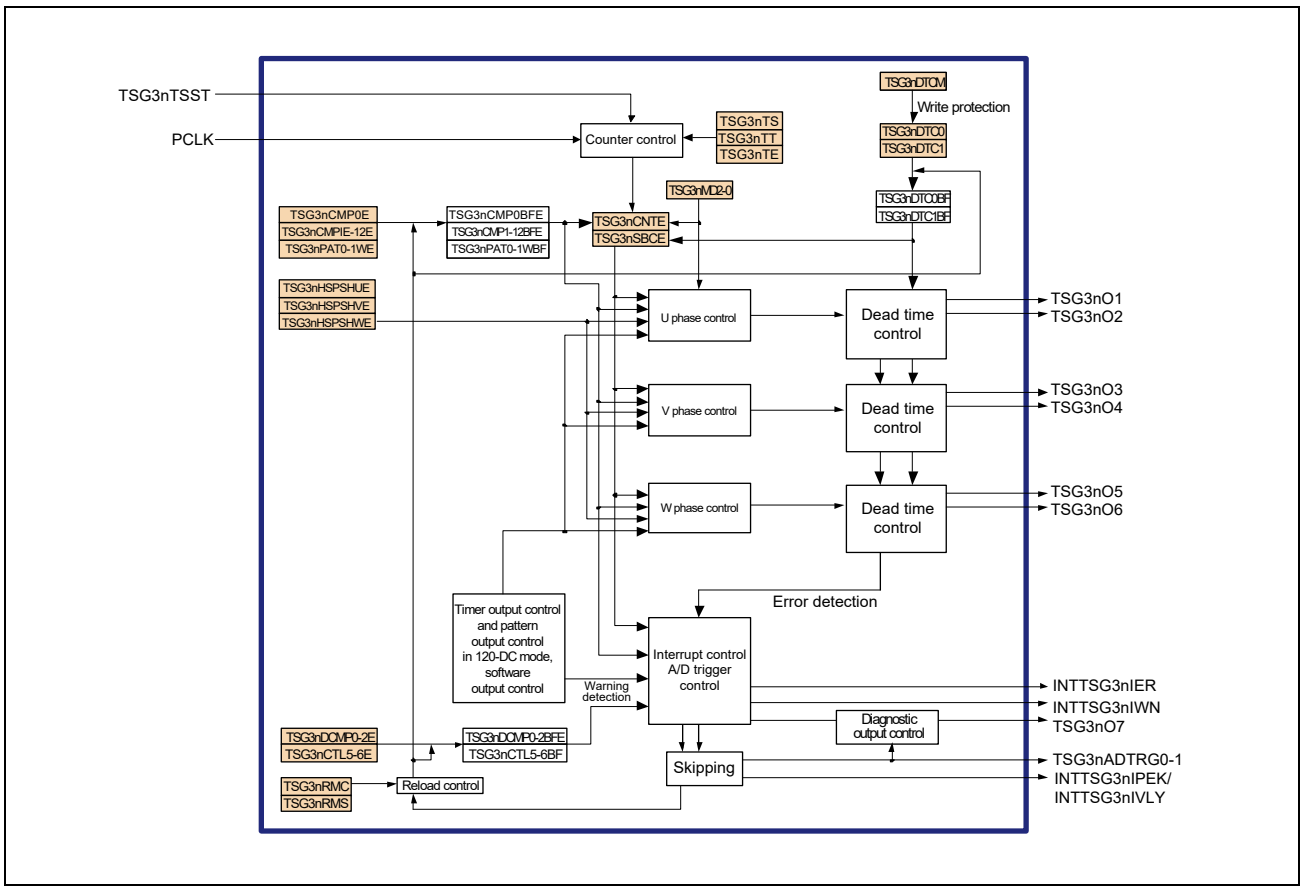


Figure 20.2 TSG3n Block Diagram (n = 2)

- TSG3nTSST: Simultaneous start trigger (input from PIC1B)

20.3 Registers

20.3.1 List of Registers

TSG3n registers are listed in the following table.

For <TSG3n_base>, see **Section 20.1.2, Register Base Address.**

Table 20.8 List of Registers (1/3)

Module Name	Register Name	Symbol	Address	Reload
TSG3n	TSG3n control register 0	TSG3nCTL0	<TSG3n_base> + 208_H	Disabled
TSG3n	TSG3n control register 1	TSG3nCTL1	<TSG3n_base> + $20C_H$	Disabled
TSG3n	TSG3n control register 3	TSG3nCTL3	<TSG3n_base> + 004_H	Disabled
TSG3n	TSG3n control register 4	TSG3nCTL4	<TSG3n_base> + $07C_H$	Enabled
TSG3n	TSG3n control register 5	TSG3nCTL5	<TSG3n_base> + 008_H	Disabled
TSG3n	TSG3n control register 6	TSG3nCTL6	<TSG3n_base> + $00C_H$	Disabled
TSG3n	TSG3n control register 7	TSG3nCTL7	<TSG3n_base> + 218_H	Disabled
TSG3n	TSG3n control register 8	TSG3nCTL8	<TSG3n_base> + $21C_H$	Disabled
TSG3n	TSG3n I/O control register 0	TSG3nIOC0	<TSG3n_base> + 200_H	Disabled
TSG3n	TSG3n I/O control register 1	TSG3nIOC1	<TSG3n_base> + 204_H	Disabled
TSG3n	TSG3n I/O control register 2	TSG3nIOC2	<TSG3n_base> + 000_H	Disabled
TSG3n	TSG3n I/O control register 3	TSG3nIOC3	<TSG3n_base> + 074_H	Enabled
TSG3n	TSG3n status register 0	TSG3nSTR0	<TSG3n_base> + 010_H	Disabled
TSG3n	TSG3n status register 1	TSG3nSTR1	<TSG3n_base> + 014_H	Disabled
TSG3n	TSG3n status register 2	TSG3nSTR2	<TSG3n_base> + 018_H	Disabled
TSG3n	TSG3n status clear trigger register	TSG3nSTC	<TSG3n_base> + $01C_H$	Disabled
TSG3n	TSG3n option register 0	TSG3nOPT0	<TSG3n_base> + 020_H	Disabled
TSG3n	TSG3n option register 1	TSG3nOPT1	<TSG3n_base> + 024_H	Disabled
TSG3n	TSG3n option register 2	TSG3nOPT2	<TSG3n_base> + $03C_H$	Disabled
TSG3n	TSG3n option 2 buffer register	TSG3nOPT2BF	<TSG3n_base> + $0CC_H$	Disabled
TSG3n	TSG3n trigger register 0	TSG3nTRG0	<TSG3n_base> + 030_H	Disabled
TSG3n	TSG3n trigger register 1	TSG3nTRG1	<TSG3n_base> + 034_H	Disabled
TSG3n	TSG3n trigger register 2	TSG3nTRG2	<TSG3n_base> + 038_H	Disabled
TSG3n	TSG3n counter read buffer register	TSG3nCNT	<TSG3n_base> + 028_H	Disabled
TSG3n	TSG3n bit extended counter read buffer register	TSG3nCNTE	<TSG3n_base> + $1A0_H$	Disabled
TSG3n	TSG3n sub-counter read buffer register	TSG3nSBC	<TSG3n_base> + $02C_H$	Disabled
TSG3n	TSG3n bit extended sub-counter read buffer register	TSG3nSBCE	<TSG3n_base> + $1A4_H$	Disabled
TSG3n	TSG3n compare register 0	TSG3nCMP0	<TSG3n_base> + 058_H	Enabled
TSG3n	TSG3n bit extended compare register 0	TSG3nCMP0E	<TSG3n_base> + $14C_H$	Enabled
TSG3n	TSG3n compare register 1, 2	TSG3nCMP1W	<TSG3n_base> + 040_H	Enabled
TSG3n	TSG3n compare register 5, 6	TSG3nCMP5W	<TSG3n_base> + 044_H	Enabled
TSG3n	TSG3n compare register 9, 10	TSG3nCMP9W	<TSG3n_base> + 048_H	Enabled
TSG3n	TSG3n compare register 3, 4	TSG3nCMP3W	<TSG3n_base> + $04C_H$	Enabled
TSG3n	TSG3n compare register 7, 8	TSG3nCMP7W	<TSG3n_base> + 050_H	Enabled
TSG3n	TSG3n compare register 11, 12	TSG3nCMP11W	<TSG3n_base> + 054_H	Enabled

Table 20.8 List of Registers (2/3)

Module Name	Register Name	Symbol	Address	Reload
TSG3n	TSG3n compare register 1	TSG3nCMP1	<TSG3n_base> + 080_H	Enabled
TSG3n	TSG3n compare register 2	TSG3nCMP2	<TSG3n_base> + 084_H	Enabled
TSG3n	TSG3n compare register 3	TSG3nCMP3	<TSG3n_base> + 098_H	Enabled
TSG3n	TSG3n compare register 4	TSG3nCMP4	<TSG3n_base> + $09C_H$	Enabled
TSG3n	TSG3n compare register 5	TSG3nCMP5	<TSG3n_base> + 088_H	Enabled
TSG3n	TSG3n compare register 6	TSG3nCMP6	<TSG3n_base> + $08C_H$	Enabled
TSG3n	TSG3n compare register 7	TSG3nCMP7	<TSG3n_base> + $0A0_H$	Enabled
TSG3n	TSG3n compare register 8	TSG3nCMP8	<TSG3n_base> + $0A4_H$	Enabled
TSG3n	TSG3n compare register 9	TSG3nCMP9	<TSG3n_base> + 090_H	Enabled
TSG3n	TSG3n compare register 10	TSG3nCMP10	<TSG3n_base> + 094_H	Enabled
TSG3n	TSG3n compare register 11	TSG3nCMP11	<TSG3n_base> + $0A8_H$	Enabled
TSG3n	TSG3n compare register 12	TSG3nCMP12	<TSG3n_base> + $0AC_H$	Enabled
TSG3n	TSG3n bit extended compare register 1	TSG3nCMP1E	<TSG3n_base> + $17C_H$	Enabled
TSG3n	TSG3n bit extended compare register 2	TSG3nCMP2E	<TSG3n_base> + 178_H	Enabled
TSG3n	TSG3n bit extended compare register 3	TSG3nCMP3E	<TSG3n_base> + 164_H	Enabled
TSG3n	TSG3n bit extended compare register 4	TSG3nCMP4E	<TSG3n_base> + 160_H	Enabled
TSG3n	TSG3n bit extended compare register 5	TSG3nCMP5E	<TSG3n_base> + 174_H	Enabled
TSG3n	TSG3n bit extended compare register 6	TSG3nCMP6E	<TSG3n_base> + 170_H	Enabled
TSG3n	TSG3n bit extended compare register 7	TSG3nCMP7E	<TSG3n_base> + $15C_H$	Enabled
TSG3n	TSG3n bit extended compare register 8	TSG3nCMP8E	<TSG3n_base> + 158_H	Enabled
TSG3n	TSG3n bit extended compare register 9	TSG3nCMP9E	<TSG3n_base> + $16C_H$	Enabled
TSG3n	TSG3n bit extended compare register 10	TSG3nCMP10E	<TSG3n_base> + 168_H	Enabled
TSG3n	TSG3n bit extended compare register 11	TSG3nCMP11E	<TSG3n_base> + 154_H	Enabled
TSG3n	TSG3n bit extended compare register 12	TSG3nCMP12E	<TSG3n_base> + 150_H	Enabled
TSG3n	TSG3n diagnostic output compare register 0, 1	TSG3nDCMP0W	<TSG3n_base> + $05C_H$	Enabled
TSG3n	TSG3n diagnostic output compare register 2	TSG3nDCMP2	<TSG3n_base> + 060_H	Enabled
TSG3n	TSG3n bit extended diagnostic output compare register 0	TSG3nDCMP0E	<TSG3n_base> + 148_H	Enabled
TSG3n	TSG3n bit extended diagnostic output compare register 1	TSG3nDCMP1E	<TSG3n_base> + 144_H	Enabled
TSG3n	TSG3n bit extended diagnostic output compare register 2	TSG3nDCMP2E	<TSG3n_base> + 140_H	Enabled
TSG3n	TSG3n pattern register 0	TSG3nPAT0W	<TSG3n_base> + 064_H	Enabled
TSG3n	TSG3n pattern register 1	TSG3nPAT1W	<TSG3n_base> + 068_H	Enabled
TSG3n	TSG3n dead time control register 0	TSG3nDTC0W	<TSG3n_base> + $06C_H$	Enabled
TSG3n	TSG3n dead time control register 1	TSG3nDTC1W	<TSG3n_base> + 070_H	Enabled
TSG3n	TSG3n HT-PWM U phase compare register	TSG3nCMPU	<TSG3n_base> + $0B0_H$	Enabled
TSG3n	TSG3n HT-PWM V phase compare register	TSG3nCMPV	<TSG3n_base> + $0B4_H$	Enabled
TSG3n	TSG3n HT-PWM W phase compare register	TSG3nCMPW	<TSG3n_base> + $0B8_H$	Enabled

Table 20.8 List of Registers (3/3)

Module Name	Register Name	Symbol	Address	Reload
TSG3n	TSG3n bit extended HT-PWM U phase compare register	TSG3nCMPUE	<TSG3n_base> + 188_H	Enabled
TSG3n	TSG3n bit extended HT-PWM V phase compare register	TSG3nCMPVE	<TSG3n_base> + 184_H	Enabled
TSG3n	TSG3n bit extended HT-PWM W phase compare register	TSG3nCMPWE	<TSG3n_base> + 180_H	Enabled
TSG3n	TSG3n SP-PWM U phase active width register	TSG3nUPW	<TSG3n_base> + $0BC_H$	Enabled
TSG3n	TSG3n SP-PWM V phase active width register	TSG3nVPW	<TSG3n_base> + $0C0_H$	Enabled
TSG3n	TSG3n SP-PWM W phase active width register	TSG3nWPW	<TSG3n_base> + $0C4_H$	Enabled
TSG3n	TSG3n bit extended SP-PWM U phase active width register	TSG3nUPWE	<TSG3n_base> + 198_H	Enabled
TSG3n	TSG3n bit extended SP-PWM V phase active width register	TSG3nVPWE	<TSG3n_base> + 194_H	Enabled
TSG3n	TSG3n bit extended SP-PWM W phase active width register	TSG3nWPWE	<TSG3n_base> + 190_H	Enabled
TSG3n	TSG3n HSP-PWM W phase shift register	TSG3nHSPSHWE	<TSG3n_base> + 120_H	Enabled
TSG3n	TSG3n HSP-PWM V phase shift register	TSG3nHSPSHVE	<TSG3n_base> + 124_H	Enabled
TSG3n	TSG3n HSP-PWM U phase shift register	TSG3nHSPSHUE	<TSG3n_base> + 128_H	Enabled
TSG3n	TSG3n HSP-PWM W phase compare register	TSG3nHSPCMWE	<TSG3n_base> + $12C_H$	Enabled
TSG3n	TSG3n HSP-PWM V phase compare register	TSG3nHSPCMVE	<TSG3n_base> + 130_H	Enabled
TSG3n	TSG3n HSP-PWM U phase compare register	TSG3nHSPCMUE	<TSG3n_base> + 134_H	Enabled
TSG3n	TSG3n dead timer protection register	TSG3nDTPR	<TSG3n_base> + 210_H	Disabled

20.3.2 TSG3nCTL0 — TSG3n Control Register 0

This register specifies the pulse width for the diagnostic output and operating mode of the TSG3n.

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 208_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	TSG3nDWD	—	TSG3nMD[2:0]		
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R/W	R/W	R/W

Table 20.9 TSG3nCTL0 Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	TSG3nDWD	Selects the pulse width for the diagnostic output. 0: The output pulse width is set to 8 clocks. 1: The output pulse width is set to 16 clocks. The setting of this bit is valid when diagnostic output is enabled (TSG3nIOC1.TSG3nTGS = 1).
3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2 to 0	TSG3nMD[2:0]	Selects timer mode

TSG3n MD2	TSG3n MD1	TSG3n MD0	Timer Mode
0	0	0	PWM mode
0	0	1	HT-PWM mode(HT-PWM)
0	1	0	Shift pulse PWM mode(SP-PWM)
0	1	1	120-DC mode
1	0	0	High-accuracy shift pulse PWM mode (HSP-PWM)
Other than above			Setting prohibited

CAUTION

This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer

20.3.3 TSG3nCTL1 — TSG3n Control Register 1

This register controls the flags of TSG3n.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $20C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nTBA2	TSG3nTBA1	TSG3nTBA0	TSG3nPPC	TSG3nPEC	TSG3nTDC	TSG3nNDC	TSG3nPRC	TSG3nPTC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 20.10 TSG3nCTL1 Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after reset is read. When writing, write the value after reset.
9	TSG3nTBA2	Enables or disables detection of the simultaneous active states of the TSG3nO5 and TSG3nO6 pins. 0: Disables detection of simultaneous active states of the TSG3nO5 and TSG3nO6 pins. 1: Enables detection of simultaneous active states of the TSG3nO5 and TSG3nO6 pins. If the simultaneous active state is detected when the TSG3nIOC1.TSG3nEOC and TSG3nTBA2 bits are 1, the positive phase and inverse phase simultaneous active state detection flag 2 (TSG3nTBF2) is set to 1, and an error interrupt (INTTSG3nIER) is generated.
8	TSG3nTBA1	Enables or disables detection of the simultaneous active states of the TSG3nO3 and TSG3nO4 pins. 0: Disables detection of simultaneous active states of the TSG3nO3 and TSG3nO4 pins. 1: Enables detection of simultaneous active states of the TSG3nO3 and TSG3nO4 pins. If the simultaneous active state is detected when the TSG3nIOC1.TSG3nEOC and TSG3nTBA1bits are 1, the positive phase and inverse phase simultaneous active state detection flag 1 (TSG3nTBF1) is set to 1, and an error interrupt (INTTSG3nIER) is generated.
7	TSG3nTBA0	Enables or disables detection of the simultaneous active states of the TSG3nO1 and TSG3nO2 pins. 0: Disables detection of simultaneous active states of the TSG3nO1 and TSG3nO2 pins. 1: Enables detection of simultaneous active states of the TSG3nO1 and TSG3nO2 pins. If the simultaneous active state is detected when the TSG3nIOC1.TSG3nEOC and TSG3nTBA0 bits are 1, the positive phase and inverse phase simultaneous active state detection flag 0 (TSG3nTBF0) is set to 1, and an error interrupt (INTTSG3nIER) is generated.
6	TSG3nPPC	Enables or disables detection of the pattern phase difference (TSG3nSTR2.TSG3nPPF) between the TSG3nPTSI2-0 and TSG3nOPF2-0. 0: Disables detection of I/O pattern difference. 1: Enables detection of I/O pattern difference
5	TSG3nPEC	Enables or disables detection of the pattern error (TSG3nSTR2.TSG3nPEF) of the TSG3nPTSI2-0 pins. 0: Disables detection of the pattern error of the TSG3nPTSI2-0 pins. 1: Enables detection of the pattern error of the TSG3nPTSI2-0 pins.

Table 20.10 TSG3nCTL1 Register Contents (2/2)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>4</td><td>TSG3nTDC</td><td>Enables or disables detection of the simultaneous trigger
(TSG3nSTR2.TSG3nTDF) of the TSG3nOPCI0 and TSG3nOPCI1.
0: Disables detection of the simultaneous trigger of the TSG3nOPCI0 and TSG3nOPCI1.
1: Enables detection of the simultaneous trigger of the TSG3nOPCI0 and TSG3nOPCI1.</td></tr>
<tr><td>3</td><td>TSG3nNDC</td><td>Enables or disables detection of the noise generation (two or more pins change simultaneously) (TSG3nSTR2.TSG3nNDF) on the TSG3nPTSI2-0 pins.
0: Disables detection of the noise generation on the TSG3nPTSI2-0 pins.
1: Enables detection of the noise generation on the TSG3nPTSI2-0 pins.</td></tr>
<tr><td>2</td><td>TSG3nPRC</td><td>Enables or disables detection of the reversal of the pattern (TSG3nSTR2.TSG3nPRF) of the TSG3nPTSI2-0 pins.
0: Disables detection of the reversal of the pattern of the TSG3nPTSI2-0 pins.
1: Enables detection of the reversal of the pattern of the TSG3nPTSI2-0 pins.</td></tr>
<tr><td>1, 0</td><td>TSG3nPTC[1:0]</td><td>Enables or disables detection of an abnormal toggle (TSG3nSTR2.TSG3nPTF) of the TSG3nPTSI2-0 pins between TSG3nOPCI1 and TSG3nOPCI10 triggers
<table>
<tr><th>TSG3n PTC1</th><th>TSG3n PTC0</th><th>Function</th></tr>
<tr><td>0</td><td>0</td><td rowspan="2">Disables detection of an abnormal toggle of the TSG3nPTSI2-0 pins.</td></tr>
<tr><td>0</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>Enables detection of an abnormal toggle of the TSG3nPTSI2-0 pins.</td></tr>
<tr><td>1</td><td>1</td><td>Enables detection of an abnormal toggle of the TSG3nPTSI2-0 pins. When an abnormal toggle is detected, the pattern output switch trigger is automatically switched from trigger switch to pattern switch (TSG3nPOT is switched from 1 to 0.)</td></tr>
</table></td></tr>
</table>

CAUTIONS

1. If TSG3nDTC0 or TSG3nDTC1 is set to 0000_H (without dead time), the TSG3nTBA2-0 bits should be set to 0.
2. This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer

20.3.4 TSG3nCTL3 — TSG3n Control Register 3

This register selects the rewrite method of the compare registers.

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 004_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nRIA	TSG3nRMC
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 20.11 TSG3nCTL3 Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1	TSG3nRIA	Selects the reload timing of the compare register values. 0: The reload timing is set to peak reload timing (set by TSG3nCTL4.TSG3nPRE) and trough reload timing (set by TSG3nCTL4.TSG3nVRE). 1: The reload timing is set to peak interrupt timing and trough interrupt timing. The setting of this bit is valid in reload mode (TSG3nRMC = 0).
0	TSG3nRMC	Selects the transfer timing of the compare register values. 0: Reload mode (simultaneous rewrite) Writing to registers to be reloaded enables reloading and the register values are rewritten simultaneously at the next reload timing. Writing to any register other than registers to be reloaded does not enable reloading. For the register to be reloaded, see **Section 20.3.1, List of Registers.** 1: Anytime rewrite mode The compare registers are rewritten independently. Whenever a value is written to the compare register, the written value is reflected immediately. TSG3nRSF is cleared. Do not set TSG3nRMC to 1 when operated in 120-DC mode or in HSP-PWM mode.

20.3.5 TSG3nCTL4 — TSG3n Control Register 4

This register enables or disables generation of a peak interrupt and a trough interrupt, and the reload timing.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $07C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3n PRE	TSG3n VRE	TSG3n PIE	TSG3n VIE	TSG3nRCC[04:00]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 20.12 TSG3nCTL4 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	TSG3nPRE	Enables or disables the peak reload timing. 0: Disables reload operation at the peak timing of the 18-bit counter. 1: Enables reload operation at the peak timing of the 18-bit counter. • The peak reload timing means the peak timing of the 18-bit counter in HT-PWM mode and the clear timing of the 18-bit counter by compare match in any mode other than HT-PWM mode. • When the reload operation at the peak timing of the 18-bit counter is disabled (TSG3nPRE = 0), reload is not executed in any mode other than HT-PWM mode.
7	TSG3nVRE	Enables or disables the trough reload timing. 0: Disables reload operation at the trough timing of the 18-bit counter. 1: Enables reload operation at the trough timing of the 18-bit counter. The setting of this bit is valid only in HT-PWM mode.
6	TSG3nPIE	Enables or disables generation of a peak interrupt (INTTSG3nIPEK). 0: Disables generation of a peak interrupt (INTTSG3nIPEK) at the peak timing of the 18-bit counter. Interrupts are not skipped. 1: Enables generation of a peak interrupt (INTTSG3nIPEK) at the peak timing of the 18-bit counter. Interrupts are skipped.
5	TSG3nVIE	Enables or disables generation of a trough interrupt (INTTSG3nIVLY). 0: Disables generation of a trough interrupt (INTTSG3nIVLY) at the trough timing of the 18-bit counter. Interrupts are not skipped. 1: Enables generation of a trough interrupt (INTTSG3nIVLY) at the trough timing of the 18-bit counter. Interrupts are skipped. The setting of this bit is valid only in HT-PWM mode.

Table 20.12 TSG3nCTL4 Register Contents (2/2)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>4 to 0</td><td>TSG3nRCC[04:00]</td><td>Specifies the skipping rate of the interrupts (INTTSG3nIPEK and INTTSG3nIVLY) and reload.
<table>
<tr><th>TSG3nRCC04</th><th>TSG3nRCC03</th><th>TSG3nRCC02</th><th>TSG3nRCC01</th><th>TSG3nRCC00</th><th>Skipping Rate</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Skipping disabled</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1/2</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1/3</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1/4</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1/30</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1/31</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1/32</td></tr>
</table>
When a write access is made (including a write of the same value to TSG3nRCC04-TSG3nRCC00) to TSG3nCTL4 during timer operation (TSG3nSTR0.TSG3nTE = 1), the interrupt skipping counter is cleared.</td></tr>
</table>

20.3.6 TSG3nCTL5 — TSG3n Control Register5

This register controls A/D conversion trigger output (TSG3nADTRG0).

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + 008_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TSG3nACC[01:00]		TSG3nAT09	TSG3nAT08	TSG3nAT07	TSG3nAT06	TSG3nAT05	TSG3nAT04	TSG3nAT03	TSG3nAT02	TSG3nAT01	TSG3nAT00
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 20.13 TSG3nCTL5 Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	TSG3nACC[01:00]	Specifies the skipping rate of the A/D conversion trigger (TSG3nADTRG0). TSG3n ACC01 / TSG3n ACC00 / Skipping Rate 0 / 0 / Skipping disabled 0 / 1 / 1/2 1 / 0 / 1/4 1 / 1 / 1/8 When a write access is made (including a write of the same value to TSG3nACC01 and TSG3nACC00) to TSG3nCTL5 during timer operation (TSG3nSTR0.TSG3nTE = 1), the interrupt skipping counter is cleared.
9	TSG3nAT09	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the (peak) timing when the 18-bit sub-counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the peak timing of the 18-bit sub-counter. 1: Enables generation of the A/D conversion trigger at the peak timing of the 18-bit sub-counter. • The TSG3nAT09 bit can be set to 1 only in HT-PWM mode. In other modes, the TSG3nAT09 bit should be set to 0. • Do not set the TSG3nAT09 bit to 1 when TSG3nDTC0W is not 0000_H and TSG3nDTC1W is 0000_H. A/D conversion trigger is not generated at the peak timing of the 18-bit sub-counter even if set so.
8	TSG3nAT08	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the (trough) timing when the 18-bit sub-counter switches from decrementing to incrementing. 0: Disables generation of the A/D conversion trigger at the trough timing of the 18-bit sub-counter. 1: Enables generation of the A/D conversion trigger at the trough timing of the 18-bit sub-counter. • The TSG3nAT08 bit can be set to 1 only in HT-PWM mode. In other modes, the TSG3nAT08 bit should be set to 0. • Do not set the TSG3nAT08 bit to 1 when TSG3nDTC0W is 0000_H and TSG3nDTC1W is not 0000_H. A/D conversion trigger is not generated at the trough timing of the 18-bit sub-counter even if set so.

Table 20.13 TSG3nCTL5 Register Contents (2/2)

Bit Position	Bit Name	Function
7	TSG3nAT07	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
6	TSG3nAT06	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value
5	TSG3nAT05	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
4	TSG3nAT04	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1Evalue. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E value.
3	TSG3nAT03	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
2	TSG3nAT02	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E value. 1: Enables generation of the A/D conversion trigger at the timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E.
1	TSG3nAT01	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the timing (peak interrupt) when the 18-bit counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIPEK) after being skipped. 1: Enables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIPEK) after being skipped.
0	TSG3nAT00	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the timing (trough interrupt) when the 18-bit counter switches from decrementing to incrementing. 0: Disables generation of the A/D conversion trigger at the timing of a trough interrupt (INTTSG3nIVLY) after being skipped. 1: Enables generation of the A/D conversion trigger at the timing of a trough interrupt (INTTSG3nIVLY) after being skipped. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.

20.3.7 TSG3nCTL6 — TSG3n Control Register 6

This register controls the A/D conversion trigger output (TSG3nADTRG1).

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $00C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TSG3nACC [11:10]		TSG3n AT19	TSG3n AT18	TSG3n AT17	TSG3n AT16	TSG3n AT15	TSG3n AT14	TSG3n AT13	TSG3n AT12	TSG3n AT11	TSG3n AT10
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 20.14 TSG3nCTL6 Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	TSG3nACC [11:10]	Specifies the skipping rate of the A/D conversion trigger (TSG3nADTRG1). TSG3n ACC11 / TSG3n ACC10 / Skipping Rate 0 / 0 / Skipping disabled 0 / 1 / 1/2 1 / 0 / 1/4 1 / 1 / 1/8 When a write access is made (including a write of the same value to TSG3nACC11 and TSG3nACC10) to TSG3nCTL5 during timer operation (TSG3nSTR0.TSG3nTE = 1), the interrupt skipping counter is cleared.
9	TSG3nAT19	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the (peak) timing when the 18-bit sub-counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the peak timing of the 18-bit sub-counter. 1: Enables generation of the A/D conversion trigger at the peak timing of the 18-bit sub-counter. • The TSG3nAT19 bit can be set to 1 only in HT-PWM mode. In other modes, the TSG3nAT19 bit should be set to 0. • Do not set the TSG3nAT19 bit to 1 when TSG3nDTC0W is not 0000_H and TSG3nDTC1W is 0000_H. A/D conversion trigger is not generated at the peak timing of the 18-bit sub-counter even if set so.
8	TSG3nAT18	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the (trough) timing when the 18-bit sub-counter switches from decrementing to incrementing. 0: Disables generation of the A/D conversion trigger at the trough timing of the 18-bit sub-counter. 1: Enables generation of the A/D conversion trigger at the trough timing of the 18-bit sub-counter. • The TSG3nAT18 bit can be set to 1 only in HT-PWM mode. In other modes, the TSG3nAT18 bit should be set to 0. • Do not set the TSG3nAT08 bit to 1 when TSG3nDTC0W is 0000_H and TSG3nDTC1W is not 0000_H. A/D conversion trigger is not generated at the trough timing of the 18-bit sub-counter even if set so.

Table 20.14 TSG3nCTL6 Register Contents (2/2)

Bit Position	Bit Name	Function
7	TSG3nAT17	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
6	TSG3nAT16	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value
5	TSG3nAT15	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
4	TSG3nAT14	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1Evalue.. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E value.
3	TSG3nAT13	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
2	TSG3nAT12	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E value. 1: Enables generation of the A/D conversion trigger at the timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E.
1	TSG3nAT11	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the timing (peak interrupt) when the 18-bit counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIPEK) after being skipped. 1: Enables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIPEK) after being skipped.
0	TSG3nAT10	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the timing (trough interrupt) when the 18-bit counter switches from decrementing to incrementing. 0: Disables generation of the A/D conversion trigger at the timing of a trough interrupt (INTTSG3nIVLY) after being skipped. 1: Enables generation of the A/D conversion trigger at the timing of a trough interrupt (INTTSG3nIVLY) after being skipped. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.

20.3.8 TSG3nCTL7 — TSG3n Control Register 7

This register sets the level of PWM output from the TSG3O1 to TSG3O6 pins at operation start (TSG3nTE is changed from 0 to 1) and at operation restart in SP-PWM mode.

This register can be written only when the SP-PWM mode is selected (TSG3nMD2-0 = 010) and when the operation is stopped (TSG3nTE = 0).

Do not rewrite this register when other modes are selected (PWM mode, HT-PWM mode, 120-DC mode, and HSP-PWM mode), or during operation (TSG3nTE = 1).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 218_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TSG3n SPSTL2	TSG3n SPSTL1	TSG3n SPSTL0
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 20.15 TSG3nCTL7 Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2	TSG3nSPSTL2	SP-PWM Mode Start Level Control Bit 2 0: TSG3nO5 (W phase) is cleared and TSG3nO6 (WB phase) is set at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode. 1: TSG3nO5 (W phase) is set and TSG3nO6 (WB phase) is cleared at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode.
1	TSG3nSPSTL1	SP-PWM Mode Start Level Control Bit 1 0: TSG3nO3 (V phase) is cleared and TSG3nO4 (VB phase) is set at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode. 1: TSG3nO3 (V phase) is set and TSG3nO4 (VB phase) is cleared at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode.
0	TSG3nSPSTL0	SP-PWM Mode Start Level Control Bit 0 0: TSG3nO1 (U phase) is cleared and TSG3nO2 (UB phase) is set at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode. 1: TSG3nO1 (U phase) is set and TSG3nO2 (UB phase) is cleared at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode.

NOTE

The settings of bits TSG3nSPSTL2 to TSG3nSPSTL0 affect output on the TSG3nO1 to TSG3nO6 pins when operation starts or is restarted. The set dead time is always inserted at these times.

20.3.9 TSG3nCTL8 — TSG3n Control Register 8

This register specifies timer output timing when input patterns are changed in 120-DC mode.

This register can be written only when 120-DC mode is selected (TSG3nMD2-0 = 011) and the timer is stopped (TSG3nTE = 0).

Do not rewrite this register in other modes (PWM mode, SP-PMW mode, HT-PWM mode, HSP-PWM mode) or while the timer is operating (TSG3nTE = 1).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + $21C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3n S120DCO
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 20.16 TSG3nCTL8 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TSG3nS120DCO	120-DC Mode Control Bit 0 0: When the input patterns are changed while 120-DC mode is selected, the main counter (TSG3nCNTE) is cleared and the change of input patterns is immediately reflected to timer output. 1: When the input patterns are changed while 120-DC mode is selected, the change of input patterns is reflected to timer output after a match of the main counter (TSG3nCNTE) with TSG3nCMP0E (from the next timer period).

CAUTION

When TSG3nS120DCO is set to 1 in 120DC mode, set the TSG3nOPT0.TSG3nSOC and TSG3nOPT2.TSG3nESSC bits to 0.

The settings of the TSG3nOPT0.TSG3nSTE and TSG3nOPT0.TSG3nPOT bits must not be changed while the timer is operating (TSG3nSTR0.TSG3nTE = 1).

20.3.10 TSG3nIOC0 — TSG3n I/O Control Register 0

This register controls the timer output pins (TSG3nO1 to TSG3nO6).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 200_H

Value after reset: $7E_H$

Bit	7	6	5	4	3	2	1	0
	—	TSG3nTOE6	TSG3nTOE5	TSG3nTOE4	TSG3nTOE3	TSG3nTOE2	TSG3nTOE1	—
Value after reset:	0	1	1	1	1	1	1	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 20.17 TSG3nIOC0 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG3nTOE6-TSG3nTOE1	Enables or disables TSG3nIOC2 controlling TSG3nO6 to TSG3nO1.(Rewriting by TSG3nIOC2 is ignored when these bits are 1.) 0: Disabled 1: Enabled
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

CAUTION

This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer.

20.3.11 TSG3nIOC1 — TSG3n I/O Control Register 1

This register controls the timer output pins (TSG3nO1 to TSG3nO6).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 204_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	TSG3nPTS	TSG3nEOC	TSG3nWOC	TSG3nTGS	TSG3nTOS
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 20.18 TSG3nIOC1 Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	TSG3nPTS	Enables or disables output of the edge detection signal (TSG3nPTE) of TSG3nPTSI0 to TSG3nPTSI2 and two-phase encoder count signal (TSG3nPEC). 0: Disables output of the toggle signal by edge detection of TSG3nPTSI0 to TSG3nPTSI2. 1: Enables output of the toggle signal by edge detection of TSG3nPTSI0 to TSG3nPTSI2.
3	TSG3nEOC	Enables or disables detection of the error condition at the motor control. 0: Disables generation of an error interrupt (INTTSG3nIER). 1: Enables generation of an error interrupt (INTTSG3nIER). For details on controlling the error interrupt, see **Section 20.4.6.1, Error Interrupt Function**.
2	TSG3nWOC	Enables or disables detection of the warning condition at the motor control. 0: Disables generation of a warning interrupt (INTTSG3nIWN). 1: Enables generation of a warning interrupt (INTTSG3nIWN). For details on the controlling generation of warning interrupt, see **Section 20.4.6.2, Warning Interrupt Function**.
1	TSG3nTGS	Selects the A/D conversion trigger diagnostic output (TSG3nO7) signal. 0: Selects A/D conversion trigger output. 1: Selects diagnostic output.
0	TSG3nTOS	Selects the timer counter increment/decrement status output (TSG3nO0) signal. 0: Outputs the up/down count flag of the 18-bit counter. 1: Outputs the up/down count flag of the 18-bit sub-counter. When TSG3nTOS is 0, the status of TSG3nSTR0.TSG3nCUF is output to TSG3nO0. When TSG3nTOS is 1, the status of TSG3nSTR0.TSG3nSUF is output to TSG3nO0. The setting of this bit is valid only in HT-PWM mode.

CAUTION

This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer.

20.3.12 TSG3nIOC2 — TSG3n I/O Control Register 2

This register controls the timer output pins (TSG3nO1 to TSG3nO6).

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + 000_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TSG3nOL6	TSG3nOL5	TSG3nOL4	TSG3nOL3	TSG3nOL2	TSG3nOL1	—	—	TSG3nTO6	TSG3nTO5	TSG3nTO4	TSG3nTO3	TSG3nTO2	TSG3nTO1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 20.19 TSG3nIOC2 Register Contents

Bit Position	Bit Name	Function
15	Reserved	When read, the value after reset is read. When writing, write the value after reset.
14 to 9	TSG3nOL6 to TSG3nOL1	Specifies the active level of TSG3nO6 to TSG3nO1 outputs. 0: Active level is high level 1: Active level is low level
8 to 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG3nTO6 to TSG3nTO1	Specifies the latch level of the output buffer of the TSG3nO6 to TSG3nO1. 0: Latch level of output buffer is low level 1: Latch level of output buffer is high level
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

CAUTION

When counting is stopped (TSG3nSTR0.TSG3nTE = 0), the TSG3nO1-6 pins maintain their previous output states. The output level should be changed by setting the TSG3nIOC0.TSG3nTOEm bit to 0, using the TSG3nTOm bit. This register can be rewritten when TSG3nIOC0.TSG3nTOEm = 0 (m = 1 to 6).

NOTE

While the timer is stopped (TSG3nSTR0.TSG3nTE = 0) and control of TSG3nOm by rewriting TSG3nIOC2 is enabled (TSG3nIOC0.TSG3nTOEm = 0), any level of the following can be output on TSG3nOm by the combination of the values of TSG3nOLm and TSG3nTOm of TSG3nIOC2.

TSG3nOLm	TSG3nTOm	Output level of TSG3nOm
0	0	Low level
0	1	High level
1	0	High level
1	1	Low level

20.3.13 TSG3nIOC3 — TSG3n I/O Control Register3

This register controls timer output pins (TSG3nO1 to TSG3nO6).

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 074_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TSG3n TOL6	TSG3n TOL5	TSG3n TOL4	TSG3n TOL3	TSG3n TOL2	TSG3n TOL1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 20.20 TSG3nIOC3 Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG3nTOL6 to TSG3nTOL1	Controls the set/clear level of output. 0: Outputs the normal level. 1: Outputs the reversed level. Setting of this bit is reflected at the start of output. The change of the output level is reflected at the next compare match timing after the change.
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

CAUTION

TSG3nTOL6 to 1 should be set to 0 in HT-PWM mode and HSP-PWM mode.

20.3.14 TSG3nSTR0 — TSG3n Status Register 0

This register consists of various status flags.

Access: This register can be read only in 8-bit units.

Address: <TSG3n_base> + 010_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TSG3nCUF	TSG3nSUF	TSG3nRSF	TSG3nTE
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 20.21 TSG3nSTR0 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read.
3	TSG3nCUF	Indicates the count direction of the 18-bit counter. 0: The 18-bit counter is incremented. 1: The 18-bit counter is decremented. TSG3nCUF is valid only in HT-PWM mode. In other modes, it is invalid (TSG3nCUF = 0).
2	TSG3nSUF	Indicates the count direction of the 18-bit sub-counter. 0: The 18-bit sub-counter is incremented. 1: The 18-bit sub-counter is decremented. • TSG3nSUF detects counting of the 18-bit sub-counter from 0000_H to (TSG3nCMP0E value - 0002_H) as up-counting, and counting from the TSG3nCMP0E value to 0002_H as down-counting. • This bit is valid only in HT-PWM mode.
1	TSG3nRSF	Indicates whether there is a reload request. 0: No reload request or reload has completed. 1: There is a reload request. • This bit is valid only in TSG3nRMC = 0. • This bit indicates that the data to be transferred next is held. • This bit is set to 1 by writing to registers to be reloaded, and cleared to 0 when reload has completed. • When TSG3nRMC is changed from 0 to 1 in HT-PWM mode, TSG3nRSF is cleared to 0. For registers to be reloaded, see **Section 20.3.1, List of Registers.**
0	TSG3nTE	Indicates the TSG3n operation status. 0: TSG3n is stopped. 1: TSG3n is operating. This bit is set when TSG3nTRG0.TSG3nTS = 1, and cleared when TSG3nTRG1.TSG3nTT = 1.

20.3.15 TSG3nSTR1 — TSG3n Status Register 1

This register consists of various status flags.

Access: This register can be read only in 8-bit units.

Address: <TSG3n_base> + 014_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TSG3nTSF	TSG3nOPF[2:0]		
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 20.22 TSG3nSTR1 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read.
3	TSG3nTSF	Indicates the pattern change order of TSG3nPTSI0 to TSG3nPTSI2. 0: Indicates that patterns are input to TSG3nPTSI0 to TSG3nPTSI2 in the normal rotation pattern order 1: Indicates that patterns are input to TSG3nPTSI0 to TSG3nPTSI2 in the reverse rotation pattern order. Normal Rotation → Reverse Rotation ← TSG3nPTSI2-TSG3nPTSI0: [1,0,1] [1,0,0] [1,1,0] [0,1,0] [0,1,1] [0,0,1] Normal or reverse rotation can be detected from the first change of TSG3nPTSI0 to TSG3nPTSI2 after TSG3nTRG0.TSG3nTS has been set to 1. For details, see **Section 20.4.3.5(b), Detection of Input Pattern Order**.
2 to 0	TSG3nOPF[2:0]	Indicates the output pattern of the timer output pins (TSG3nO1 to TSG3nO6).

20.3.16 TSG3nSTR2 — TSG3n Status Register 2

This register consists of various status flags.

Access: This register can be read only in 16-bit units.

Address: <TSG3n_base> + 018_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nTBF2	TSG3nTBF1	TSG3nTBF0	TSG3nPPF	TSG3nPEF	TSG3nTDF	TSG3nNDF	TSG3nPRF	TSG3nPTF	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 20.23 TSG3nSTR2 Register Contents (1/3)

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after reset is read.
9	TSG3nTBF2	Indicates whether the simultaneous active state of the positive phase and inverse phase is detected when TSG3nCTL1.TSG3nTBA2 is 1. 0: Positive phase (TSG3nO5) and inverse phase (TSG3nO6) are not active simultaneously. 1: Positive phase (TSG3nO5) and inverse phase (TSG3nO6) are active simultaneously. • TSG3nTBF2 is set to 1 when the simultaneous active state of the positive phase (TSG3nO5) and inverse phase (TSG3nO6) is detected, and an error interrupt (INTTSG3nIER) is generated. TSG3nTBF2 can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (operation start) or by writing 1 to TSG3nSTC.TSG3nTBR2. • The simultaneous active state is not detected when TSG3nTBA2 = 0.
8	TSG3nTBF1	Indicates whether the simultaneous active state of the positive phase and inverse phase is detected when TSG3nCTL1.TSG3nTBA1 is 1. 0: Positive phase (TSG3nO3) and inverse phase (TSG3nO4) are not active simultaneously. 1: Positive phase (TSG3nO3) and inverse phase (TSG3nO4) are active simultaneously. • TSG3nTBF1 is set to 1 when the simultaneous active state of the positive phase (TSG3nO3) and inverse phase (TSG3nO4) is detected, and an error interrupt (INTTSG3nIER) is generated. TSG3nTBF1 can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (operation start) or by writing 1 to TSG3nSTC.TSG3nTBR1. • The simultaneous active state is not detected when TSG3nTBA1 = 0.
7	TSG3nTBF0	Indicates whether the simultaneous active state of the positive phase and inverse phase is detected when TSG3nCTL1.TSG3nTBA0 is 1. 0: Positive phase (TSG3nO1) and inverse phase (TSG3nO2) are not active simultaneously. 1: Positive phase (TSG3nO1) and inverse phase (TSG3nO2) are active simultaneously. • TSG3nTBF0 is set to 1 when the simultaneous active state of the positive phase (TSG3nO1) and inverse phase (TSG3nO2) is detected, and an error interrupt (INTTSG3nIER) is generated. TSG3nTBF0 can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (operation start) or by writing 1 to TSG3nSTC.TSG3nTBR0. • The simultaneous active state is not detected when TSG3nTBA0 = 0.

Table 20.23 TSG3nSTR2 Register Contents (2/3)

Bit Position	Bit Name	Function
6	TSG3nPPF	Indicates detection of the difference between the input patterns (TSG3nPTSI0 to TSG3nPTSI2) and the output patterns (TSG3nO1 to TSG3nO6) after they are compared. 0: No phase difference detected between the input patterns (TSG3nPTSI0 to TSG3nPTSI2) and the output patterns (TSG3nO1 to TSG3nO6). 1: A phase difference detected between the input patterns (TSG3nPTSI0 to TSG3nPTSI2) and the output patterns (TSG3nO1 to TSG3nO6). • TSG3nPPF is set to 1 when a difference between input and output patterns is detected, and a warning interrupt (INTTSG3nIWN) is generated. This bit can be cleared either by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), setting TSG3nTRG0.TSG3nTS to 1 (starting the timer), input to TSG3nTSST (restarting the timer), or writing 1 to TSG3nSTC.TSG3nPPR.
5	TSG3nPEF	Indicates whether an abnormal input (000_B or 111_B is input to TSG3nPTSI0 to TSG3nPTSI2) is detected. 0: No abnormal input (000_B or 111_B) to TSG3nPTSI0 to TSG3nPTSI2 detected. 1: Abnormal input (000_B or 111_B) to TSG3nPTSI0 to TSG3nPTSI2 detected. TSG3nPEF is set to 1 when an input of 000_B or 111_B to TSG3nPTSI0 to TSG3nPTSI2 is detected, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPEF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), by setting TSG3nTRG0.TSG3nTS to 1 (starting timer operation), by input to TSG3nTSST (restarting the timer), or by writing 1 to TSG3nSTC.TSG3nPER. TSG3nPEF is valid when TSG3nCTL1.TSG3nPEC = 1.
4	TSG3nTDF	Indicates whether simultaneous generation of the TSG3nOPCI0 and TSG3nOPCI1 triggers is detected. 0: Simultaneous generation of the TSG3nOPCI0 and TSG3nOPCI1 triggers not detected. 1: Simultaneous generation of the TSG3nOPCI0 and TSG3nOPCI1 triggers detected. TSG3nTDF is set to 1 when simultaneous generation of the TSG3nOPCI0 and TSG3nOPCI1 triggers is detected, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nTDF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), by setting TSG3nTRG0.TSG3nTS to 1 (starting timer operation), by input to TSG3nTSST (restarting the timer), or by writing 1 to TSG3nSTC.TSG3nTDR. TSG3nTDF is valid when TSG3nCTL1.TSG3nTDC = 1.
3	TSG3nNDF	Indicates whether noise on TSG3nPTSI0 to TSG3nPTSI2 is detected. 0: Noise on TSG3nPTSI0 to TSG3nPTSI2 due to simultaneous change of two or more pins not detected. 1: Noise on TSG3nPTSI0 to TSG3nPTSI2 due to simultaneous change of two or more pins detected. TSG3nNDF is set to 1 when simultaneous change of two or more pins in TSG3nPTSI0 to TSG3nPTSI2 is detected, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nNDF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), by setting TSG3nTRG0.TSG3nTS to 1(starting timer operation), by input to TSG3nTSST (restarting the timer), or by writing 1 to TSG3nSTC.TSG3nNDR. TSG3nNDF is valid when TSG3nCTL1.TSG3nNDC = 1.

Table 20.23 TSG3nSTR2 Register Contents (3/3)

Bit Position	Bit Name	Function
2	TSG3nPRF	Indicates whether reversal of the TSG3nPTSI0 to TSG3nPTSI2 input order is detected. 0: The reversal of the TSG3nPTSI0 TSG3nPTSI2 input order not detected. 1: The reversal of the TSG3nPTSI0 to TSG3nPTSI2 input order detected. TSG3nPRF is set to 1 when TSG3nSTR1.TSG3nTSF changes, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPRF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (operation start), by setting TSG3nTRG0.TSG3nTS to 1 (timer operation start), by input to TSG3nTSST(timer restart), or by writing 1 to TSG3nSTC.TSG3nPRR. Detection is possible from the second TSG3nPTSI0 to TSG3nPTSI2 change timing after setting TSG3nTRG0.TSG3nTS = 1. TSG3nPRF is valid when TSG3nCTL1.TSG3nPRC = 1.
1	TSG3nPTF	Indicates whether an abnormal toggle of TSG3nPTSI0 to TSG3nPTSI2 is detected. 0: No abnormal toggle of TSG3nPTSI0 to TSG3nPTSI2 detected. 1: An abnormal toggle of TSG3nPTSI0 to TSG3nPTSI2 detected. TSG3nPTF is set to 1 when TSG3nPTSI0 to TSG3nPTSI2 (TSG3nPTE signal toggle) are changed three times or more during TSG3nOPCI0 trigger or TSG3nPTSI0 to TSG3nPTSI2 (TSG3nPTE signal toggle) are changed three times or more during TSG3nOPCI1 trigger, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPTF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), by setting TSG3nTRG0.TSG3nTS to 1 (starting timer operation), by input to TSG3nTSST (restarting the timer), or by writing 1 to TSG3nSTC.TSG3nPTR. TSG3nPTF is valid when TSG3nCTL1.TSG3nPTC[1:0] = 10_B or 11_B.
0	Reserved	When read, the value after reset is read.

20.3.17 TSG3nSTC — TSG3n Status Clear Trigger Register

This register controls the flags.

Access: This register can be written only in 16-bit units.

Address: <TSG3n_base> + $01C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nTBR2	TSG3nTBR1	TSG3nTBR0	TSG3nPPR	TSG3nPER	TSG3nTDR	TSG3nNDR	TSG3nPRR	TSG3nPTR	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W	W	W	W	W	W	W	W	R

Table 20.24 TSG3nSTC Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 10	Reserved	When writing, write the value after reset.
9	TSG3nTBR2	This is a trigger bit that clears TSG3nSTR2.TSG3nTBF2. 0: Does not clear TSG3nTBF2. 1: Clears TSG3nTBF2. When TSG3nTBR2 writing and TSG3nSTR2.TSG3nTBF2 setting occur simultaneously, TSG3nSTR2.TSG3nTBF2 setting has a priority, and the flag is not cleared.
8	TSG3nTBR1	This is a trigger bit that clears TSG3nSTR2.TSG3nTBF1. 0: Does not clear TSG3nTBF1. 1: Clears TSG3nTBF1. When TSG3nTBR1 writing and TSG3nSTR2.TSG3nTBF1 setting occur simultaneously, TSG3nSTR2.TSG3nTBF1 setting has a priority, and the flag is not cleared.
7	TSG3nTBR0	This is a trigger bit that clears TSG3nSTR2.TSG3nTBF0. 0: Does not clear TSG3nTBF0. 1: Clears TSG3nTBF0. When TSG3nTBR0 writing and TSG3nSTR2.TSG3nTBF0 setting occur simultaneously, TSG3nSTR2.TSG3nTBF0 setting has a priority, and the flag is not cleared.
6	TSG3nPPR	This is a trigger bit that clears TSG3nSTR2.TSG3nPPF. 0: Does not clear TSG3nPPF. 1: Clears TSG3nPPF. When TSG3nPPR writing and TSG3nSTR2.TSG3nPPF setting occur simultaneously, TSG3nSTR2.TSG3nPPF setting has a priority, and the flag is not cleared.
5	TSG3nPER	This is a trigger bit that clears TSG3nSTR2.TSG3nPEF. 0: Does not clear TSG3nPEF. 1: Clears TSG3nPEF. When TSG3nPER writing and TSG3nSTR2.TSG3nPEF setting occur simultaneously, TSG3nSTR2.TSG3nPEF setting has a priority, and the flag is not cleared.

Table 20.24 TSG3nSTC Register Contents (2/2)

Bit Position	Bit Name	Function
4	TSG3nTDR	This is a trigger bit that clears TSG3nSTR2.TSG3nTDF. 0: Does not clear TSG3nTDF. 1: Clears TSG3nTDF. When TSG3nTDR writing and TSG3nSTR2.TSG3nTDF setting occur simultaneously, TSG3nSTR2.TSG3nTDF setting has a priority, and the flag is not cleared.
3	TSG3nNDR	This is a trigger bit that clears TSG3nSTR2.TSG3nNDF. 0: Does not clear TSG3nNDF. 1: Clears TSG3nNDF. When TSG3nNDR writing and TSG3nSTR2.TSG3nNDF setting occur simultaneously, TSG3nSTR2.TSG3nNDF setting has a priority, and the flag is not cleared.
2	TSG3nPRR	This is a trigger bit that clears TSG3nSTR2.TSG3nPRF. 0: Does not clear TSG3nPRF. 1: Clears TSG3nPRF. When TSG3nPRR writing and TSG3nSTR2.TSG3nPRF setting occur simultaneously, TSG3nSTR2.TSG3nPRF setting has a priority, and the flag is not cleared.
1	TSG3nPTR	This is a trigger bit that clears TSG3nSTR2.TSG3nPTF. 0: Does not clear TSG3nPTF. 1: Clears TSG3nPTF. When TSG3nPTR writing and TSG3nSTR2.TSG3nPTF setting occur simultaneously, TSG3nSTR2.TSG3nPTF setting has a priority, and the flag is not cleared.
0	Reserved	When writing, write the value after reset (or a fixed value).

20.3.18 TSG3nOPT0 — TSG3n Option Register 0

This register sets the optional functions.

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 020_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	TSG3nSOC	TSG3nSTE	TSG3nPOT	TSG3nPSS	TSG3nIDC	TSG3nPSC	—
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W*1	R/W*1	R/W	R/W	R/W	R

Note 1. TSG32 is not writable.

Table 20.25 TSG3nOPT0 Register Contents (1/2)

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6	TSG3nSOC	Enables or disables control of the timer output (TSG3nO1 to TSG3nO6 pins) by software. 0: Disables control by software. 1: Enables control by software. When TSG3nSOC is set to 1, timer output is switched to the software control/trigger control output pattern specified by TSG3nSPC2 to TSG3nSPC0. The dead time is secured by the dead time counter.
5	TSG3nSTE	Enables or disables control by the pattern output trigger. 0: Disables the TSG3nPTSI0 to TSG3nPTSI2 and TSG3nOPCI0, and TSG3nOPCI1 inputs. 1: Enables TSG3nPTSI0 to TSG3nPTSI2 and TSG3nOPCI0, and TSG3nOPCI1 inputs. • The pattern output trigger is selected by TSG3nPOT. • TSG3nSTE is valid in 120-DC mode and when software output control function is enabled.
4	TSG3nPOT	Selects the pattern output trigger. 0: Switches the output pattern by the external pattern input pins (TSG3nPTSI0 to TSG3nPTSI2) (pattern switch method). 1: Switches the output pattern by the rising edge of the TSG3nOPCI0 and TSG3nOPCI1 (trigger switch method).
3	TSG3nPSS	Selects the pattern output order switch factor. 0: The pattern output order is not switched by TSG3nPSC. 1: The pattern output order is switched by TSG3nPSC.
2	TSG3nIDC	Determines the output pattern from the TSG3nO1 to TSG3nO6 pins in combination with the TSG3nIDC and TSG3nSTR1.TSG3nTSF and TSG3nPSC signals. For the timer output order and patterns to be output, see **Figure 20.88** to **Figure 20.91**, Example of Operation in 120-DC Mode, in **Section 20.4.7.6(5), Operation in 120-DC Mode**.

Table 20.25 TSG3nOPT0 Register Contents (2/2)

Bit Position	Bit Name	Function
1	TSG3nPSC	Selects the pattern output order when the semi-automatic cruise function is enabled. 0: Switches the timer output (TSG3nO1 to TSG3nO6) in the normal rotation. 1: Switches the timer output (TSG3nO1 to TSG3nO6) in the reverse rotation. • TSG3nPSC specifies the timer output pattern order assuming the output pattern specified by TSG3nSPC2 to TSG3nSPC0 as the initial pattern. TSG3nPSC is valid when TSG3nPOT = 1 and TSG3nPSS = 1. • It is recommended to rewrite TSG3nPSC when TSG3nSTR0.TSG3nTE = 0 or TSG3nPOT = 0. If TSG3nPSC is rewritten when TSG3nPOT = 1, the unexpected timer output pattern might be caused. • If the signal input to TSG3nPTSI0 to TSG3nPTSI2 changes with TSG3n operation being stopped (TSG3nSTR0.TSG3nTE = 0), the TSG3nTRG0.TSG3nTS bit should be set to 1 after matching the input signal change logic with the TSG3nPSC order. • For output order in normal or reverse rotation, see **Section 20.4.7.6, 120-DC Mode**. Here, normal rotation and reverse rotation refer to the change of output, and they are different from normal rotation and reverse rotation of a motor.
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

20.3.19 TSG3nOPT1 — TSG3n Option Register 1

This register sets the optional functions.

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 024_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TSG3nSPC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 20.26 TSG3nOPT1 Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2 to 0	TSG3nSPC[2:0]	Specifies the timer output pattern when software output function is enabled and in 120-DC mode. For the output pattern, see **Section 20.4.7.10, Software Output Control Function**, and **Section 20.4.7.6, 120-DC Mode**.

20.3.20 TSG3nOPT2 — TSG3n Option Register 2

This register sets rectangular waveform output from EMU3 to be used as PWM output by TSG3. This register can be set to 1 only when HT-PWM mode is selected (TSG3nMD2-0 = 001) and the timer is stopped (TSG3nRMC = 0). Do not rewrite this register in other modes (PWM mode, SP-PMW mode, 120-DC mode, HSP-PWM mode) or while the timer is operating (TSG3nRMC = 1).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + $03C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nESSC
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W*1

Note 1. TSG32 is not writable.

Table 20.27 TSG3nOPT2 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TSG3nESSC	EMU3 Rectangular Waveform Select Control 0: Timer output of TSG3 is set to HT-PWM mode selected by TSG3nMD2-0. 1: Timer output of TSG3 is switched to U-phase output pattern value, V-phase output pattern value, or W-phase output pattern value, input from EMU3 (ESW function). When TSG3nESSC is changed from 0 to 1, the timer output is switched to U-phase output pattern value, V-phase output pattern value, or W-phase output pattern value immediately. When TSG3nESSC is changed from 1 to 0, the timer output is switched to HT-PWM mode at the next reload timing.

20.3.21 TSG3nOPT2BF — TSG3n Option 2 Buffer Register

TSG3nOPT2BF is the buffer register for TSG3nOPT2. The value set in TSG3nOPT2 is captured at the timing described below.

TSG3nOPT2BF is a read-only register that indicates whether the output from TSG3nO1-6 is the rectangular waveform from EMU3. Writing to this register is ignored.

Access: This register can be read only in 8-bit units.

Address: <TSG3n_base> + $0CC_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nESSCBF
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 20.28 TSG3nOPT2BF Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read.
0	TSG3nESSCBF	EMU3 Rectangular Waveform Select Control Buffer 0: Timer output of TSG3 (TSG3nO1-6) is set to the PWM output mode selected by TSG3nMD2-0. 1: Timer output of TSG3 (TSG3nO1-6) is switched to the output using ESW function (dead time is added to U-phase output pattern value, V-phase output pattern value, or W-phase output pattern value, input from EMU3).

Capture Timing

- At operation start (1 is written to TSG3nTS when TSG3nTE = 0).
- At restart (1 is written to TSG3nTS when TSG3nTE = 1).
- At rising of the next PCLK if TSG3nESSC is changed from 0 to 1.
- At the next reload timing if TSG3nESSC is changed from 1 to 0.

20.3.22 TSG3nTRG0 — TSG3n Trigger Register 0

This register controls the start of the timer.

Access: This register can be written only in 8-bit units.

Address: <TSG3n_base> + 030_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nTS
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.29 TSG3nTRG0 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TSG3nTS	This bit is a trigger bit that controls the start of the timer. 0: The timer is not started. 1: The timer is started (restarted if TSG3nSTR0.TSG3nTE = 1). When restarted, the 18-bit counter is initialized. This bit is always read as 0.

20.3.23 TSG3nTRG1 — TSG3n Trigger Register 1

This register controls the stop of the timer.

Access: This register can be written only in 8-bit units.

Address: <TSG3n_base> + 034_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nTT
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.30 TSG3nTRG1 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TSG3nTT	This is a trigger bit that controls the stop of the timer. 0: The timer is not stopped. 1: The timer is stopped (TSG3nSTR0.TSG3nTE = 0). This bit is always read as 0.

20.3.24 TSG3nTRG2 — TSG3n Trigger Register 2

TSG3nTRG2 is a trigger bit to reflect the PWM duty setting to TSG3nO1-6 in anytime rewrite mode of HT-PWM mode.

This register can be set to 1 only in HT-PWM mode and when anytime rewrite mode is selected (TSG3nRMC = 1). Do not rewrite this register in other modes (PWM mode, SP-PMW mode, 120-DC mode, HSP-PWM mode) or when reload mode is selected (TSG3nRMC = 0).

Access: This register can be written only in 8-bit units.

Address: <TSG3n_base> + 038_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nIMT
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.31 TSG3nTRG2 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TSG3nIMT	Anytime Rewrite Trigger 0: Disabled 1: Changes in duty settings for U, V, and W phases are reflected to timer output in HT-PWM mode and when anytime rewrite mode is selected.

20.3.25 TSG3nCNT — TSG3n Counter Read Buffer Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nCNTE.

For the operation of this register, see **Section 20.3.26, TSG3nCNTE — TSG3n Bit-Extended Counter Read Buffer Register**.

Access: This register can be read only in 16-bit units.

Address: <TSG3n_base> + 028_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit counter															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

20.3.26 TSG3nCNTE — TSG3n Bit-Extended Counter Read Buffer Register

The counter values can be read from this register. This register mirrors the contents of TSG3nCNT from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read only in 32-bit units.

Address: <TSG3n_base> + $1A0_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit counter	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit counter															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

18-bit counter

This register is a timer read buffer register from which the 18-bit counter value can be read. In HT-PWM mode, the 18-bit counter provides the triangular waveform control in which the counter value is incremented and decremented by 2. Bit 0 is always read as 0.

In other modes, the 18-bit counter provides the sawtooth waveform control in which the counter value is incremented by 1.

Table 20.32 TSG3nCNTE Register Count Value

Operating mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG3nDTC0	TSG3nDTC0	TSG3nDTC0 + TSG3nCMP0E*1
Other modes	00000_H	00000_H	TSG3nCMP0E

Note 1. Set the value as TSG3nDTC0+TSG3nCMP0E < $3FFFF_H$.

20.3.27 TSG3nSBC — TSG3n Sub-Counter Read Buffer Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nSBCE.

For the operation of this register, see **Section 20.3.28,** TSG3nSBCE — TSG3n Bit Extended Sub-Counter Read Buffer Register.

Access: This register can be read only in 16-bit units.

Address: <TSG3n_base> + $02C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit sub-counter															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

20.3.28 TSG3nSBCE — TSG3n Bit Extended Sub-Counter Read Buffer Register

The sub-counter values can be read from this register. This register mirrors the contents of TSG3nSBC from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read only in 32-bit units.

Address: <TSG3n_base> + $1A4_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit sub-counter	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit sub-counter															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

18-bit counter

This register is a timer read buffer register from which the 18-bit sub-counter value can be read. In HT-PWM mode, the 18-bit counter provides the triangular waveform control in which the counter value is incremented and decremented by 2. Bit 0 is always read as 0. (Available only in HT-PWM mode.).

Table 20.33 TSG3nSBCE Register Count Value

Operating mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG3nDTC0	00000_H	TSG3nDTC0 + TSG3nDTC1 + TSG3nCMP0E*1
Other modes	00000_H	00000_H	00000_H

Note 1. Set the value as TSG3nDTC0 + TSG3nDTC1 + TSG3nCMP0E < $3FFFF_H$.

20.3.29 TSG3nCMP0 — TSG3n Compare Register 0

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP0E.

For the operation of this register, see **Section 20.3.30, TSG3nCMP0E — TSG3n Bit Extended Compare Register 0**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 058_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.30 TSG3nCMP0E — TSG3n Bit Extended Compare Register 0

This register is an 18-bit compare register that specifies the PWM period in all modes. This register mirrors the contents of TSG3nCMP0 from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $14C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 20.34 TSG3nCMP0E Register Setting

Operating mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG3nCMP0E*1	00002_H	$3FFFE_H$
Other modes	TSG3nCMP0E + 1	1 (TSG3nCMP0E = 00000_H)	40000_H (TSG3nCMP0E = $3FFFF_H$)

Note 1. In HT-PWM mode, the least significant bit is ignored.

20.3.31 TSG3nCMP1W — TSG3n Compare Register 1, 2

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP1E and TSG3nCMP2E.

For the operation of this register, see **Section 20.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 040_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP2 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP1 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.32 TSG3nCMP3W — TSG3n Compare Register 3, 4

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP3E and TSG3nCMP4E.

For the operation of this register, see **Section 20.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $04C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP4 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP3 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.33 TSG3nCMP5W — TSG3n Compare Register 5, 6

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP5E and TSG3nCMP6E.

For the operation of this register, see **Section 20.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 044_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP6 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP5 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.34 TSG3nCMP7W — TSG3n Compare Registers 7, 8

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP7E and TSG3nCMP8E.

For the operation of this register, see **Section 20.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12.**

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 050_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP8 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP7 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.35 TSG3nCMP9W — TSG3n Compare Registers 9, 10

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP9E and TSG3nCMP10E.

For the operation of this register, see **Section 20.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 048_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP10 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP9 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.36 TSG3nCMP11W — TSG3n Compare Registers 11, 12

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP11E and TSG3nCMP12E.

For the operation of this register, see **Section 20.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 054_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP12 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP11 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.37 TSG3nCMP1 to TSG3nCMP12 — TSG3n Compare Registers 1 to 12

These registers can access the 16 lower-order bits of the 18-bit registers TSG3nCMP1E-12E.

For the operation of these registers, see **Section 20.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 16-bit units.

Address: TSG3nCMP1 <TSG3n_base> + 080_H
TSG3nCMP2 <TSG3n_base> + 084_H
TSG3nCMP3 <TSG3n_base> + 098_H
TSG3nCMP4 <TSG3n_base> + $09C_H$
TSG3nCMP5 <TSG3n_base> + 088_H
TSG3nCMP6 <TSG3n_base> + $08C_H$
TSG3nCMP7 <TSG3n_base> + $0A0_H$
TSG3nCMP8 <TSG3n_base> + $0A4_H$
TSG3nCMP9 <TSG3n_base> + 090_H
TSG3nCMP10 <TSG3n_base> + 094_H
TSG3nCMP11 <TSG3n_base> + $0A8_H$
TSG3nCMP12 <TSG3n_base> + $0AC_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.38 TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12

The compare value is set by these registers. These registers mirror the contents of TSG3nCMP1-12, TSG3nCMP1W, 3W, 5W, 7W, 9W, and 11W from which the 16 lower-order bits of these registers can be accessed.

Access: This register can be read/written in 32-bit units.

Address: TSG3nCMP1E <TSG3n_base> + $17C_H$
TSG3nCMP2E <TSG3n_base> + 178_H
TSG3nCMP3E <TSG3n_base> + 164_H
TSG3nCMP4E <TSG3n_base> + 160_H
TSG3nCMP5E <TSG3n_base> + 174_H
TSG3nCMP6E <TSG3n_base> + 170_H
TSG3nCMP7E <TSG3n_base> + $15C_H$
TSG3nCMP8E <TSG3n_base> + 158_H
TSG3nCMP9E <TSG3n_base> + $16C_H$
TSG3nCMP10E <TSG3n_base> + 168_H
TSG3nCMP11E <TSG3n_base> + 154_H
TSG3nCMP12E <TSG3n_base> + 150_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 20.35 TSG3nCMP1E-TSG3nCMP12E Register Setting (1/2)

Register	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode	HSP-PWM Mode
TSG3nCMP1E	TSG3nO1 clear timing	TSG3nO1 clear timing / TSG3nO2 set timing		Duty when TSG3nO1, 3, or 5 output pattern is selected by TSG3nPAT0.	TSG3nO1 clear timing
TSG3nCMP2E	TSG3nO1 set timing	TSG3nO1 set timing / TSG3nO2 clear timing			TSG3nO1 set timing
TSG3nCMP3E	TSG3nO2 clear timing	—		Duty whenTSG3nO2, 4, or 6 output pattern is selected by TSG3nPAT1.	TSG3nO2 clear timing
TSG3nCMP4E	TSG3nO2 set timing	—			TSG3nO2 set timing
TSG3nCMP5E	TSG3nO3 clear timing	TSG3nO3 clear timing / TSG3nO4 set timing		Duty when TSG3nO1, 3, or 5 output pattern is selected by TSG3nPAT0.	TSG3nO3 clear timing
TSG3nCMP6E	TSG3nO3 set timing	TSG3nO3 set timing / TSG3nO4 clear timing			TSG3nO3 set timing

Table 20.35 TSG3nCMP1E-TSG3nCMP12E Register Setting (2/2)

Register	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode	HSP-PWM Mode
TSG3nCMP7E	TSG3nO4 clear timing	—		Duty when TSG3nO2, 4, or 6 output pattern is selected by TSG3nPAT1.	TSG3nO4 clear timing
TSG3nCMP8E	TSG3nO4 set timing	—			TSG3nO4 set timing
TSG3nCMP9E	TSG3nO5 clear timing	TSG3nO5 clear timing / TSG3nO6 set timing		Duty when TSG3nO1, 3, or 5 output pattern is selected by TSG3nPAT0.	TSG3nO5 clear timing
TSG3nCMP10E	TSG3nO5 set timing	TSG3nO5 set timing / TSG3nO6 clear timing			TSG3nO5 set timing
TSG3nCMP11E	TSG3nO6 clear timing	—		Duty when TSG3nO2, 4, or 6 output pattern is selected by TSG3nPAT1.	TSG3nO6 clear timing
TSG3nCMP12E	TSG3nO6 set timing	—			TSG3nO6 set timing

NOTE

The dead time function is used in all operating modes.

In HT-PWM mode, a compare match occurs not only in TSG3nCNTE but also in TSG3nSBCE.

In 120-DC mode, the output from TSG3nO1-6 is controlled by the TSG3nCMPmE, TSG3nPAT0, and TSG3nPAT1 registers.

20.3.39 TSG3nDCMP0W — TSG3n Diagnostic Output Compare Register 0, 1

This register can access the 16 lower-order bits of the 18-bit register TSG3nDCMP0E and TSG3nDCMP1E.

For the operation of this register, see **Section 20.3.41, TSG3nDCMP0E to 2E — TSG3n Bit Extended Diagnostic Output Compare Register 0 to 2**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $05C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nDCMP1(16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nDCMP0(16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.40 TSG3nDCMP2 — TSG3n Diagnostic Output Compare Register 2

This register can access the 16 lower-order bits of the 18-bit register TSG3nDCMP2E.

For the operation of this register, see **Section 20.3.41, TSG3nDCMP0E to 2E — TSG3n Bit Extended Diagnostic Output Compare Register 0 to 2**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 060_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nDCMP2 (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.41 TSG3nDCMP0E to 2E — TSG3n Bit Extended Diagnostic Output Compare Register 0 to 2

The compare value is set by these registers. These registers mirror the contents of TSG3nDCMP0W and TSG3nDCMP2 from which the 16 lower-order bits of these registers can be accessed.

Access: This register can be read/written in 32-bit units.

Address: TSG3nDCMP0E <TSG3n_base> + 148_H

TSG3nDCMP1E <TSG3n_base> + 144_H

TSG3nDCMP2E <TSG3n_base> + 140_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

These registers control the diagnostic output timing or AD conversion trigger timing in all modes. A pulse is generated at the match timing of the 18-bit counter value with this register.

20.3.42 TSG3nPAT0W — TSG3n Pattern Register 0

This register specifies the output pattern.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 064_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PAT5T	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT5T	PAT4T			PAT3T			PAT2T			PAT1T			PAT0T		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output Pattern

This register controls UT/VT/WT output in 120-DC mode.

Table 20.36 TSG3nPAT0W Register Setting value and Output Control

PATmT Value	Output Control
000	Fixed Low level
001	PWM output set by TSG3nCMP1E
010	PWM output set by TSG3nCMP2E
011	PWM output set by TSG3nCMP5E
100	PWM output set by TSG3nCMP6E
101	PWM output set by TSG3nCMP9E
110	PWM output set by TSG3nCMP10E
111	Fixed High level

Note: m = 0, 1, 2, 3, 4, 5

20.3.43 TSG3nPAT1W — TSG3n Pattern Register 1

This register specifies the output pattern.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 068_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PAT5B	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT5B	PAT4B			PAT3B			PAT2B			PAT1B			PAT0B		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output Pattern

This register controls UB/VB/WB output in 120-DC mode.

Table 20.37 TSG3nPAT1W Register Setting value and Output Control

PATmB Value	Output Control
000	Fixed Low level
001	PWM output set by TSG3nCMP3E
010	PWM output set by TSG3nCMP4E
011	PWM output set by TSG3nCMP7E
100	PWM output set by TSG3nCMP8E
101	PWM output set by TSG3nCMP11E
110	PWM output set by TSG3nCMP12E
111	Fixed High level

Note: m = 0, 1, 2, 3, 4, 5

20.3.44 TSG3nDTC0W — TSG3n Dead Time Setting Register 0

This register sets the dead time value (the period from inverse phase inactivation to positive phase activation).

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $06C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	Write Protection Code Check														
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nDTC0(10-bit dead time compare)									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

To rewrite TSG3nDTC0W[0:9], set bit 14 to bit 0 and TSG3nDTCM to 0 in TSG3nDTPR, and rewrite the TSG3nDTC0W. At this time, when the rewritten value of TSG3nDTC0W[30:16] and the TSG3nDTPR value match, TSG3nDTC0W is rewritten.

During timer operation (TSG3nSTR0.TSG3nTE = 1), rewriting should be performed in reload mode (TSG3nCTL3.TSG3nRMC = 0).

20.3.45 TSG3nDTC1W — TSG3n Dead Time Setting Register 1

This register sets the dead time (the period from positive phase inactivation to inverse phase activation).

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 070_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	Write Protection Code Check														
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nDTC1(10-bit dead time compare)									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

To rewrite TSG3nDTC1W[0:9], set bit 14 to bit 0 and TSG3nDTCM to 0 in TSG3nDTPR, and rewrite the TSG3nDTC1W. At this time, when the rewritten value of TSG3nDTC1W[30:16] and the TSG3nDTPR value match, TSG3nDTC1W is rewritten.

During timer operation (TSG3nSTR0.TSG3nTE = 1), rewriting should be performed in reload mode (TSG3nCTL3.TSG3nRMC = 0).

20.3.46 TSG3nCMPU — TSG3n HT-PWM U Phase Compare Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMPUE.

For the operation of this register, see **Section 20.3.49, TSG3nCMPUE — TSG3n Bit Extended HT-PWM U Phase Compare** Register.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0B0_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMPU(16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.47 TSG3nCMPV — TSG3n HT-PWM V Phase Compare Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMPVE.

For the operation of this register, see **Section 20.3.50, TSG3nCMPVE — TSG3n Bit Extended HT-PWM V Phase Compare** Register.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0B4_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMPV (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.48 TSG3nCMPW — TSG3n HT-PWM W Phase Compare Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMPWE.

For the operation of this register, see **Section 20.3.51, TSG3nCMPWE — TSG3n Bit Extended HT-PWM W Phase Compare** Register.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0B8_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMPW (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.49 TSG3nCMPUE — TSG3n Bit Extended HT-PWM U Phase Compare Register

This register sets the compare value for U phase in HT-PWM. In addition to the functions of TSG3nCMP1E and TSG3nCMP2E, this register can access specific registers.

The data written to this register is stored in the TSG3nCMP1E and TSG3nCMP2E registers which allows a generation of symmetry triangular waveform of PWM by one write access (see **Figure 20.3**). When this register is read, the same value as TSG3nCMP1E is returned. This register mirrors the contents of TSG3nCMPU from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 188_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

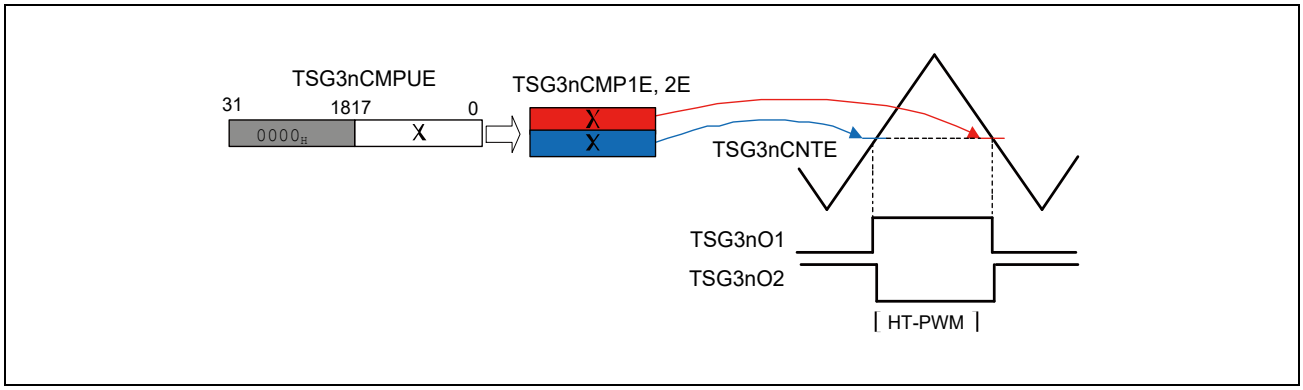


Figure 20.3 TSG3nCMPUE Register Accesses

20.3.50 TSG3nCMPVE — TSG3n Bit Extended HT-PWM V Phase Compare Register

This register sets the compare value for V phase in HT-PWM. In addition to the functions of TSG3nCMP5E and TSG3nCMP6E, this register can access specific registers.

The data written to this register is stored in the TSG3nCMP5E and TSG3nCMP6E registers which allows a generation of symmetry triangular waveform of PWM by one write access (see **Figure 20.4**). When this register is read, the same value as TSG3nCMP5E is returned. This register mirrors the contents of TSG3nCMPV from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 184_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

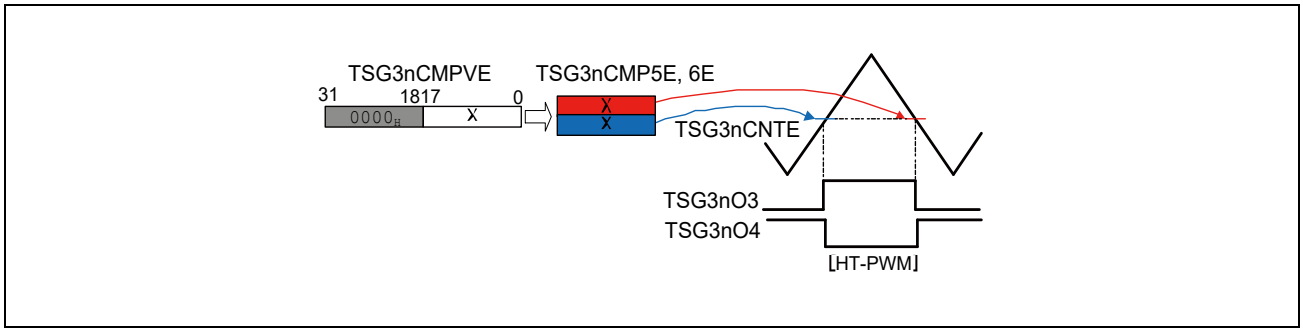


Figure 20.4 TSG3nCMPVE Register Accesses

20.3.51 TSG3nCMPWE — TSG3n Bit Extended HT-PWM W Phase Compare Register

This register sets the compare value for W phase in HT-PWM. In addition to the functions of TSG3nCMP9E and TSG3nCMP10E, this register can access specific registers.

The data written to this register is stored in the TSG3nCMP9E and TSG3nCMP10E registers which allows a generation of symmetry triangular waveform of PWM by one write access (see **Figure 20.5**). When this register is read, the same value as TSG3nCMP9E is returned. This register mirrors the contents of TSG3nCMPW from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 180_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

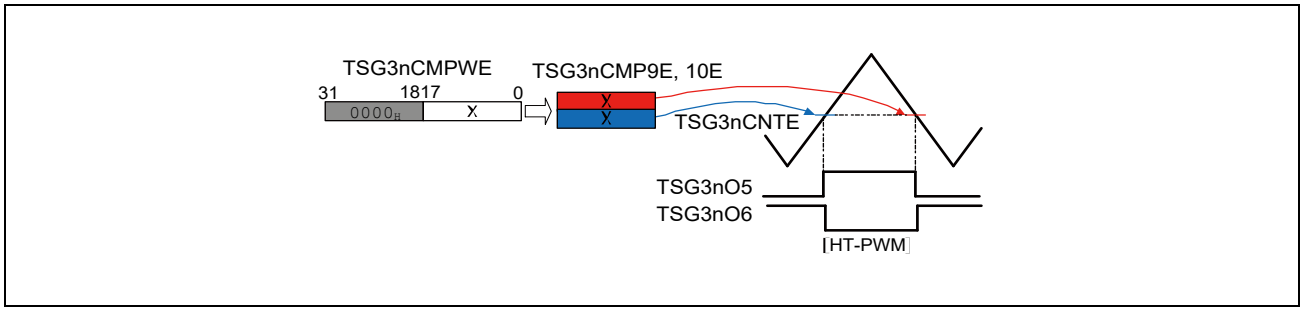


Figure 20.5 TSG3nCMPWE Register Accesses

20.3.52 TSG3nUPW — TSG3n SP-PWM U Phase Active Width Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nUPWE.

For the operation of this register, see **Section 20.3.55, TSG3nUPWE — TSG3n Bit Extended SP-PWM U Phase Active Width** Register.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0_B C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nUPW (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.53 TSG3nVPW — TSG3n SP-PWM V Phase Active Width Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nVPWE.

For the operation of this register, see **Section 20.3.56, TSG3nVPWE — TSG3n Bit Extended SP-PWM V Phase Active Width** Register.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0C0_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nVPW (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.54 TSG3nWPW — TSG3n SP-PWM W Phase Active Width Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nWPWE.

For the operation of this register, see **Section 20.3.57, TSG3nWPWE — TSG3n Bit Extended SP-PWM W Phase Active Width** Register.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0C4_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nWPW (16-bit compare register)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.55 TSG3nUPWE — TSG3n Bit Extended SP-PWM U Phase Active Width Register

This register sets the active width for U phase in SP-PWM mode. The sum of the TSG3nUPWE value and the TSG3nCMP2E value is stored in TSG3nCMP1E (see **Figure 20.6**). When this register is read, the same value as TSG3nCMP1E is returned. This register mirrors the contents of TSG3nUPW from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 198_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

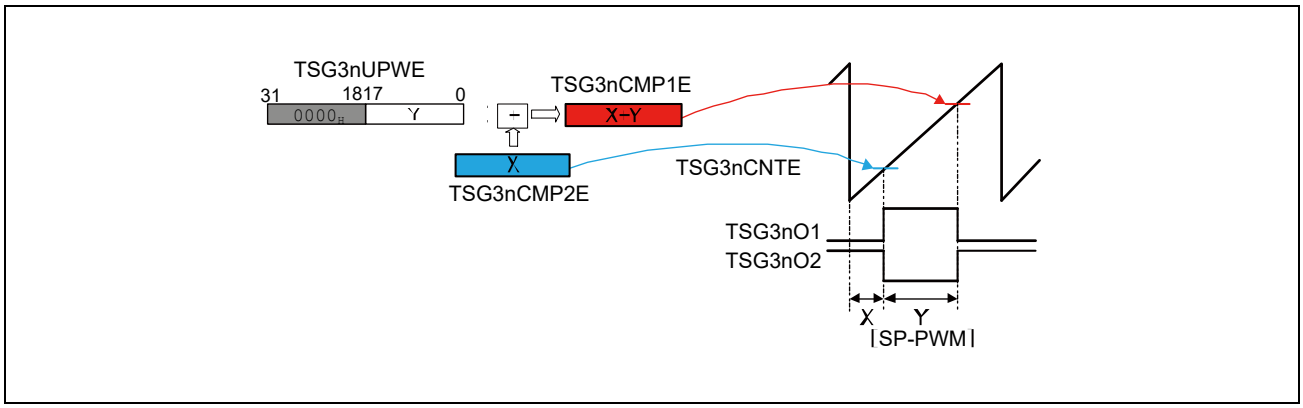


Figure 20.6 TSG3nUPWE Register Accesses

20.3.56 TSG3nVPWE — TSG3n Bit Extended SP-PWM V Phase Active Width Register

This register sets the active width for V phase in SP-PWM mode. The sum of the TSG3nVPWE value and the TSG3nCMP6E value is stored in TSG3nCMP5E (see **Figure 20.7**). When this register is read, the same value as TSG3nCMP5E is returned. This register mirrors the contents of TSG3nVPW from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 194_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

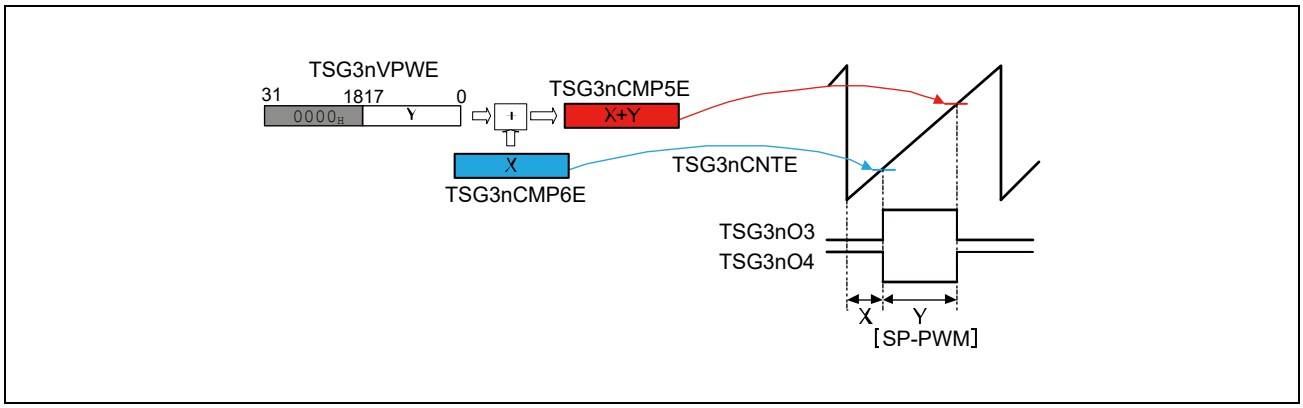


Figure 20.7 TSG3nVPWE Register Accesses

20.3.57 TSG3nWPWE — TSG3n Bit Extended SP-PWM W Phase Active Width Register

This register sets the active width for W phase in SP-PWM mode. The sum of the TSG3nWPWE value and the TSG3nCMP10E value is stored in TSG3nCMP9E (see **Figure 20.8**). When this register is read, the same value as TSG3nCMP9E is returned. This register mirrors the contents of TSG3nWPW from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 190_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

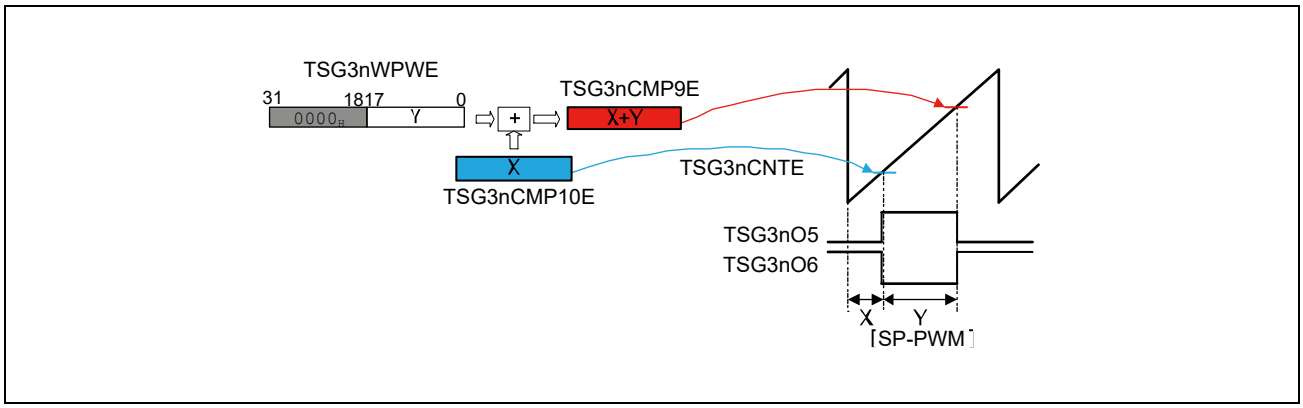


Figure 20.8 TSG3nWPWE Register Accesses

20.3.58 TSG3nHSPCMUE — TSG3n HSP-PWM Mode U Phase Compare Register

This register sets the PWM output width for U phase in HSP-PWM mode.

When a write access is made to this register, the values are set to TSG3nCMP1E to TSG3nCMP4E registers according to the formulas described in **Section 20.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

When this register is read, the same value as TSG3nCMP1E is returned.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 134_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.59 TSG3nHSPCMVE — TSG3n HSP-PWM Mode V Phase Compare Register

This register sets the PWM output width for V phase in HSP-PWM mode.

When a write access is made to this register, the values are set to TSG3nCMP5E to TSG3nCMP8E registers according to the formulas described in Section **20.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

When this register is read, the same value as TSG3nCMP5E is returned.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 130_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.60 TSG3nHSPCMWE — TSG3n HSP-PWM Mode W Phase Compare Register

This register sets the PWM output width for W phase in HSP-PWM mode.

When a write access is made to this register, the values are set to TSG3nCMP9E to TSG3nCMP12E registers according to the formulas described in Section **20.4.7.8, Compare Register Set Value in HSP-PWM Mode.**

When this register is read, the same value as TSG3nCMP9E is returned.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $12C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.61 TSG3nHSPSHUE — TSG3n HSP-PWM Mode U Phase Shift Register

This register sets the PWM shift width for U phase in HSP-PWM mode.

When a write access is made to TSG3nHSPCMUE after setting this register, the values are set to TSG3nCMP1E to TSG3nCMP4E registers according to the formulas described in Section **20.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 128_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit shift register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit shift register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.62 TSG3nHSPSHVE — TSG3n HSP-PWM Mode V Phase Shift Mode Register

This register sets the PWM shift width for V phase in HSP-PWM mode.

When a write access is made to TSG3nHSPCMVE after setting this register, the values are set to TSG3nCMP5E to TSG3nCMP8E registers according to the formulas described in **Section 20.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 124_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit shift register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit shift register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.63 TSG3nHSPSHWE — TSG3n HSP-PWM Mode W Phase Sift Register

This register sets the PWM shift width for U phase in HSP-PWM mode.

When a write access is made to TSG3nHSPCMWE after setting this register, the values are set to TSG3nCMP9E to TSG3nCMP12E registers according to the formulas described in **Section 20.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 120_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit shift register	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit shift register															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3.64 TSG3nDTPR — TSG3n Dead Time Protection Register

This register controls protection of the write access to the dead time register.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + 210_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nDTCM	Write Protection Code														
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 20.38 TSG3nDTPR Register Contents

Bit Position	Bit Name	Function
15	TSG3nDTCM	Enables or disables rewriting of TSG3nDTC0 and TSG3nDTC1. 0: Enables rewriting of TSG3nDTC0 and TSG3nDTC1. 1: Disables rewriting of TSG3nDTC0 and TSG3nDTC1.
14 to 0	TSG3nDTPR[14:0]	Sets the write protection code (any value from 0000 to 7FFF).

This register protects TSG3nDTC0 and TSG3nDTC1 from illegal rewriting.

Functions are described below.

- TSG3nDTCM enables or disables rewriting of TSG3nDTC0 and TSG3nDTC1.
- Rewriting of TSG3nDTC0 and TSG3nDTC1 is enabled or disabled by checking a match of the write protection code (bits 30 to 16) of TSG3nDTC0 with TSG3nDTC1 and the write protection code of TSG3nDTPR, and the TSG3nDTCM setting.

CAUTION

This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer.

20.4 Function

Table 20.39 List of Modes

TSG3nCTL0 Register			
TSG3nMD2	TSG3nMD1	TSG3nMD0	Timer Mode
0	0	0	PWM mode
0	0	1	HT-PWM mode (HT-PWM)
0	1	0	Shift pulse PWM mode (SP-PWM)
0	1	1	120-DC mode
1	0	0	High-accuracy shift pulse PWM mode (HSP-PWM)
Other than above			Setting prohibited

20.4.1 Basic Operation

20.4.1.1 Basic Operation of 18-Bit Counter

The basic operation of the 18-bit counter is described. For details, see **Section 20.4.7, Operating Modes**.

Counting start

The 18-bit counter of TSG3n starts counting in HT-PWM mode when the initial value is 00000_H and after the TSG3nDTC0 value is loaded. The counter starts counting from the initial value 00000_H in all modes except for HT-PWM mode.

In HT-PWM mode, the counter value is incremented by 2 from the value of TSG3nDTC0 and decremented by 2 up to the value of TSG3nDTC0 after the counter value matches with the value of TSG3nCMP0E + TSG3nDTC0. The counter increments from 00000_H, 00001_H, 00002_H, 00003_H, … in all modes except for HT-PWM mode.

Counter clear

The 18-bit counter is cleared by the match of the counter value and the value of TSG3nCMP0E in all modes except for HT-PWM mode. (Clearing operation is not available in HT-PWM mode.)

Counter read during counting

In the TSG3n, the 18-bit counter value during counting can be read through TSG3nCNTE.

Count stop operation

When count operation is stopped (when TS0TE is changed from 1 to 0), TSG3nCNTE and TSG3nSBCE retain the counter value when stopped.

Interrupt operation

In the TSG3n, the following interrupts are generated.

- INTTSG3nI0: A period interrupt by a match of the 18-bit counter value with the TSG3nDTC0 value in HT-PWM mode. A compare match interrupt of the 18-bit counter value with the TSG3nCMP0E buffer register in any mode other than HT-PWM mode.
- INTTSG3nI1: A compare match interrupt of the 18-bit counter value with the TSG3nCMP1E buffer register.
- INTTSG3nI2: A compare match interrupt of the 18-bit counter value with the TSG3nCMP2E buffer register.
- INTTSG3nI3: A compare match interrupt of the 18-bit counter value with the TSG3nCMP3E buffer register.
- INTTSG3nI4: A compare match interrupt of the 18-bit counter value with the TSG3nCMP4E buffer register.
- INTTSG3nI5: A compare match interrupt of the 18-bit counter value with the TSG3nCMP5E buffer register.
- INTTSG3nI6: A compare match interrupt of the 18-bit counter value with the TSG3nCMP6E buffer register.
- INTTSG3nI7: A compare match interrupt of the 18-bit counter value with the TSG3nCMP7E buffer register.
- INTTSG3nI8: A compare match interrupt of the 18-bit counter value with the TSG3nCMP8E buffer register.
- INTTSG3nI9: A compare match interrupt of the 18-bit counter value with the TSG3nCMP9E buffer register.
- INTTSG3nI10: A compare match interrupt of the 18-bit counter value with the TSG3nCMP10E buffer register.
- INTTSG3nI11: A compare match interrupt of the 18-bit counter value with the TSG3nCMP11E buffer register.
- INTTSG3nI12: A compare match interrupt of the 18-bit counter value with the TSG3nCMP12E buffer register.
- INTTSG3nIPEK: A peak interrupt when the 18-bit counter switches from incrementing to decrementing.
- INTTSG3nIVLY: A trough interrupt when the 18-bit counter switches from decrementing to incrementing
- INTTSG3nIER: A simultaneous active state detection interrupt of the positive phase and inverse phase
- INTTSG3nIWN: A warning detection interrupt

20.4.1.2 Function of Compare Registers

The functions of the compare registers in each operating mode are shown in the following tales.

Table 20.40 Compare Register Functions in Each Mode (1/7)

Operating Mode	TSG3nCMP0E	TSG3nCMP1E	TSG3nCMP2E
PWM mode	PWM period	TSG3nO1 clear timing	TSG3nO1 set timing
HT-PWM mode	PWM period	TSG3nO1 clear timing TSG3nO2 set timing	TSG3nO1 set timing TSG3nO2 clear timing
SP-PWM mode	PWM period	TSG3nO1 clear timing TSG3nO2 set timing	TSG3nO1 set timing TSG3nO2 clear timing
120-DC mode	PWM period	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0
HSP-PWM mode	PWM period	TSG3nO1 clear timing	TSG3nO1 set timing

Table 20.40 Compare Register Functions in Each Mode (2/7)

Operating Mode	TSG3nCMP3E	TSG3nCMP4E	TSG3nCMP5E	TSG3nCMP6E
PWM mode	TSG3nO2 clear timing	TSG3nO2 set timing	TSG3nO3 clear timing	TSG3nO3 set timing
HT-PWM mode	Compare match interrupt	Compare match interrupt	TSG3nO3 clear timing TSG3nO4 set timing	TSG3nO3 set timing TSG3nO4 clear timing
SP-PWM mode	—	—	TSG3nO3 clear timing TSG3nO4 set timing	TSG3nO3 set timing TSG3nO4 clear timing
120-DC mode	Select TSG3nO2, TSG3nO4, or TSG3nO6 output by TSG3nPAT1W	Select TSG3nO2, TSG3nO4, or TSG3nO6 output by TSG3nPAT1W	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0W	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0W
HSP-PWM mode	TSG3nO2 clear timing	TSG3nO2 set timing	TSG3nO3 clear timing	TSG3nO3 set timing

Table 20.40 Compare Register Functions in Each Mode (3/7)

Operating Mode	TSG3nCMP7E	TSG3nCMP8E	TSG3nCMP9E	TSG3nCMP10E
PWM mode	TSG3nO4 clear timing	TSG3nO4 set timing	TSG3nO5 clear timing	TSG3nO5 set timing
HT-PWM mode	Compare match interrupt	Compare match interrupt	TSG3nO5 clear timing TSG3nO6 set timing	TSG3nO5 set timing TSG3nO6 clear timing
SP-PWM mode	—	—	TSG3nO5 clear timing TSG3nO6 set timing	TSG3nO5 set timing TSG3nO6 clear timing
120-DC mode	Select TSG3nO2, TSG3nO4, or TSG3nO6 output by TSG3nPAT1W	Select TSG3nO2, TSG3nO4, or TSG3nO6 output by TSG3nPAT1W	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0W	TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0W
HSP-PWM mode	TSG3nO4 clear timing	TSG3nO4 set timing	TSG3nO5 clear timing	TSG3nO5 set timing

Table 20.40 Compare Register Functions in Each Mode (4/7)

Operating Mode	TSG3nCMP11E	TSG3nCMP12E	TSG3nDCMP0E	TSG3nDCMP1E
PWM mode	TSG3nO6 clear timing	TSG3nO6 set timing	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing
HT-PWM mode	Compare match interrupt	Compare match interrupt	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing
SP-PWM mode	—	—	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing
120-DC mode	Select TSG3nO2, TSG3nO4, TSG3nO6 by TSG3nPAT1W	Select TSG3nO2, TSG3nO4, TSG3nO6 by TSG3nPAT1W	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing
HSP-PWM mode	TSG3nO6 clear timing	TSG3nO6 set timing	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing

Table 20.40 Compare Register Functions in Each Mode (5/7)

Operating Mode	TSG3nDCMP2E	TSG3nCMPUE	TSG3nCMPVE	TSG3nCMPWE
PWM mode	Diagnostic output or A/D conversion trigger timing	—	—	—
HT-PWM mode	Diagnostic output or A/D conversion trigger timing	The TSG3nCMPUE set value is used as the set value of TSG3nCMP1E and TSG3nCMP2E.	The TSG3nCMPVE set value is used as the set value of TSG3nCMP5E and TSG3nCMP6E.	The TSG3nCMPWE set value is used as the set value of TSG3nCMP9E and TSG3nCMP10E.
SP-PWM mode	Diagnostic output or A/D conversion trigger timing	—	—	—
120-DC mode	Diagnostic output or A/D conversion trigger timing	—	—	—
HSP-PWM mode	Diagnostic output or A/D conversion trigger timing	—	—	—

Table 20.40 Compare Register Functions in Each Mode (6/7)

Operating Mode	TSG3nUPWE	TSG3nVPWE	TSG3nWPWE
PWM mode	—	—	—
HT-PWM mode	—	—	—
SP-PWM mode	The sum of the TSG3nUPWE set value and the TSG3nCMP2E set value is used as the TSG3nCMP1E set value.	The sum of the TSG3nVPWE set value and the TSG3nCMP2E set value is used as the TSG3nCMP1E set value.	The sum of the TSG3nWPWE set value and the TSG3nCMP2E set value is used as the TSG3nCMP1E set value.
120-DC mode	—	—	—
HSP-PWM mode	—	—	—

Table 20.40 Compare Register Functions in Each Mode (7/7)

Operating Mode	TSG3nHSPCMUE, TSG3nHSPSHUE	TSG3nHSPCMVE, TSG3nHSPSHVE	TSG3nHSPCMWE, TSG3nHSPSHWE
PWM mode	—	—	—
HT-PWM mode	—	—	—
SP-PWM mode	—	—	—
120-DC mode	—	—	—
HSP-PWM mode	TSG3nCMP1E-4E is set based on the TSG3nHSPCMUE set value and the set values in TSG3nCMP0E, TSG3nDTC0, TSG3nDTC1, and TSG3nHSPSHUE.	TSG3nCMP5E-8E is set based on the TSG3nHSPCMVE set value and the set values of TSG3nCMP0E, TSG3nDTC0, TSG3nDTC1, and TSG3nHSPSHVE.	TSG3nCMP9E-12E is set based on the TSG3nHSPCMWE set value and the set values of TSG3nCMP0E, TSG3nDTC0, TSG3nDTC1, and TSG3nHSPSHWE.

20.4.1.3 Compare Register Rewrite Operation

TSG3 can be set to reload mode or anytime rewrite mode using theTSG3nRMC bit.

Reload mode is enabled when TSG3nRMC = 0. The registers listed as “enabled” on the “Reload” column in **Section 20.3.1, List of Registers** are simultaneously updated at the reload timing.

Anytime rewrite mode is enabled when TSG3nRMC = 1. The registers are updated independently every time the value is written to the relevant register.

The update timing of the registers to be reloaded in reload mode and anytime rewrite mode in each mode are listed in the following table.

Table 20.41 Updating Timing of Compare Registers by Mode

Mode	Anytime Rewrite TSG3nRMC = 1	Reload TSG3nRMC = 0
PWM mode	TSG3nCMP0E: At the next counter clear timing of the 18-bit counter	At reload timing
	Registers other than TSG3nCMP0E: At a write access to the register	
HT-PWM mode	TSG3nCMP0E: At the next peak or trough timing of TSG3nCNTE	At reload timing
	TSG3nCMP1E, 2E, 5E, 6E, 9E, 10E: At writing 1 to the TSG3nIMT bit	
	Registers other than TSG3nCMP0E,1E, 2E, 5E, 6E, 9E, 10E: At a write access to the register	
SP-PWM mode	TSG3nCMP0E: At the next counter clear timing of the 18-bit counter	At reload timing
	Registers other than TSG3nCMP0E: At a write access to the register	
120-DC mode	Setting prohibited	At reload timing
HSP-PWM mode	Setting prohibited	At reload timing

Anytime Rewrite Mode

In this mode, the compare registers are rewritten independently. Whenever a value is written to the compare register, the written value is reflected at the timing of **Table 20.41**.

Reload mode (Simultaneous Rewrite Function)

Writing to TSG3nCMP1E (TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE) enables reload (sets the reload request flag TSG3nSTR0.TSG3nRSF), and the values of all the pertinent registers are updated simultaneously at the next reload timing (reload).

The reload timing is the peak or trough timing of the 18-bit counter when the TSG3nTRG0.TSG3nTS bit is changed from 0 to 1. Reloading is controlled by TSG3nCTL4.TSG3nPRE, and TSG3nVRE.

Writing to any register other than TSG3nCMP1E (TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE) does not enable reloading.

Do not write to the registers to be reloaded until the next reload timing after reloading is enabled by writing to TSG3nCMP1E (TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE).

The pertinent registers should be rewritten when the reload request flag (TSG3nSTR0.TSG3nRSF) is 0.

Rewriting registers to be reloaded by DMA transfer

Some of the registers to be reloaded can be rewritten by DMA transfer. DMA transfer is performed as follows.

Table 20.42 Example of DMA Transfer Order of Registers to be Reloaded

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 040_H	TSG3nCMP1W	*1
<TSG3n_base> + 044_H	TSG3nCMP5W	*1
<TSG3n_base> + 048_H	TSG3nCMP9W	*1
<TSG3n_base> + $04C_H$	TSG3nCMP3W	*1
<TSG3n_base> + 050_H	TSG3nCMP7W	*1
<TSG3n_base> + 054_H	TSG3nCMP11W	*1
<TSG3n_base> + 058_H	TSG3nCMP0	*1
<TSG3n_base> + $05C_H$	TSG3nDCMP0W	*1
<TSG3n_base> + 060_H	TSG3nDCMP2	*1
<TSG3n_base> + 064_H	TSG3nPAT0W	*1
<TSG3n_base> + 068_H	TSG3nPAT1W	*1
<TSG3n_base> + $06C_H$	TSG3nDTC0W	*1
<TSG3n_base> + 070_H	TSG3nDTC1W	*1

Note 1. Greater addresses correspond to higher priority levels.

Table 20.43 Example of DMA Transfer Order of Registers to be Reloaded

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 140_H	TSG3nDCMP2E	*1
<TSG3n_base> + 144_H	TSG3nDCMP1E	*1
<TSG3n_base> + 148_H	TSG3nDCMP0E	*1
<TSG3n_base> + $14C_H$	TSG3nCMP0E	*1
<TSG3n_base> + 150_H	TSG3nCMP12E	*1
<TSG3n_base> + 154_H	TSG3nCMP11E	*1
<TSG3n_base> + 158_H	TSG3nCMP8E	*1
<TSG3n_base> + $15C_H$	TSG3nCMP7E	*1
<TSG3n_base> + 160_H	TSG3nCMP4E	*1
<TSG3n_base> + 164_H	TSG3nCMP3E	*1
<TSG3n_base> + 168_H	TSG3nCMP10E	*1
<TSG3n_base> + $16C_H$	TSG3nCMP9E	*1
<TSG3n_base> + 170_H	TSG3nCMP6E	*1
<TSG3n_base> + 174_H	TSG3nCMP5E	*1
<TSG3n_base> + 178_H	TSG3nCMP2E	*1
<TSG3n_base> + $17C_H$	TSG3nCMP1E	*1

Note 1. Smaller addresses correspond to higher priority levels

Table 20.44 Duty Setting in HT-PWM Mode

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 180_H	TSG3nCMPWE	*1
<TSG3n_base> + 184_H	TSG3nCMPVE	*1
<TSG3n_base> + 188_H	TSG3nCMPUE	*1

Note 1. Smaller addresses correspond to higher priority levels.

Table 20.45 Active Width Setting in SP-PWM Mode

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 190_H	TSG3nWPWE	*1
<TSG3n_base> + 194_H	TSG3nVPWE	*1
<TSG3n_base> + 198_H	TSG3nUPWE	*1

Note 1. Smaller addresses correspond to higher priority levels.

Table 20.46 Shift Width and Duty Setting in HSP-PWM Mode

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 120_H	TSG3nHSPSHWE	*1
<TSG3n_base> + 124_H	TSG3nHSPSHVE	*1
<TSG3n_base> + 128_H	TSG3nHSPSHUE	*1
<TSG3n_base> + $12C_H$	TSG3nHSPCMWE	*1
<TSG3n_base> + 130_H	TSG3nHSPCMVE	*1
<TSG3n_base> + 134_H	TSG3nHSPCMUE	*1

Note 1. Smaller addresses correspond to higher priority levels.

NOTES

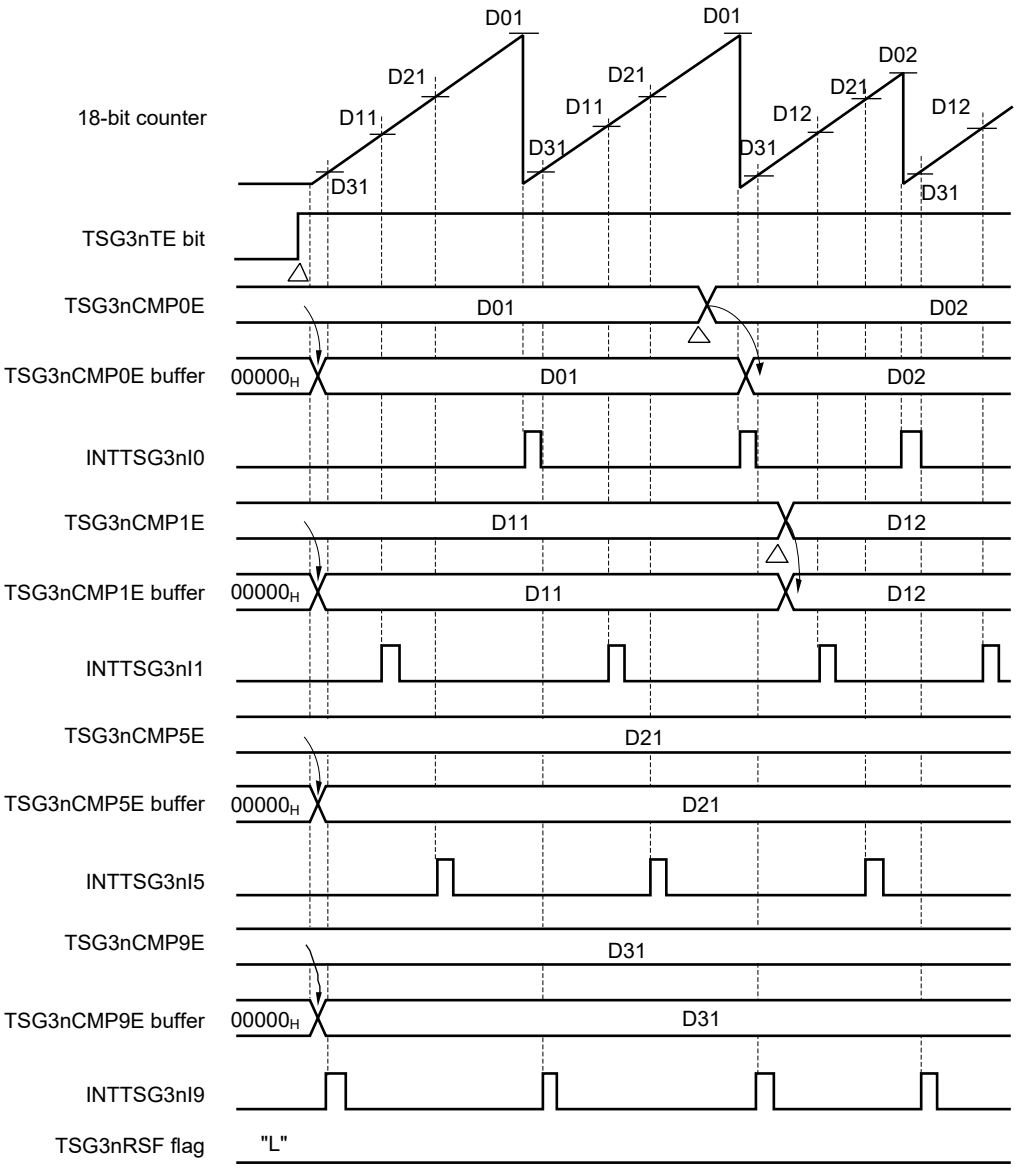
1. TSG3nCTL4 and TSG3nIOC3 should be rewritten individually.
2. Since writing to TSG3nCMP1E (including TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE) enables reloading, it should be rewritten after all the other registers to be reloaded have been rewritten (ready to be reloaded)

(1) Example of Operation in Anytime Rewrite Mode

In this mode, the values written to the compare registers (TSG3nCMP1E to TSG3nCMP12E) are transferred to the internal buffer registers immediately, and are compared with the counter value.

The values are transferred to the internal compare buffer registers one clock cycle (PCLK) after being written to the compare registers (TSG3nCMP1E to TSG3nCMP12E).

The transfer timing of the TSG3nCMP0E is the peak or trough timing (only in HT-PWM mode) of the 18-bit counter after being written to the compare registers, or at the match timing of the TSG3nCMP0E value with the 18-bit counter value (in any mode other than HT-PWM mode).



Note: D01, D02: TSG3nCMP0E setting value (00000_H-$3FFFF_H$)
D11, D12: TSG3nCMP1E setting value (00000_H-$3FFFF_H$)
D21: TSG3nCMP5E setting value (00000_H-$3FFFF_H$)
D31: TSG3nCMP9E setting value (00000_H-$3FFFF_H$)
Δ: Write Access

Figure 20.9 Anytime Rewrite Timing (Example in PWM Mode)

(a) Data reflection on PWM an Anytime Rewrite in HT-PWM

In anytime rewrite operation in HT-PWM mode, the values are transferred to the buffer at the timing to write 1 to the TSG3nIMT bit after the settings of TSG3nCMP1E, 2E, 5E, 6E, 9E, and 10E registers are modified and PWM output is forcibly set/cleared depending on the modified set value.

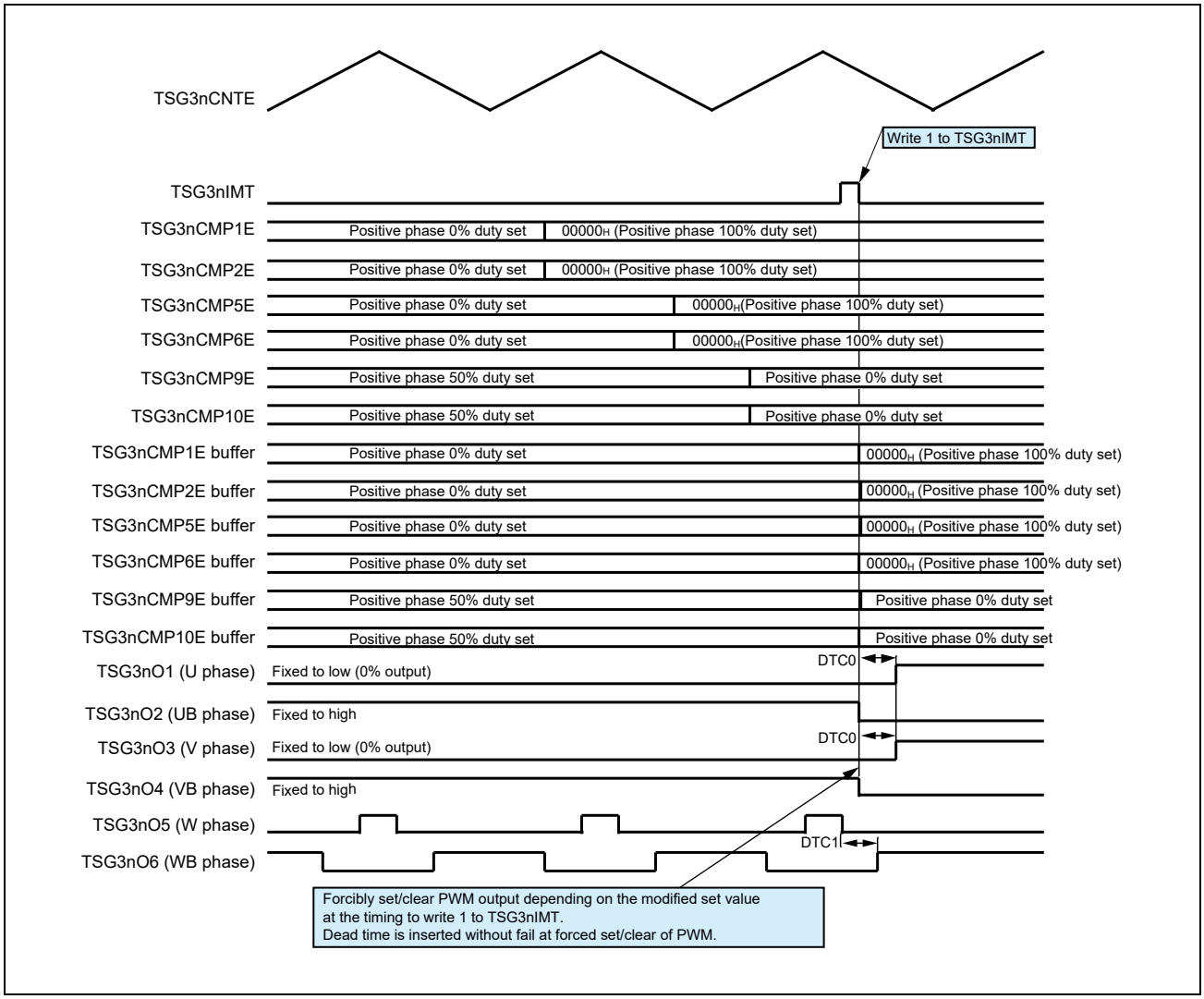


Figure 20.10 Update Timing of TSG3nCMP1E, 2E, 5E, 6E, 9E, and 10E at Anytime Rewrite Operation in HT-PWM Mode

(2) Example of Operation in Reload Mode (Simultaneous Rewrite Function)

The rewritten values of the registers to be reloaded (the registers listed in the **Section 20.3.1, List of Registers** with "Enabled" in the column of "Reload") can be transferred to the corresponding buffer registers simultaneously at the reload timing.

The registers should be rewritten when the pertinent reload request flag (TSG3nSTR0.TSG3nRSF) is 0.

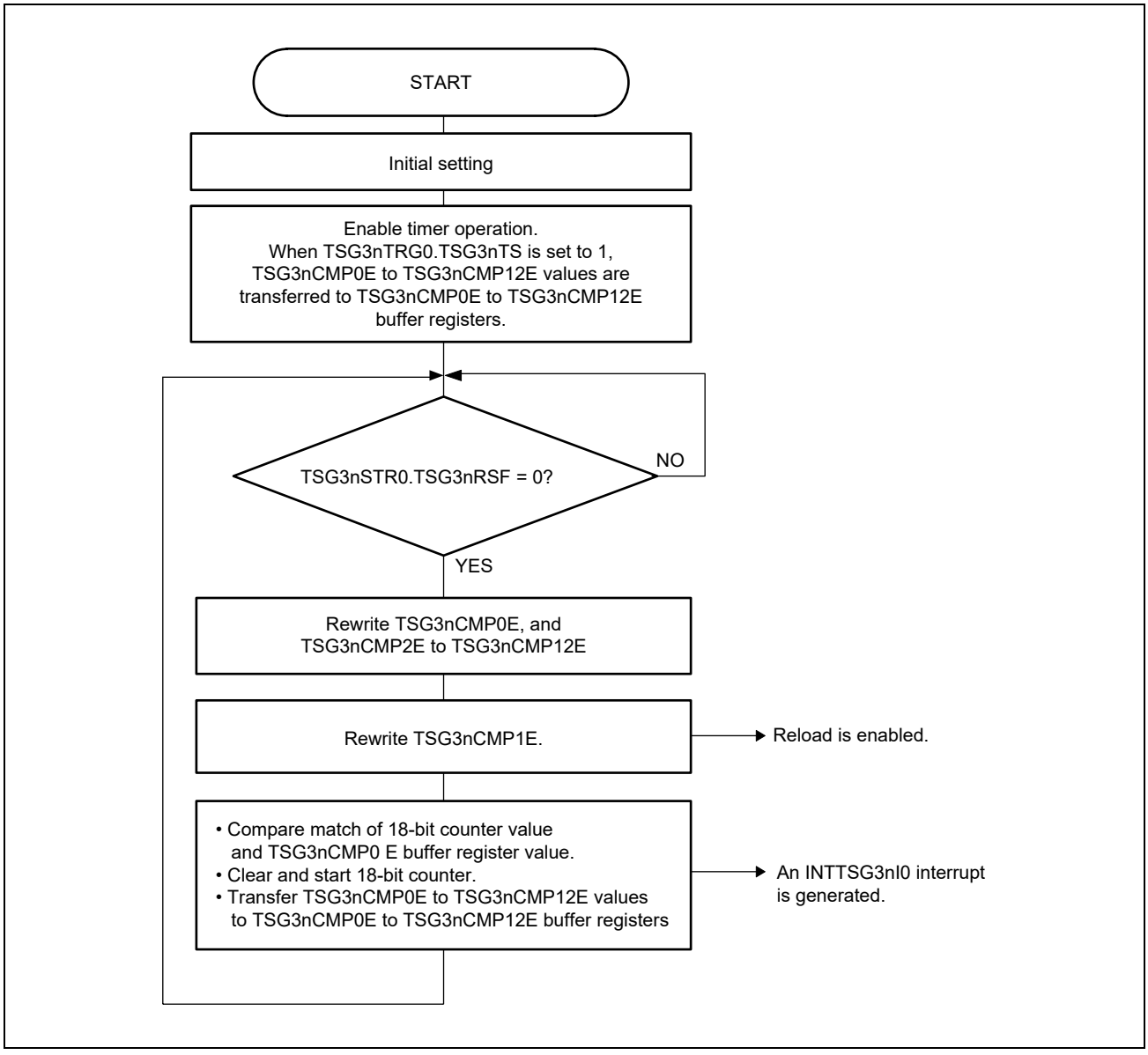


Figure 20.11 Basic Operation Flow in Reload Mode (Simultaneous Rewrite Function) (Example of PWM Mode)

CAUTION

Writing to TSG3nCMP1E also enables reloading. Therefore, TSG3nCMP1E should be rewritten after TSG3nCMP0E and TSG3nCMP2E to TSG3nCMP12E registers have been rewritten.

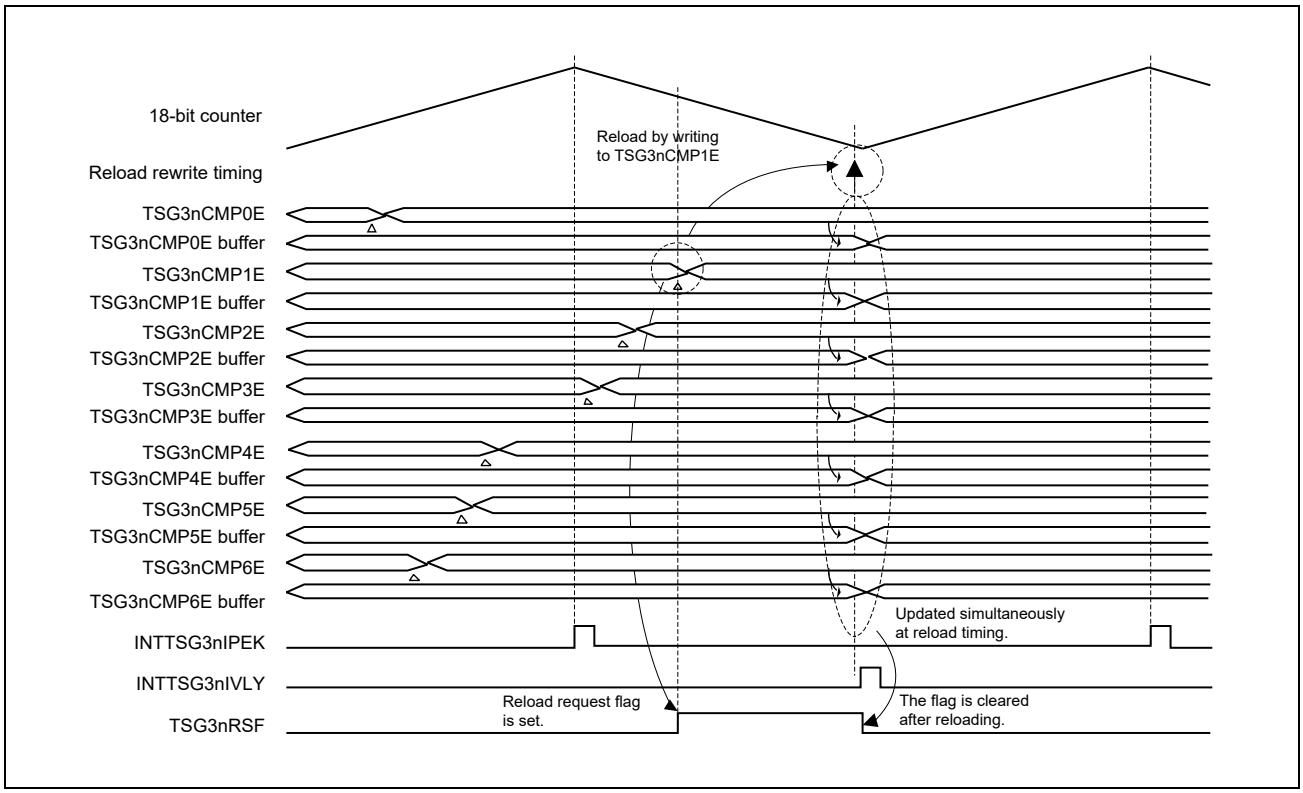


Figure 20.12 Simultaneous Rewrite Timing (Example of HT-PWM Mode) (1/2)

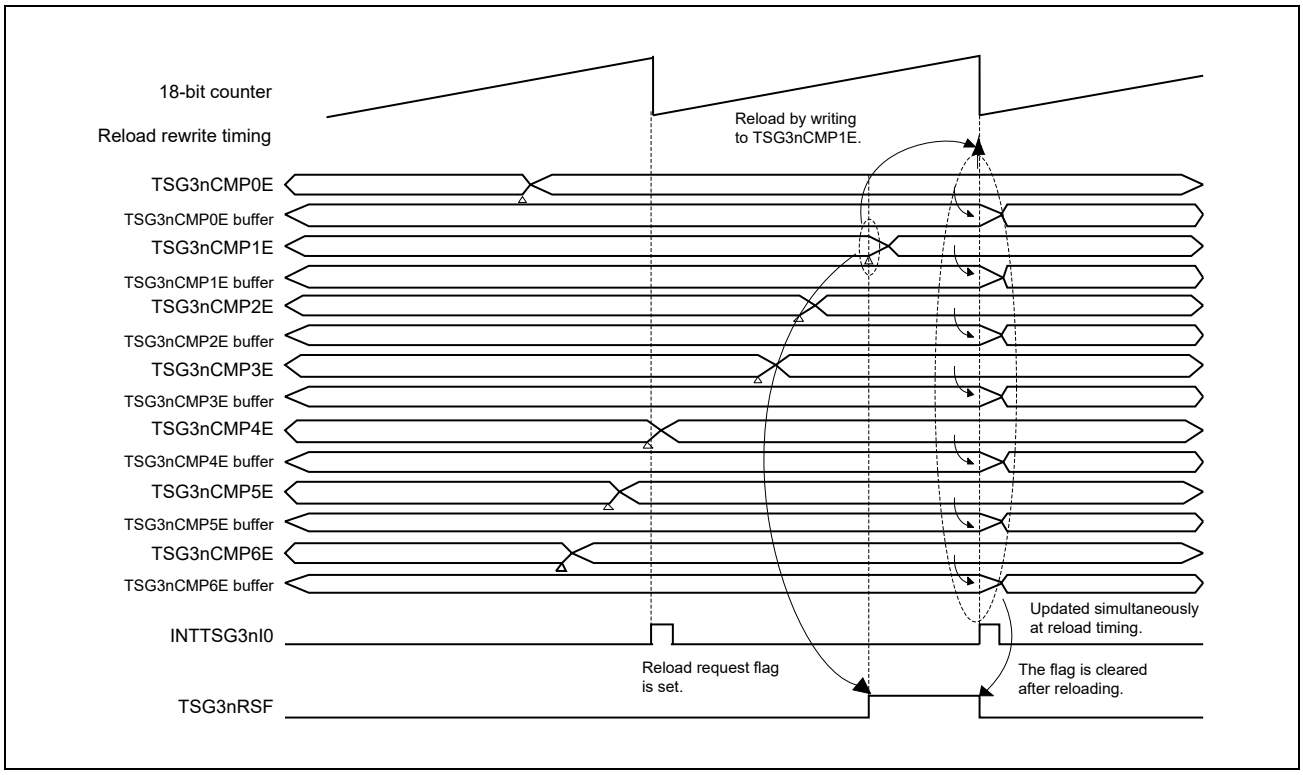


Figure 20.12 Simultaneous Rewrite Timing (Example of PWM Mode) (2/2)

(a) Reload Rewrite Setting Example in Each Mode

Reloading conditions and setting examples are shown in the following tables.

Table 20.47 List of Reload Settings (when TSG3nCTL3.TSG3nRIA = 0)

Mode	TSG3nCTL4.TSG3nPRE	TSG3nCTL4.TSG3nVRE	TSG3nCTL4.TSG3nPIE	TSG3nCTL4.TSG3nVIE	TSG3nCTL4.TSG3nRCC04-TSG3nRCC00	Reload
PWM mode SP-PWM mode 120-DC mode HSP-PWM mode	0	0/1	0/1	0/1	Any value	Setting prohibited
	1	0	0/1	0/1	Any value	When INTTSG3nI0 is generated.
	1	1	0/1	0/1	Any value	When INTTSG3nI0 is generated.
HT-PWM mode	0	0	0/1	0/1	Any value	Setting prohibited
	0	1	0/1	0/1	Any value	When INTTSG3nIVLY is generated.
	1	0	0/1	0/1	Any value	When INTTSG3nIPEK is generated
	1	1	0/1	0/1	Any value	When INTTSG3nIPEK or INTTSG3nIVLY is generated.

Table 20.48 List of Reload Settings (when TSG3nCTL3.TSG3nRIA = 1)

Mode	TSG3nCTL4.TSG3nPRE	TSG3nCTL4.TSG3nVRE	TSG3nCTL4.TSG3nPIE	TSG3nCTL4.TSG3nVIE	TSG3nCTL4.TSG3nRCC04-TSG3nRCC00	Reload
PWM mode SP-PWM mode 120-DC mode HSP-PWM mode	0	0/1	0/1	0/1	Any value	Setting prohibited
	1	0	0	0/1	Any value	Setting prohibited
	1	0	1	0/1	Any value	When INTTSG3nI0 is generated.
	1	1	0	0/1	Any value	Setting prohibited
	1	1	1	0/1	Any value	When INTTSG3nI0 is generated.
HT-PWM mode	0	0	0/1	0/1	Any value	Setting prohibited
	0	1	0	0	Any value	Setting prohibited
	0	1	0	1	Any value	When INTTSG3nIVLY is generated
	0	1	1	0	Any value	Setting prohibited
	0	1	1	1	Any value	When INTTSG3nIVLY is generated
	1	0	0	0/1	Any value	Setting prohibited
	1	0	1	0/1	Any value	When INTTSG3nIPEK is generated
	1	1	0	0	Any value	Setting prohibited
	1	1	0	1	Any value	When INTTSG3nIVLY is generated
	1	1	1	0	Any value	When INTTSG3nIPEK is generated
	1	1	1	1	Any value	When INTTSG3nIPEK or INTTSG3nIVLY is generated

20.4.1.4 List or Outputs in Each Mode

The list of timer outputs (TSG3nO0-7 pins) in each mode is shown in the following tables.

Table 20.49 List of Timer Outputs in Each mode (1/3)

Operating Mode	TSG3nO0 Pin	TSG3nO1 Pin	TSG3nO2 Pin
PWM mode	— (Fixed to low)	Outputs a PWM signal by compare match of TSG3nCMP1E and TSG3nCMP2E.	Outputs a PWM signal by compare match of TSG3nCMP3E and TSG3nCMP4E.
HT-PWM mode	Outputs the status indicating whether the 18-bit counter or 18-bit sub-counter is incremented or decremented.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP1E and TSG3nCMP2E.	Outputs an inverse phase PWM signal (with dead time) to TSG3nO1 pin.
SP-PWM mode	— (Fixed to low)	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP1E and TSG3nCMP2E.	Outputs an inverse phase PWM signal (with dead time) to TSG3nO1.
120-DC mode	— (Fixed to low)	Outputs a PWM signal by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E.	Outputs a PWM signal by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E.
HSP-PWM mode	— (Fixed to low)	Outputs a PWM signal by compare match of TSG3nCMP1E and TSG3nCMP2E.	Outputs a PWM signal by compare match of TSG3nCMP3E and TSG3nCMP4E.

Table 20.49 List of Timer Outputs in Each mode (2/3)

Operating Mode	TSG3nO3 Pin	TSG3nO4 Pin	TSG3nO5 Pin
PWM mode	Outputs a PWM signal by compare match of TSG3nCMP5E and TSG3nCMP6E.	Outputs a PWM signal by compare match of TSG3nCMP7E and TSG3nCMP8E.	Outputs a PWM signal by compare match of TSG3nCMP9E and TSG3nCMP10E.
HT-PWM mode	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP5E and TSG3nCMP6E.	Outputs an inverse phase PWM signal (with dead time) to TSG3nO3.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP9E and TSG3nCMP10E.
SP-PWM mode	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP5E and TSG3nCMP6E.	Outputs an inverse phase PWM signal (with dead time) to TSG3nO3.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP9E and TSG3nCMP10E.
120-DC mode	Outputs a PWM signal by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E.	Outputs a PWM signal by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E.	Outputs a PWM signal by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E.
HSP-PWM mode	Outputs a PWM signal by compare match of TSG3nCMP5E and TSG3nCMP6E.	Outputs a PWM signal by compare match of TSG3nCMP7E and TSG3nCMP8E.	Outputs a PWM signal by compare match of TSG3nCMP9E and TSG3nCMP10E.

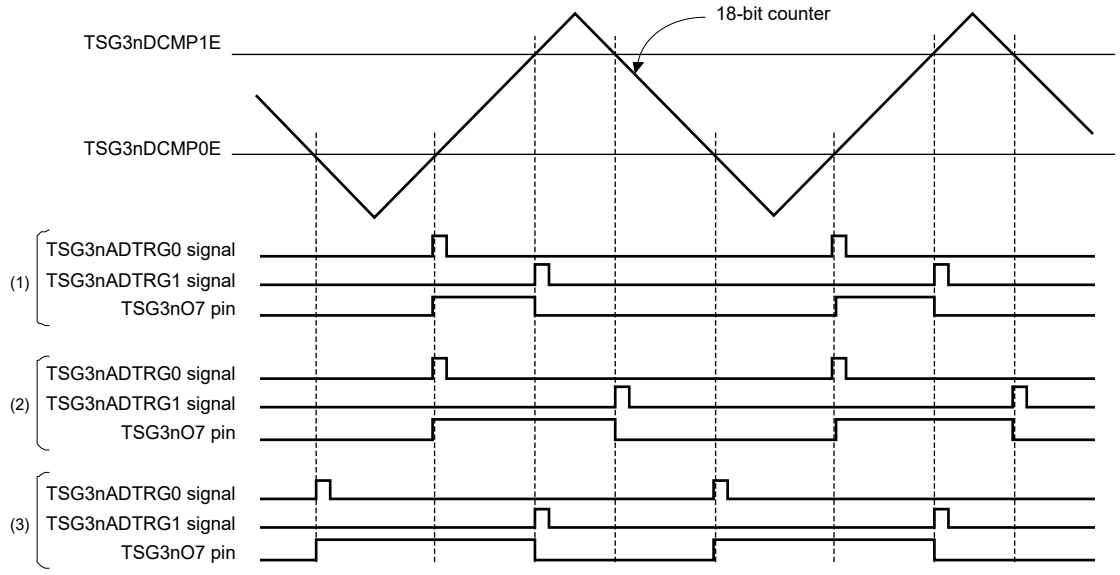
Table 20.49 List of Timer Outputs in Each mode (3/3)

Operating Mode	TSG3nO6 Pin	TSG3nO7 Pin
PWM mode	Outputs a PWM signal by compare match of TSG3nCMP11E and TSG3nCMP12E.	Outputs a diagnostic signal or A/D conversion trigger.*1
HT-PWM mode	Outputs an inverse phase PWM signal (with dead time) to TSG3nO5.	Outputs a diagnostic signal or A/D conversion trigger.*1
SP-PWM mode	Outputs an inverse phase PWM signal (with dead time) to TSG3nO5.	Outputs a diagnostic signal or A/D conversion trigger.*1
120-DC mode	Outputs a PWM signal by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E.	Outputs a diagnostic signal or A/D conversion trigger.*1
HSP-PWM mode	Outputs a PWM signal by compare match of TSG3nCMP11E and TSG3nCMP12E.	Outputs a diagnostic signal or A/D conversion trigger.*1

Note 1. For TSG3nO7, see **Section 20.4.1.4(a), TSG3nO7 Pin Output Control**.

(a) TSG3nO7 Pin Output Control

The TSG3nO7 pin can output a pulse of A/D conversion trigger (TSG3nIOC1.TSG3nTGS = 0) or diagnostic output (TSG3nIOC1.TSG3nTGS = 1). When outputting a pulse of A/D conversion trigger, the TSG3nO7 pin is activated at the rising edge of the TSG3nADTRG0 signal, and inactivated at the rising edge of the TSG3nADTRG1 signal. When the TSG3nADTRG0 signal is detected while the TSG3nO7 pin is active, the TSG3nO7 pin remains active. When the TSG3nADTRG1 signal is detected while the TSG3nO7 pin is inactive, the TSG3nO7 pin remains inactive. If TSG3nADTRG0 and TSG3nADTRG1 signal triggers occur simultaneously, the TSG3nO7 pin is inactivated.



Note: (1) When TSG3nDCMP0E register<TSG3nDCMP1E register, TSG3nCTL5 register = 0004_H, and TSG3nCTL6 register = 0010_H
(2) When TSG3nDCMP0E register<TSG3nDCMP1E register, TSG3nCTL5 register = 0004_H, and TSG3nCTL6 register = 0020_H
(3) When TSG3nDCMP0E register<TSG3nDCMP1E register, TSG3nCTL5 register = 0008_H, and TSG3nCTL6 register = 0010_H

Figure 20.13 Example of A/D Trigger Output Timing of TSG3nO7 Pin (TSG3nIOC1.TSG3nTGS = 0)

During diagnostic output, the active level is output on the TSG3nO7 pin with the width specified by the TSG3nCTL0.TSG3nDWD bit when the values of the TSG3nDCMP0E to TSG3nDCMP2E bits match that of the 18-bit counter. If a TSG3nDCMP0E to TSG3nDCMP2E value again matches the value of the 18-bit counter value while the diagnostic output on the TSG3nO7 pin is already at the active level, pulse output on the TSG3nO7 pin continues for the number of cycles of PCLK (8 or 16) set by the TSG3nDWD bit from the point of the later match.

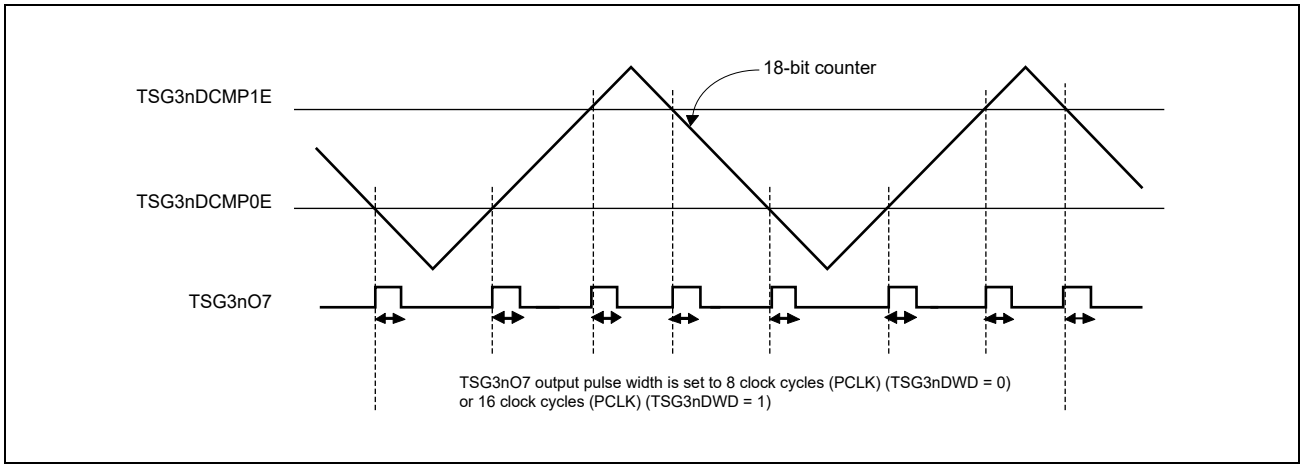


Figure 20.14 Example of TSG3nO7 Pin Diagnostic Pulse Output Timing (1) (TSG3nIOC1.TSG3nTGS = 1)

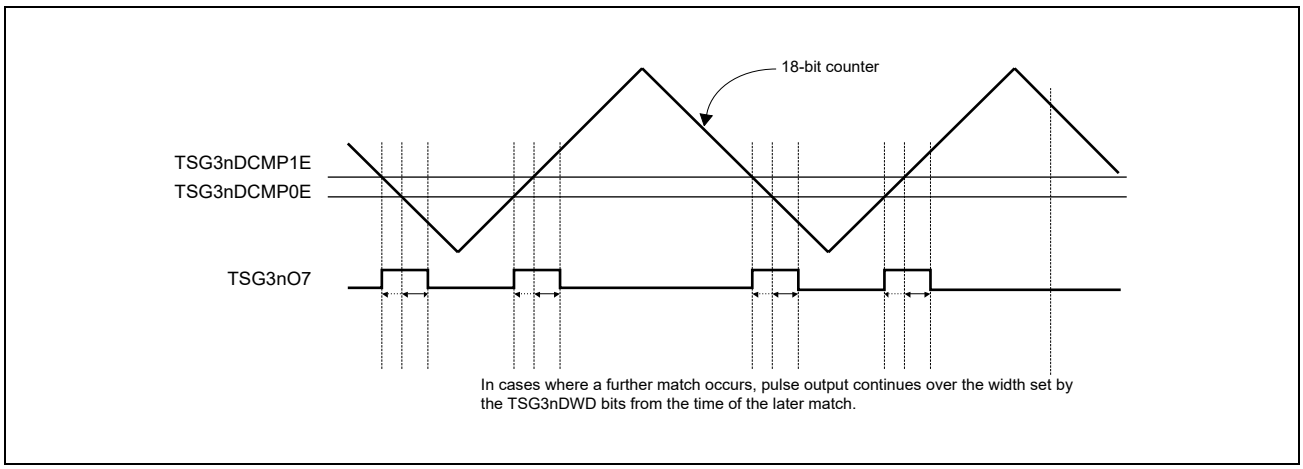


Figure 20.15 Example of TSG3nO7 Pin Diagnostic Pulse Output Timing (2) (with Pulse Output Width Overlapped)

20.4.2 Match Interrupt

The TSG3n can generate interrupts such as a compare match interrupt (INTTSG3nIm), a peak interrupt (INTTSG3nIPEK), and a trough interrupt (INTTSG3nIVLY). For an error interrupt and warning interrupt (INTTSG3nIER and INTTSG3nIWN), see Section **20.4.6, Error/Warning Interrupt**.

A period interrupt (INTTSG3nI0) is generated for each timer period. In HT-PWM mode, it is generated when the TSG3nDTC0 buffer register value matches with the 18-bit counter value. When the 18-bit counter performs sawtooth waveform operation (PWM mode, SP-PWM mode, 120-DC mode, and HSP-PWM mode), it is generated after the 18-bit counter value has matched with the TSG3nCMP0E buffer register value.

A compare-match interrupt (INTTSG3nIm) is generated by a match of the TSG3nCMPmE buffer register value with the 18-bit counter value depending on the compare register to be used in each operating mode (m = 1 to 12).

A peak interrupt (INTTSG3nIPEK) is generated in all the modes. In HT-PWM mode, it is generated when the 18-bit counter switches from incrementing to decrementing. When the 18-bit counter performs sawtooth waveform operation (PWM mode, SP-PWM mode, 120-DC mode, and HSP-PWM mode), it is generated after the 18-bit counter value has matched with the TSG3nCMP0E buffer register value (the same timing as an INTTSG3nI0 interrupt).

A trough interrupt (INTTSG3nIVLY) is generated when the 18-bit counter switches from decrementing to incrementing in HT-PWM mode.

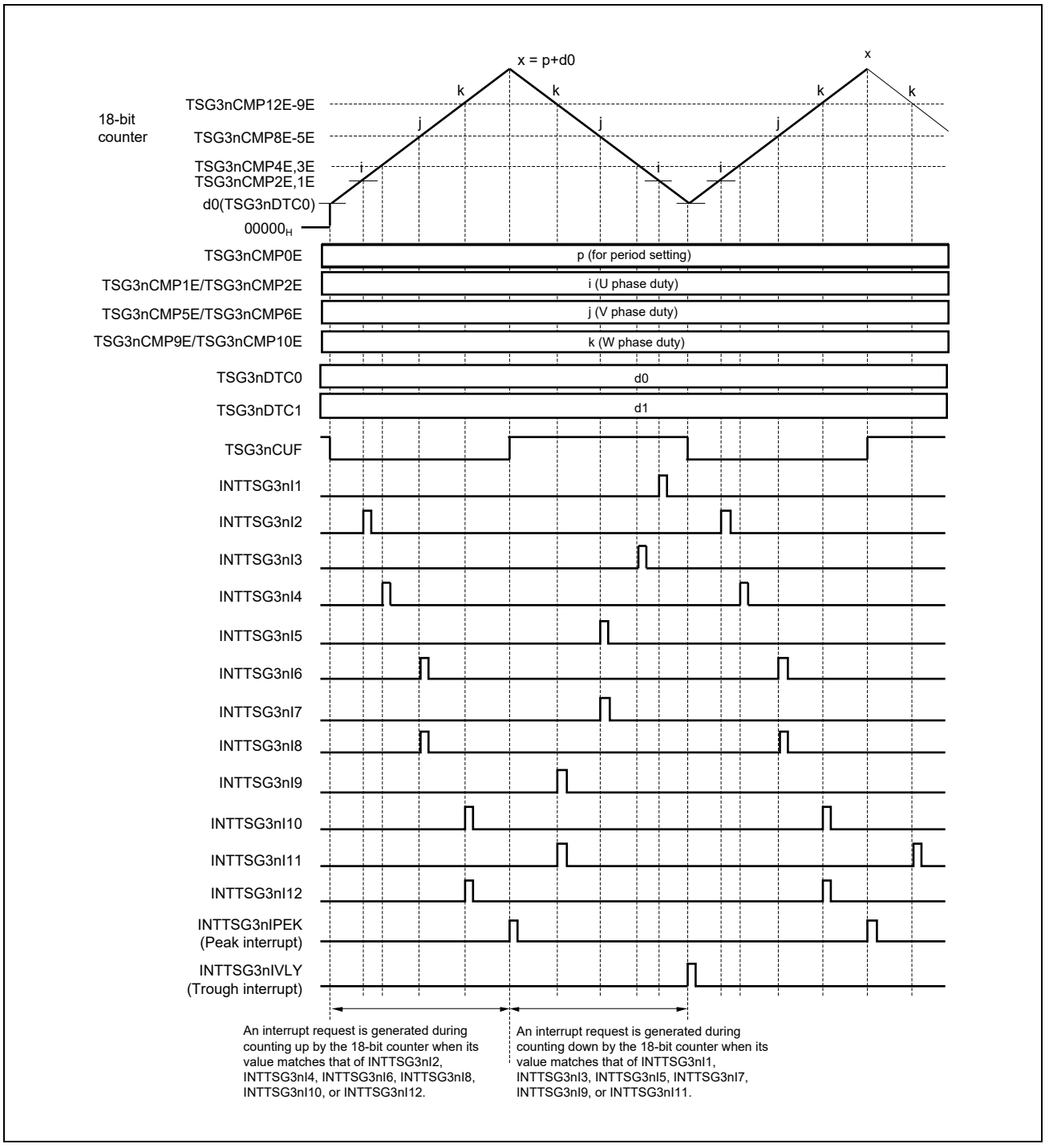


Figure 20.16 Interrupt Generation Example (Example of HT-PWM Mode)

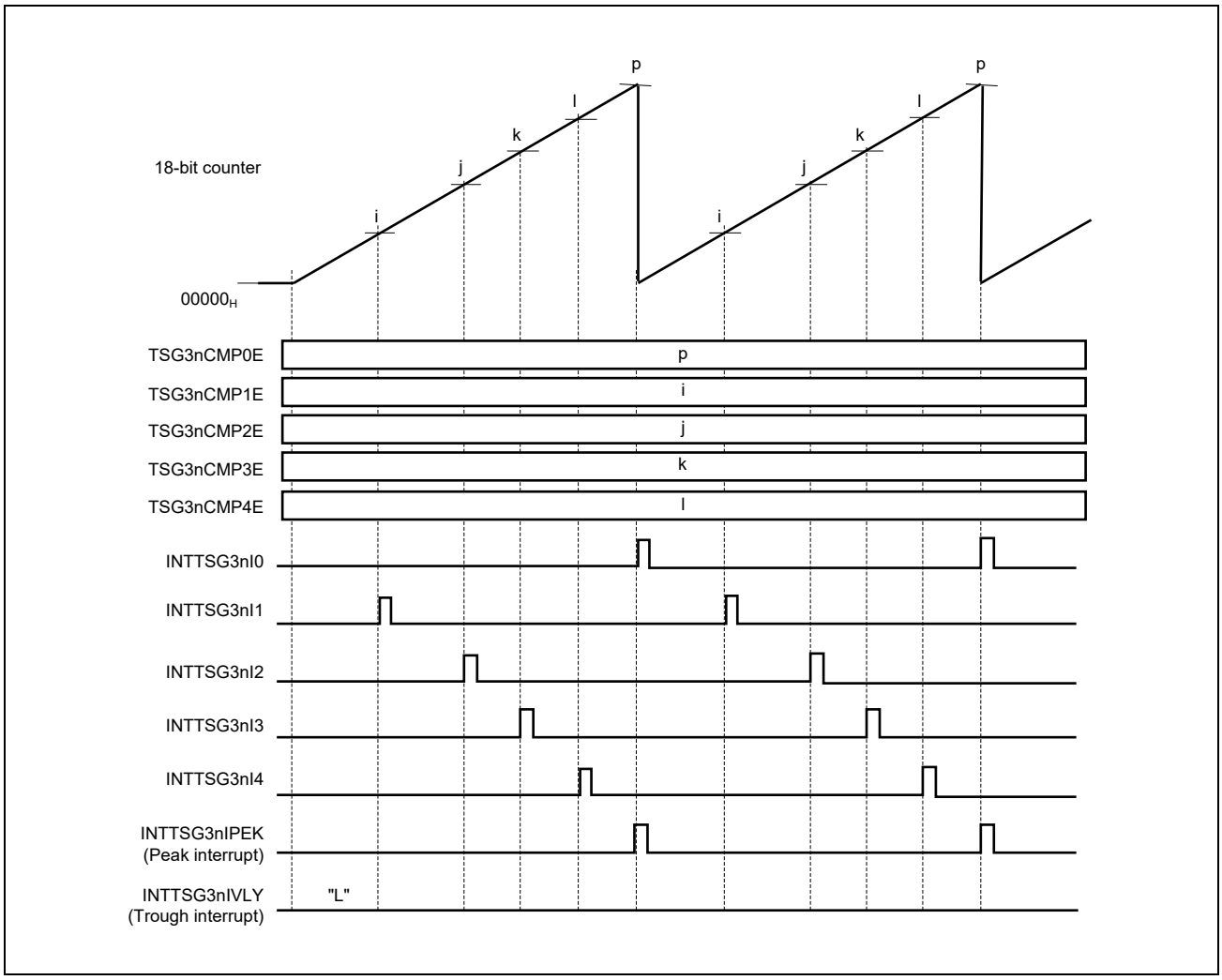


Figure 20.17 Interrupt Generation Example (Example of PWM Mode)

Interrupt in each mode (INTTSG3nI0 to INTTSG3nI12, INTTSG3nIPEK, INTTSG3nIVLY, INTTSG3nIER, and INTTSG3nIWN) are listed in **Table 20.50**.

Table 20.50 List of Interrupts in Each Mode (1/5)

Operating Mode	INTTSG3nI0	INTTSG3nI1	INTTSG3nI2	INTTSG3nI3
PWM mode	TSG3nCMP0E compare match interrupt	TSG3nCMP1E compare match interrupt*1	TSG3nCMP2E compare match interrupt*1	TSG3nCMP3E compare match interrupt*1
HT-PWM mode	Period interrupt	TSG3nCMP1E compare match interrupt*2 when decrementing (TSG3nCUF = 1)	TSG3nCMP2E compare match interrupt*2 when incrementing (TSG3nCUF = 0)	TSG3nCMP3E compare match interrupt*2 when decrementing (TSG3nCUF = 1)
SP-PWM mode	TSG3nCMP0E compare match interrupt	TSG3nCMP1E compare match interrupt*1	TSG3nCMP2E compare match interrupt*1	—
120-DC mode	TSG3nCMP0E compare match interrupt	TSG3nCMP1E compare match interrupt*1	TSG3nCMP2E compare match interrupt*1	TSG3nCMP3E compare match interrupt*1
HSP-PWM mode	TSG3nCMP0E compare match interrupt	TSG3nCMP1E compare match interrupt*1	TSG3nCMP2E compare match interrupt*1	TSG3nCMP3E compare match interrupt*1

Note 1. A compare match interrupt is not generated when TSG3nCMP0E < TSG3nCMPmE (m = 1 to 12).

Note 2. A compare match interrupt is not generated when $00000_H \leq$ TSG3nCMPmE < TSG3nDTC0, (TSG3nCMP0E + TSG3nDTC0) < TSG3nCMPmE

Table 20.50 List of Interrupts in Each Mode (2/5)

Operating Mode	INTTSG3nI4	INTTSG3nI5	INTTSG3nI6	INTTSG3nI7
PWM mode	TSG3nCMP4E compare match interrupt*1	TSG3nCMP5E compare match interrupt*1	TSG3nCMP6E compare match interrupt*1	TSG3nCMP7E compare match interrupt*1
HT-PWM mode	TSG3nCMP4E compare match interrupt*2 when incrementing (TSG3nCUF = 0)	TSG3nCMP5E compare match interrupt*2 when decrementing (TSG3nCUF = 1)	TSG3nCMP6E compare match interrupt*2 when incrementing (TSG3nCUF = 0)	TSG3nCMP7E compare match interrupt*2 when decrementing (TSG3nCUF = 1)
SP-PWM mode	—	TSG3nCMP5E compare match interrupt*1	TSG3nCMP6E compare match interrupt*1	—
120-DC mode	TSG3nCMP4E compare match interrupt*1	TSG3nCMP5E compare match interrupt*1	TSG3nCMP6E compare match interrupt*1	TSG3nCMP7E compare match interrupt*1
HSP-PWM mode	TSG3nCMP4E compare match interrupt*1	TSG3nCMP5E compare match interrupt*1	TSG3nCMP6E compare match interrupt*1	TSG3nCMP7E compare match interrupt*1

Note 1. A compare match interrupt is not generated when TSG3nCMP0E < TSG3nCMPmE (m = 1 to 12).

Note 2. A compare match interrupt is not generated when $00000_H \leq$ TSG3nCMPmE < TSG3nDTC0, (TSG3nCMP0E + TSG3nDTC0) < TSG3nCMPmE.

Table 20.50 List of Interrupts in Each Mode (3/5)

Operating Mode	INTTSG3nI8	INTTSG3nI9	INTTSG3nI10	INTTSG3nI11
PWM mode	TSG3nCMP8E compare match interrupt*1	TSG3nCMP9E compare match interrupt*1	TSG3nCMP10E compare match interrupt*1	TSG3nCMP11E compare match interrupt*1
HT-PWM mode	TSG3nCMP8E compare match interrupt*2 when incrementing (TSG3nCUF = 0)	TSG3nCMP9E compare match interrupt*2 when decrementing (TSG3nCUF = 1)	TSG3nCMP10E compare match interrupt*2 when incrementing (TSG3nCUF = 0)	TSG3nCMP11E compare match interrupt*2 when decrementing (TSG3nCUF = 1)
SP-PWM mode	—	TSG3nCMP9E compare match interrupt*1	TSG3nCMP10E compare match interrupt*1	—
120-DC mode	TSG3nCMP8E compare match interrupt*1	TSG3nCMP9E compare match interrupt*1	TSG3nCMP10E compare match interrupt*1	TSG3nCMP11E compare match interrupt*1
HSP-PWM mode	TSG3nCMP8E compare match interrupt*1	TSG3nCMP9E compare match interrupt*1	TSG3nCMP10E compare match interrupt*1	TSG3nCMP11E compare match interrupt*1

Note 1. A compare match interrupt is not generated when TSG3nCMP0E < TSG3nCMPmE (m = 1 to 12).

Note 2. A compare match interrupt is not generated when $00000_H \leq$ TSG3nCMPmE < TSG3nDTC0, (TSG3nCMP0E + TSG3nDTC0) < TSG3nCMPmE.

Table 20.50 List of Interrupts in Each Mode (4/5)

Operating Mode	INTTSG3nI12	INTTSG3nIPEK	INTTSG3nIVLY
PWM mode	TSG3nCMP12E compare match interrupt*1	Peak interrupt at the same timing with INTTSG3nI0	—
HT-PWM mode	TSG3nCMP12E compare match interrupt*2 when incrementing (TSG3nCUF=0)	Peak interrupt	Trough interrupt
SP-PWM mode	—	Peak interrupt at the same timing with INTTSG3nI0	—
120-DC mode	TSG3nCMP12E compare match interrupt*1	Peak interrupt at the same timing with INTTSG3nI0	—
HSP-PWM mode	TSG3nCMP12E compare match interrupt*1	Peak interrupt at the same timing with INTTSG3nI0	—

Note 1. A compare match interrupt is not generated when TSG3nCMP0E < TSG3nCMPmE (m = 1 to 12).

Note 2. A compare match interrupt is not generated when $00000_H \leq$ TSG3nCMPmE < TSG3nDTC0, (TSG3nCMP0E + TSG3nDTC0) < TSG3nCMPmE.

Table 20.50 List of Interrupts in Each Mode (5/5)

Operating Mode	INTTSG3nIER	INTTSG3nIWN
PWM mode	Error interrupt	Warning interrupt
HT-PWM mode	Error interrupt	Warning interrupt
SP-PWM mode	Error interrupt	Warning interrupt
120-DC mode	Error interrupt	Warning interrupt
HSP-PWM mode	Error interrupt	Warning interrupt

20.4.3 Flags

Table 20.51 List of Flags

Number	Flag Name	Symbol	Registers	Operating Mode
(1)	Up count flag	TSG3nCUF	TSG3nSTR0	HT-PWM mode
		TSG3nSUF	TSG3nSTR0	
(2)	Positive phase and inverse phase simultaneous active state detection flag	TSG3nTBF0-TSG3nTBF2	TSG3nSTR2	All operating modes
(3)	Reload request flag	TSG3nRSF	TSG3nSTR0	All operating modes
(4)	Noise Detection Flag	TSG3nNDF	TSG3nSTR2	All operating modes
(5)	Pattern order detection flag	TSG3nTSF	TSG3nSTR1	All operating modes
(6)	Pattern error detection flag	TSG3nPEF	TSG3nSTR2	All operating modes
(7)	Pattern reversal detection flag	TSG3nPRF	TSG3nSTR2	All operating modes
(8)	TSG3nPTSI2 to TSG3nPTSI0 pin abnormal toggle detection flag	TSG3nPTF	TSG3nSTR2	All operating modes
(9)	TSG3nSTCI0 and TSG3nSTCI1 signal simultaneous trigger detection flag	TSG3nTDF	TSG3nSTR2	All operating modes
(10)	Pattern phase difference detection flag	TSG3nPPF	TSG3nSTR2	All operating modes
(11)	Timer output pattern flag	TSG3nOPF0-TSG3nOPF2	TSG3nSTR1	All operating modes
(12)	Pattern switch detection signal (internal signal)	TSG3nPTE	—	All operating modes

20.4.3.1 Up Count Flag (TSG3nCUF, TSG3nSUF)

Name

Up count flag (TSG3nSTR0.TSG3nCUF and TSG3nSUF)

Description

There are following two up count flags.

TSG3nCUF is an up/down count flag of the 18-bit counter.

TSG3nSUF is an up/down count flag of the 18-bit sub-counter.

For both TSG3nCUF and TSG3nSUF, 0 means increment, and 1 means decrement. TSG3nCUF and TSG3nSUF can be used only in HT-PWM mode.

Example of operation

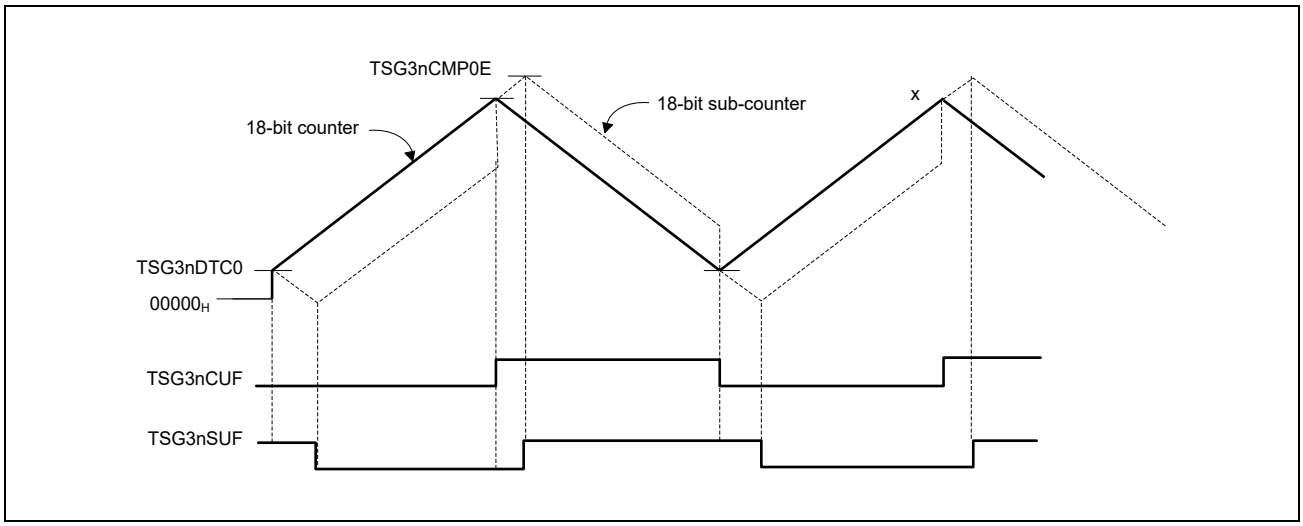


Figure 20.18 Example of Up Count Flag Operation

NOTES

1. TSG3nCUF value is:
   0 (up count) when TSG3nDTC0 ≤ 18-bit counter ≤ (TSG3nCMP0E + TSG3nDTC0-2)
   1 (down count) when (TSG3nCMP0E + TSG3nDTC0) ≥ 18-bit counter ≥ TSG3nDTC0 + 2
2. TSG3nSUF value is:
   0 (up count) when 0 ≤ 18-bit sub-counter ≤ (TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1-2)
   1 (down count) when (TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1) ≥ 18-bit sub-counter ≥ 2

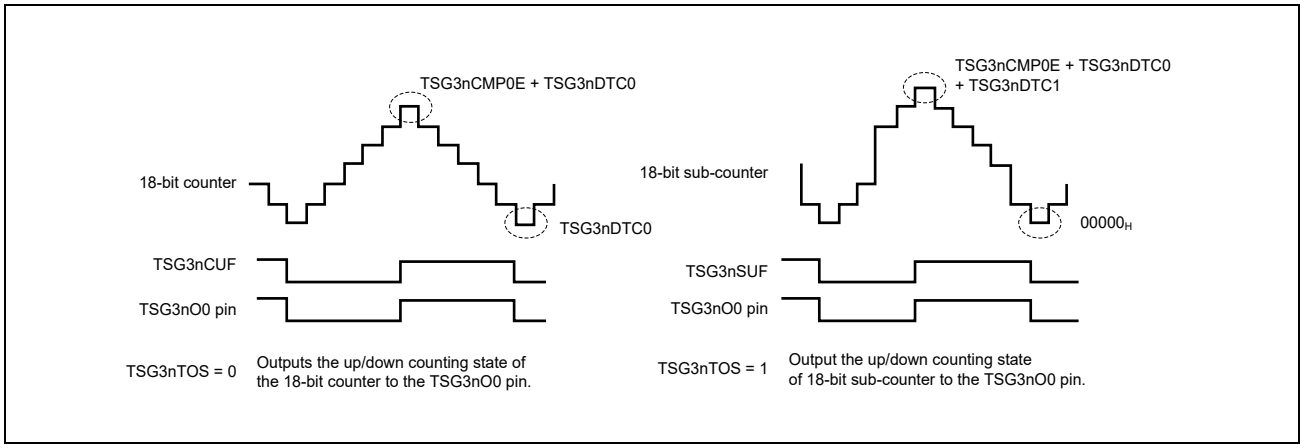


Figure 20.19 TSG3nO0 Pin Output depending on TSG3nIOC1.TSG3nTOS Setting

Operating Mode

TSG3nCUF and TSG3nSUF can be used only in HT-PWM mode.

20.4.3.2 Positive Phase and Inverse Phase Simultaneous Active State Detection Flag (TSG3nTBF0 to TSG3nTBF2)

Name

Positive phase and inverse phase simultaneous active state detection flag
(TSG3nSTR2.TSG3nTBF0 to TSG3nTBF2 flags)

Description

When any of TSG3nCTL1.TSG3nTBA2 to TSG3nTBA0 is 1, TSG3nTBF0 to TSG3nTBF2 can detect the simultaneous active state of the positive phase and inverse phase of TSG3n.

When the simultaneous active state of the positive phase and inverse phase of the TSG3n is detected, the corresponding TSG3nTBF0 to TSG3nTBF2 flags are set to 1, and an error interrupt (INTTSG3nIER) is generated. The flags are cleared when 1 is written to TSG3nSTC.TSG3nTBR0 to TSG3nTBR2, respectively.

Example of operation

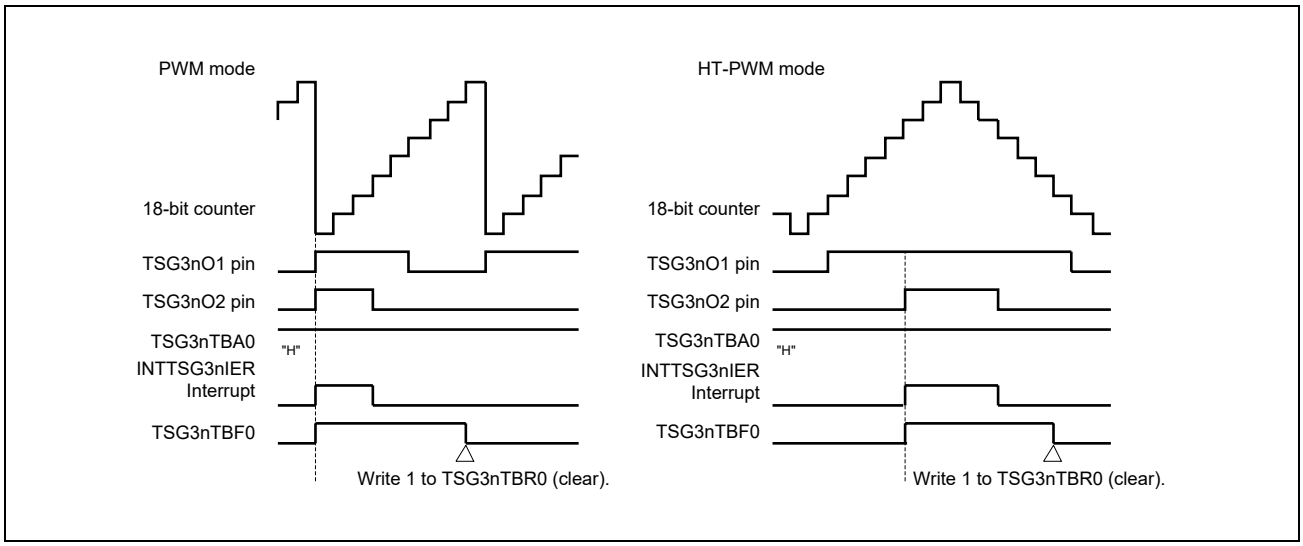


Figure 20.20 Example of Positive Phase and Inverse Phase Simultaneous Active State Detection Flag Operation

Operating Mode

Available in all operating modes.

CAUTION

TSG3nTBF0 to TSG3nTBF2 are valid only when TSG3nCTL1.TSG3nTBA0 to TSG3nTBA2 = 1 and TSG3nSTR0.TSG3nTE = 1.

20.4.3.3 Reload Request Flag (TSG3nRSF)

Name

Reload request flag (TSG3nSTR0.TSG3nRSF)

Description

TSG3nRSF is set to 1 when a reload request is generated (when a value is written to TSG3nCMP1E (TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE)), and cleared to 0 when the value is transferred to all the buffer registers.

Example of operation

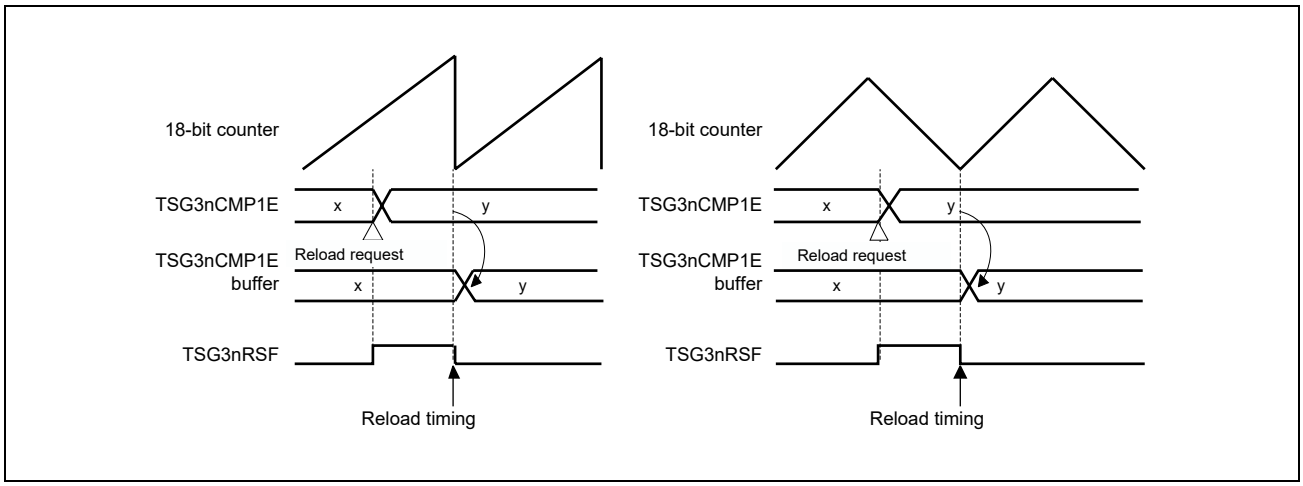


Figure 20.21 Example of Reload Request Flag Operation

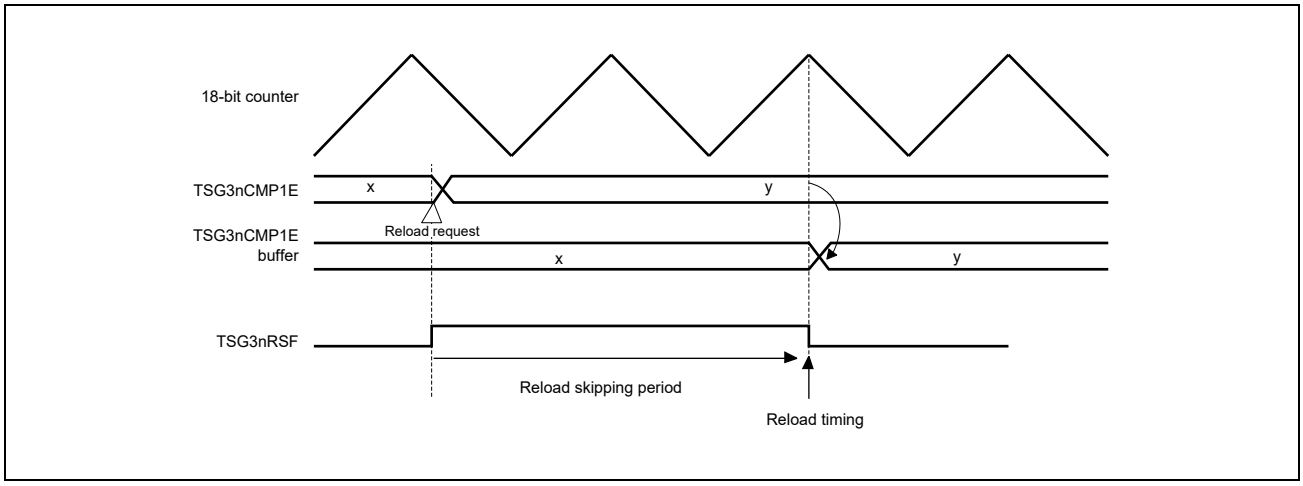


Figure 20.22 Reload Request Flag and Reload Skipping Period

Operating Mode

Available in all operating modes.

20.4.3.4 Noise Detection Flag (TSG3nNDF)

Name

Noise detection flag (TSG3nSTR2.TSG3nNDF)

Description

TSG3nNDF can detect that two or more pins of TSG3nPTSI2 to TSG3nPTSI0 have changed simultaneously (a noise is generated).

TSG3nNDF is set to 1 when two or more pins of TSG3nPTSI2 to TSG3nPTSI0 have changed simultaneously (a noise is generated), and a warning interrupt (INTTSG3nIWN) is generated. The TSG3nNDF flag is cleared to 0 when 1 is written to the TSG3nSTC.TSG3nNDR bit.

Example of operation

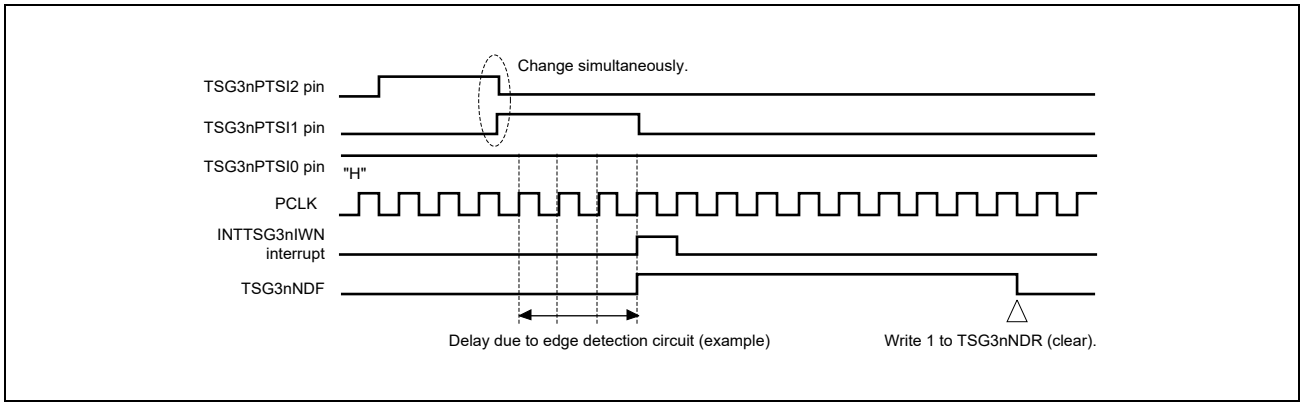


Figure 20.23 Example of Noise Detection Flag Operation

Operation mode

Available in all operating modes.

CAUTION

TSG3nNDF is valid only when TSG3nCTL1.TSG3nNDC = 1 and TSG3nSTR0.TSG3nTE = 1.

20.4.3.5 Pattern Order Detection Flag (TSG3nTSF)

Name

Pattern order detection flag (TSG3nSTR1.TSG3nTSF)

Description

TSG3nTSF can detect the order of patterns input to the TSG3nPTSI2 to TSG3nPTSI0 pins.

TSG3nTSF is set depending on the values input to the TSG3nPTSI2 to TSG3nPTSI0 pins as shown in the table below

Table 20.52 Pattern Order Detection Flag and Pattern Input Order

TSG3nTSF	Values Input to TSG3nPTSI2 to TSG3nPTSI0 Pins
0	[1,0,1] → [1,0,0] → [1,1,0] → [0,1,0] → [0,1,1] → [0,0,1]
1	[1,0,1] ← [1,0,0] ← [1,1,0] ← [0,1,0] ← [0,1,1] ← [0,0,1]

Example of operation

(a) When Normal Input to TSG3nPTSI2 to TSG3nPTSI0 Pins is Detected

As shown in **Figure 20.24**, if the TSG3nPTSI2 to TSG3nPTSI0 pins change in the normal order, 0 or 1 is set according to the change order at the change timing.

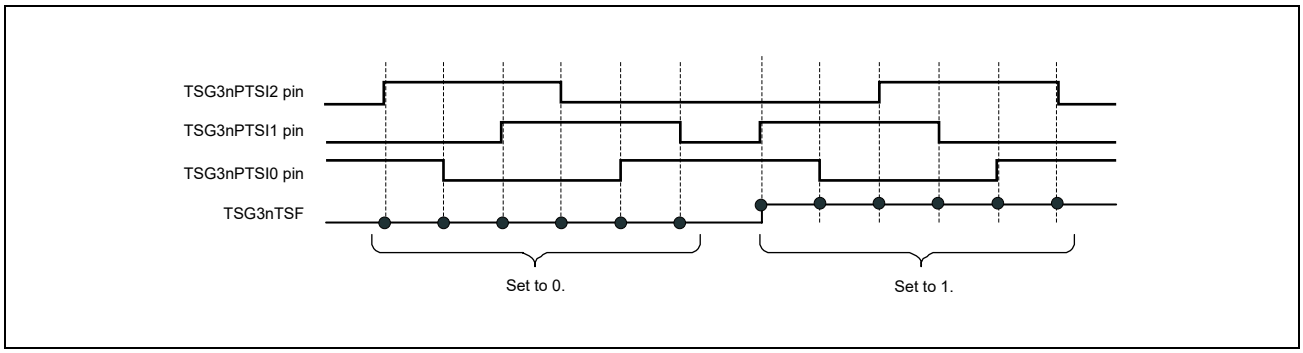


Figure 20.24 Example of Pattern Order Detection Flag Operation (Normal Operation)

(b) Detection of Input Pattern Order

Immediately after TSG3n starts operation, the rotation direction cannot be determined. Therefore, TSG3nTSF cannot detect the change (normal or reverse rotation) in the patterns input to the TSG3nPTSI2 to TSG3nPTSI0 pins. To enable detection of change immediately after the beginning of operation, TSG3nPSC should be set before operation starts (when TSG3nTE = 0, the TSG3nPSC value is reflected).

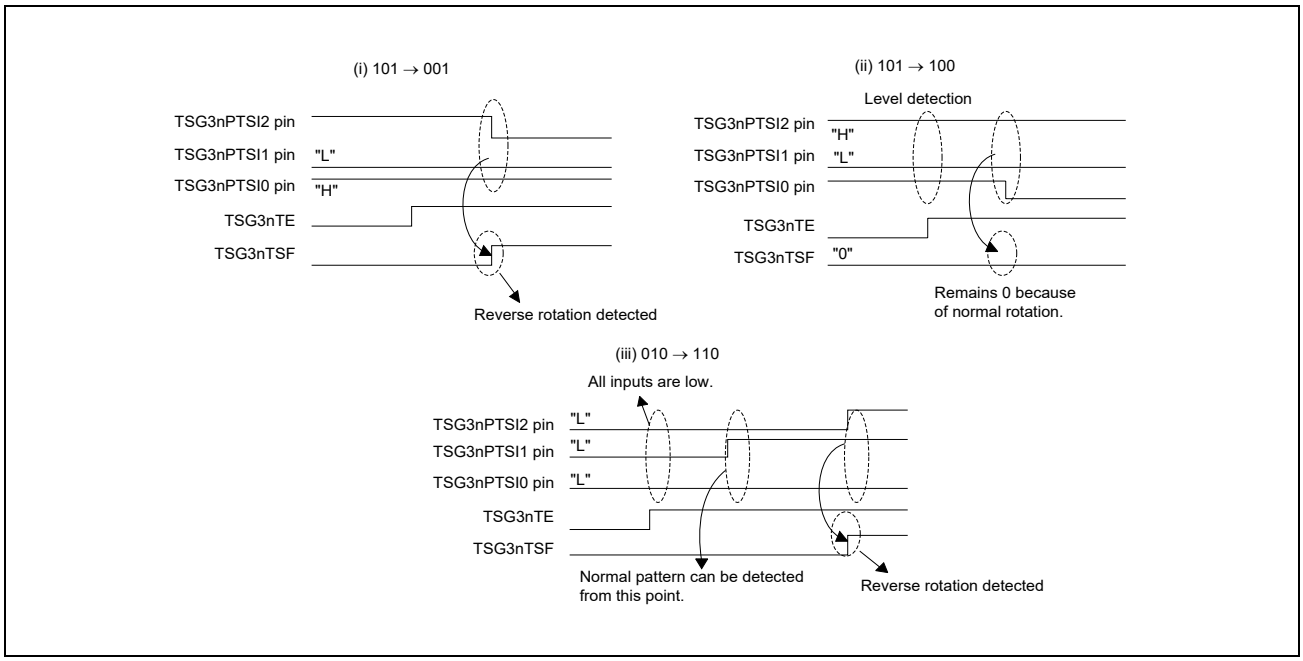


Figure 20.25 Example of Detecting Change (Normal/Reverse Rotation) in Pattern Input to TSG3nPTSI2 to TSG3nPTSI0 Pins

(c) When Abnormal Input to TSG3nPTSI2 to TSG3nPTSI0 Pins is Detected

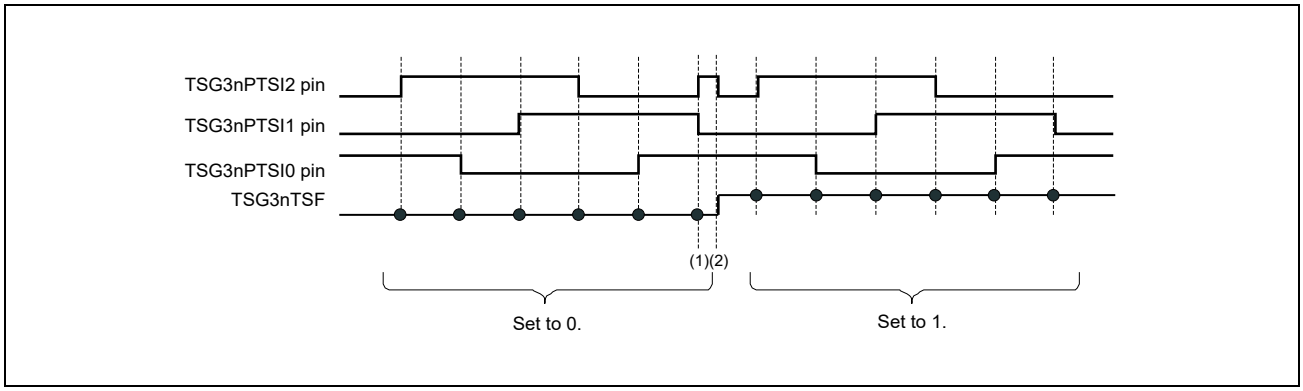


Figure 20.26 Example of Operation when Values Input to Two Pins of TSG3nPTSI2 to TSG3nPTSI0 Change (Abnormal Operation)

(1) TSG3nTSF does not change at this point because it expects the input pattern change to {0, 1, 0} or {0, 0, 1} (if values of two pins change, TSG3nTSF does not change).
(2) TSG3nPTSI2 to TSG3nPTSI0 pins are determined to have been changed from {1, 0, 1} to {0, 0, 1}, and TSG3nTSF is set to 1.

Operation mode

Available in all operating modes.

20.4.3.6 Pattern Error Detection Flag (TSG3nPEF)

Name

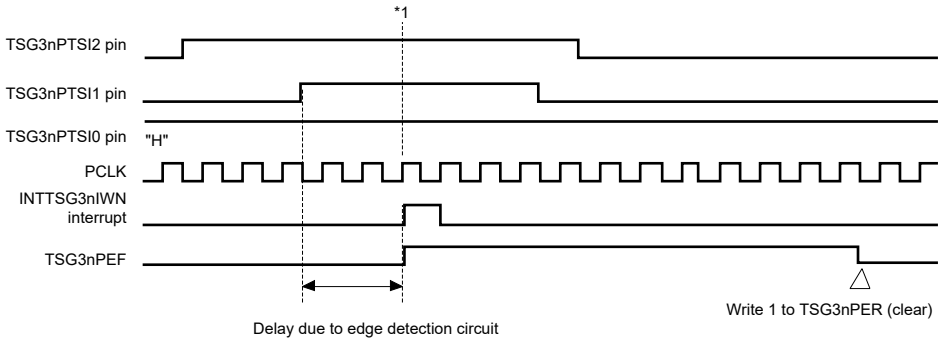
Pattern error detection flag (TSG3nSTR2.TSG3nPEF)

Description

TSG3nPEF can detect that 000 or 111 is input to the TSG3nPTSI2 to TSG3nPTSI0 pins.

TSG3nPEF is set to 1 when the levels of the TSG3nPTSI2 to TSG3nPTSI0 pins are 111 or 000, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPEF is cleared to 0 when 1 is written to TSG3nSTC.TSG3nPER.

Example of operation



Note 1. 111 is detected

Figure 20.27 Example of Pattern Error Detection Flag Operation (TSG3nPTSI2 to TSG3nPTSI0 Pins = 111)

Operation mode

Available in all operating modes.

CAUTION

TSG3nPEF is valid only when TSG3nCTL1.TSG3nPEC = 1 and TSG3nSTR0.TSG3nTE = 1.

20.4.3.7 Pattern Reversal Detection Flag (TSG3nPRF)

Name

Pattern reversal detection flag (TSG3nSTR2.TSG3nPRF)

Description

TSG3nPRF can detect that the pattern change order of the TSG3nPTSI2 to TSG3nPTSI0 pins have been reversed.

TSG3nPRF is set to 1 when the pattern order detection flag (TSG3nTSF) changes, and a warning interrupt (INTTSG3nIWN) is generated. However, immediately after TSG3nSTR0.TSG3nTE is set to 1, TSG3nPRF is valid at the timing of the second and subsequent change in TSG3nPTSI2 to TSG3nPTSI0 pins.

TSG3nPRF is cleared to 0 when 1 is written to the TSG3nSTC.TSG3nPRR bit.

Example of operation

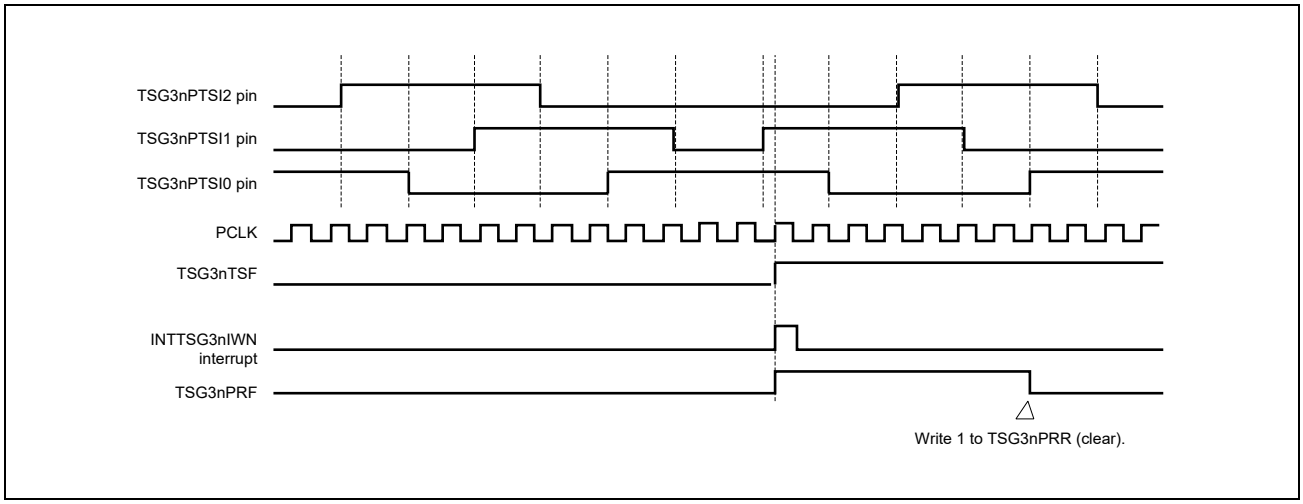
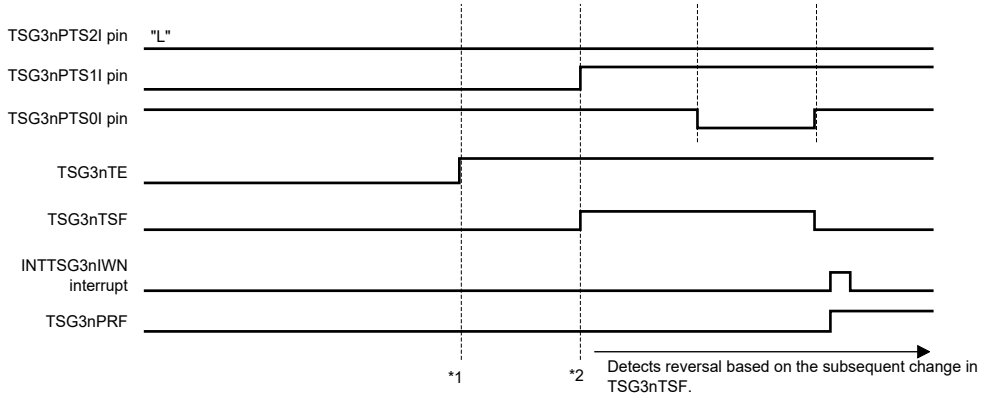


Figure 20.28 Example of Pattern Reversal Detection Flag Operation



Note 1. Operation starts.

Note 2. If TSG3nTSF is set to 1 by the first change in the TSG3nPTSI2 to TSG3nPTSI0 pins immediately after TSG3nTE is set to 1, reversal is not detected.

Figure 20.29 Example of Operation immediately after TSG3nTE Flag in TSG3nSTR0 is Set to 1

Operation mode

Available in all operating modes.

CAUTION

TSG3nPRF is valid only when TSG3nCTL1.TSG3nPRC = 1 and TSG3nSTR0.TSG3nTE = 1.

20.4.3.8 TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag (TSG3nPTF)

Name

TSG3nPTSI2 to TSG3nPTSI0 pin abnormal toggle detection flag (TSG3nSTR2.TSG3nPTF)

Description

TSG3nPTF can detect that the values of the TSG3nPTSI2 to TSG3nPTSI0 pins change three or more times during the TSG3nOPCI0 or TSG3nOPCI1 signal trigger.

TSG3nPTF is set to 1 when the third trigger of TSG3nOPCI0 or TSG3nOPCI1 signal occurs simultaneously with the change in TSG3nPTSI2 to TSG3nPTSI0, and a warning interrupt (INTTSG3nIWN) is generated.

TSG3nPTF is cleared to 0 when 1 is written to TSG3nSTC.TSG3nPTR.

Example of operation

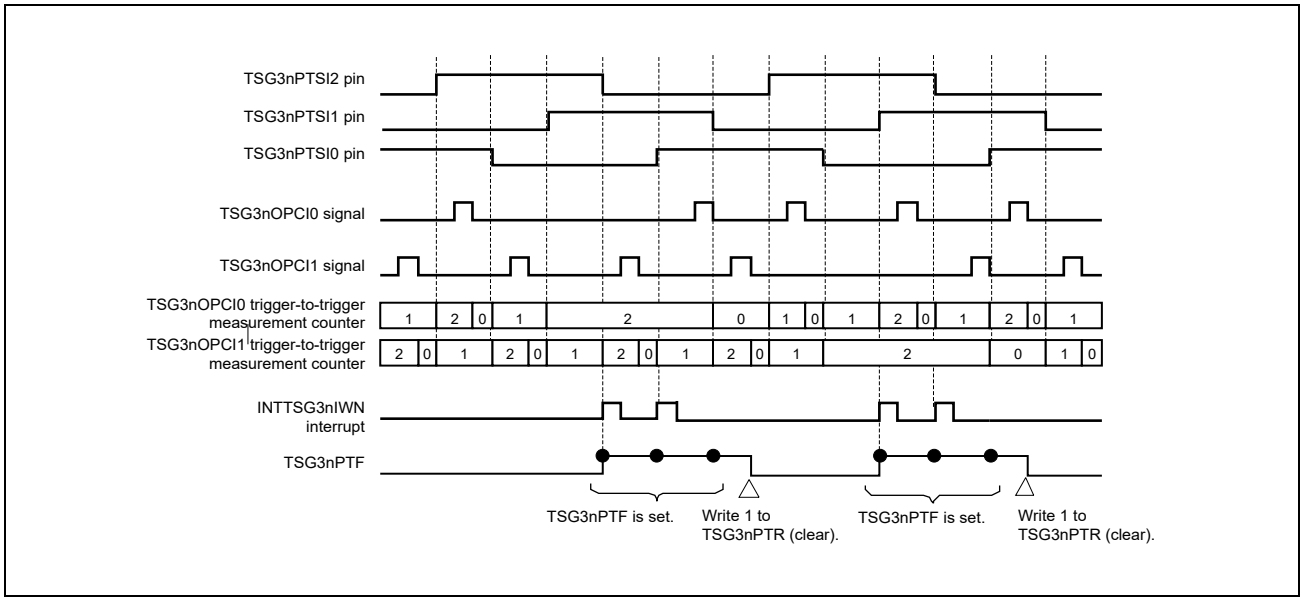


Figure 20.30 Operation of TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag Operation

Operating mode

Available in all operating modes.

NOTES

1. TSG3nPTF is valid only when TSG3nCTL1.TSG3nPTC1 bit = 1 and TSG3nSTR0.TSG3nTE = 1.
2. When TSG3nPTC0 bit = 1 and TSG3nPTC1 bit = 1, TSG3nO1 to TSG3nO6 pin output switch control is automatically switched to pattern switch method (TSG3nOPT0.TSG3nPOT bit = 0) if an abnormal toggle is detected.

20.4.3.9 TSG3nOPCI0, TSG3nOPCI1 Signal Simultaneous Trigger Detection Flag (TSG3nTDF)

Name

TSG3nOPCI0 and TSG3nOPCI1 signal simultaneous trigger detection flag (TSG3nSTR2.TSG3nTDF)

Description

TSG3nTDF can detect that TSG3nOPCI0 and TSG3nOPCI1 signals are generated simultaneously.

TSG3nTDF is set to 1 when the TSG3nOPCI0 and TSG3nOPCI1 signals are generated simultaneously, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nTDF is cleared to 0 when 1 is written to TSG3nSTC.TSG3nTDR.

Example of operation

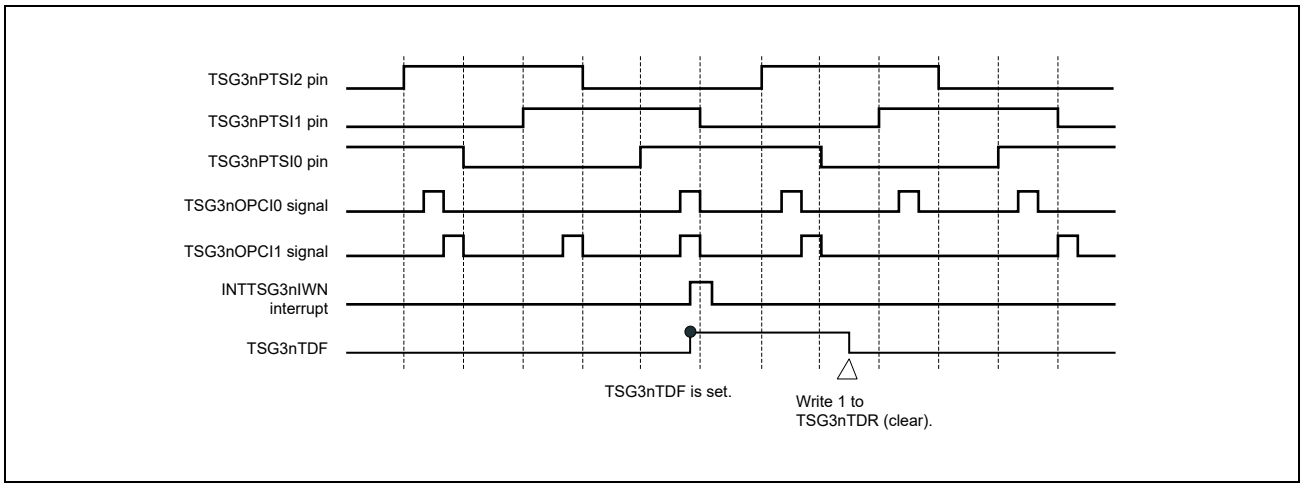


Figure 20.31 Operation of TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag Operation

Operating mode

Available in all operating modes.

CAUTION

TSG3nTDF is valid only when TSG3nCTL1.TSG3nTDC = 1 and TSG3nSTR0.TSG3nTE = 1.

20.4.3.10 Pattern Phase Difference Detection Flag (TSG3nPPF)

Name

Pattern phase difference detection flag (TSG3nSTR2.TSG3nPPF)

Description

TSG3nPPF can detect the phase difference between the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) and the output pattern (TSG3nSTR1.TSG3nOPF2 to TSG3nOPF0 flags).

TSG3nPPF is set to 1 when the pattern phase difference is detected when the TSG3nOPCI0 and TSG3nOPCI1 signal triggers are input, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPPF remains 1 until it is cleared to 0 when 1 is written to TSG3nSTC.TSG3nPPR by software. When the phase difference is detected, TSG3nPPF is set at each operation clock cycle (PCLK). TSG3nPPF should be cleared to 0 when no phase difference occurs.

Table 20.53 Correspondence between Normal Input Patterns and Output Patterns

TSG3nPTSI2 to TSG3nPTSI0 pins (Input)	“1, 0, 1”	“1, 0, 0”	“1, 1, 0”	“0, 1, 0”	“0, 1, 1”	“0, 0, 1”
TSG3nOPF2 to TSG3nOPF0 flags (Output)	“0, 0, 1” “1, 0, 1” “1, 0, 0”	“1, 0, 1” “1, 0, 0” “1, 1, 0”	“1, 0, 0” “1, 1, 0” “0, 1, 0”	“1, 1, 0” “0, 1, 0” “0, 1, 1”	“0, 1, 0” “0, 1, 1” “0, 0, 1”	“0, 1, 1” “0, 0, 1” “1, 0, 1”

Example of operation

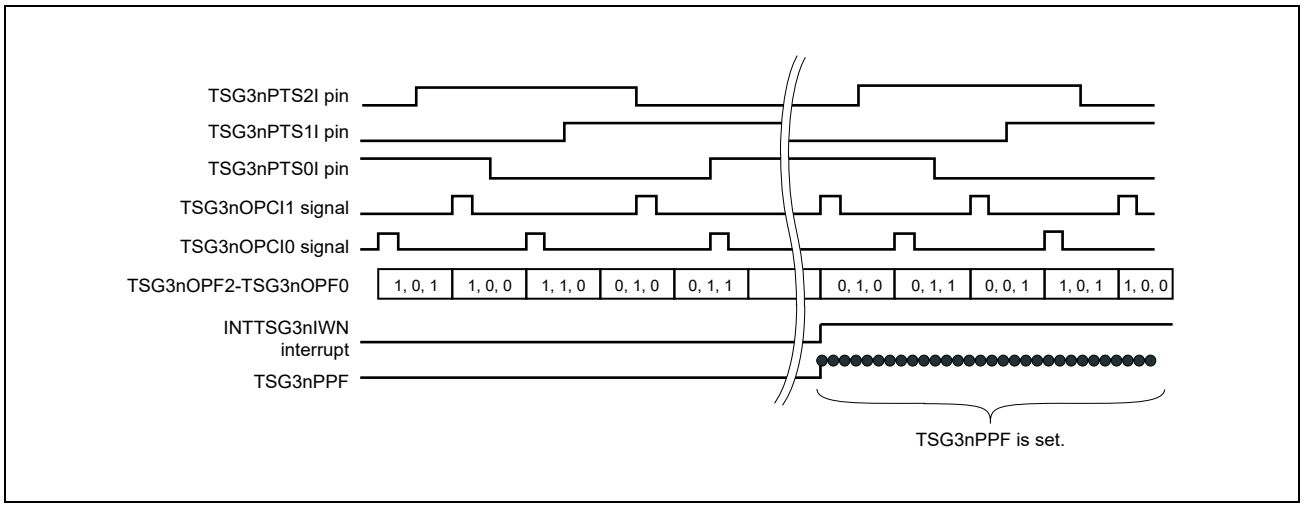


Figure 20.32 Example of Pattern Difference Detection Flag Operation

Operating mode

Available in all operating modes.

CAUTIONS

1. TSG3nPPF is valid only when TSG3nCTL1.TSG3nPPC = 1 and TSG3nSTR0.TSG3nTE = 1.
2. When 000 or 111 is input to the TSG3nPTSI2 to TSG3nPTSI0 pins, or when TSG3nOPF2 to TSG3nOPF0 are set to 000 or 111, TSG3nPPF is not set.

20.4.3.11 Timer Output Pattern Flag (TSG3nOPF2-TSG3nOPF0)

Name

Timer output pattern flag (TSG3nSTR1.TSG3nOPF2 to TSG3nOPF0)

Description

TSG3nOPF2 to TSG3nOPF0 flags indicate the timer output patterns.

For details, see **Section 20.4.7.6, 120-DC Mode**, and **Section 20.4.7.10, Software Output Control Function**.

Operating mode

Available in all operating modes.

20.4.3.12 Pattern Switch Detection Signal (TSG3nPTE)

Name

Pattern switch detection signal (TSG3nPTE signal)

Description

The TSG3nPTE signal toggles when the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) changes. The toggle pattern is determined by the TSG3nPSC bit (TSG3nOPT0.TSG3nPSS = 1).

Table 20.54 Change Timing of Pattern Switch Detection Signal (1/2)

- TSG3nPSC = 0

		TSG3nPTSI2-TSG3nPTSI0 Pins after Change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	—
	100	—	—	—	—	Toggle	—	—	—
	110	—	—	—	—	—	Toggle	—	—
	010	—	—	—	—	—	—	Toggle	—
	011	—	—	—	—	—	—	—	Toggle
	001	—	—	Toggle	—	—	—	—	—

Table 20.54 Change Timing of Pattern Switch Detection Signal (2/2)

- TSG3nPSC = 1

		TSG3nPTSI2-TSG3nPTSI0 Pins after Change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	—	—	—	—	Toggle
	100	—	—	Toggle	—	—	—	—	—
	110	—	—	—	Toggle	—	—	—	—
	010	—	—	—	—	Toggle	—	—	—
	011	—	—	—	—	—	Toggle	—	—
	001	—	—	—	—	—	—	Toggle	—

Example of operation

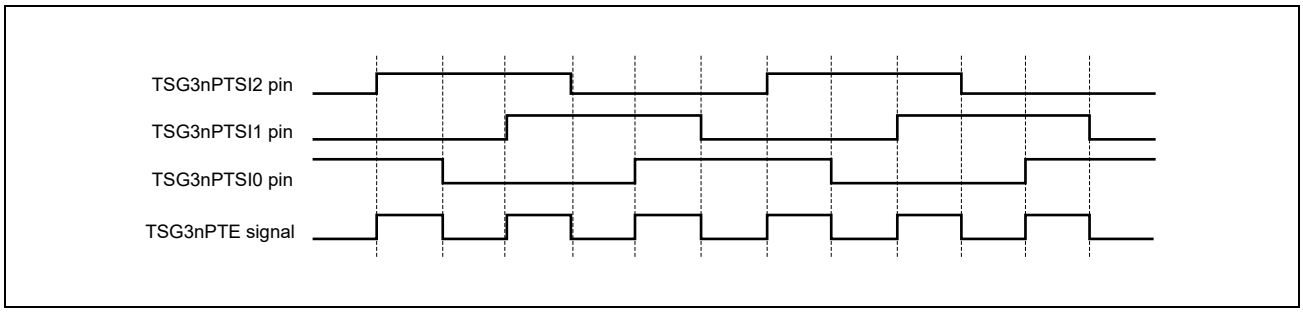


Figure 20.33 Example of Pattern Switch Detection Signal Operation

Operating mode

Available in all operating modes.

CAUTION

The TSG3nPTE signal is valid only when TSG3nIOC1.TSG3nPTS = 1 and TSG3nSTR0.TSG3nTE = 1.

20.4.4 Interrupt Skipping Function

Operation related to the interrupt skipping function is described below.

- Peak interrupts (INTTSG3nIPEK) and trough interrupts (INTTSG3nIVLY) can be skipped.
- TSG3nCTL4.TSG3nPIE enables outputting of the INTTSG3nIPEK interrupt and specifies whether to skip the interrupts.
- TSG3nCTL4.TSG3nVIE enables outputting of the INTTSG3nIVLY interrupt and specifies whether to skip the interrupts.

When TSG3nCTL3.TSG3nRIA is set to 1 (with reload skipping), reload is executed at the same timing as the interrupt after being skipped.

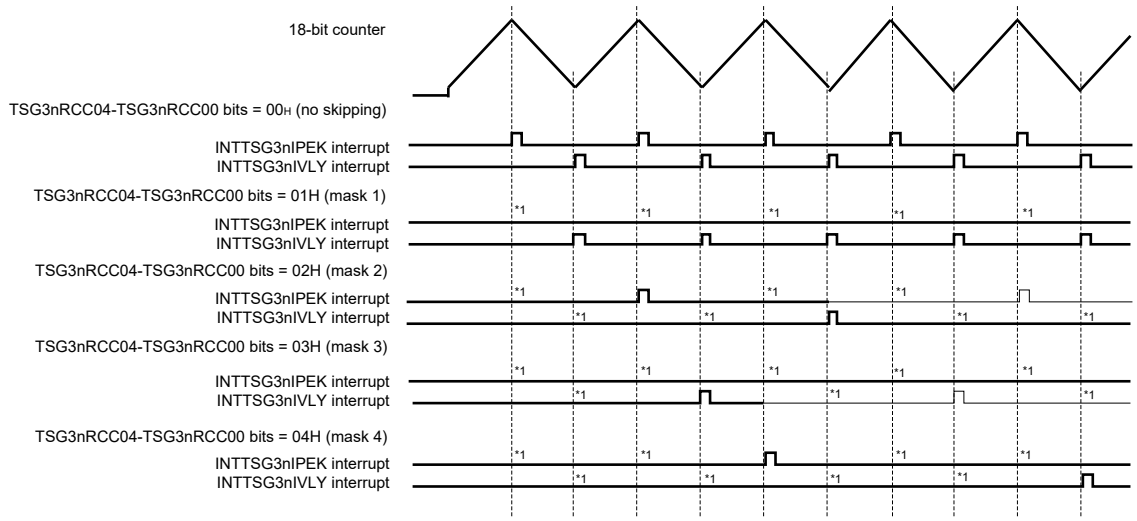
When TSG3nCTL3.TSG3nRIA is set to 0 (without reload skipping), reload is executed at the specified reload timing regardless of interrupt skipping.

CAUTION

When a value is written to TSG3nCTL4, and TSG3nRCC04 to TSG3nRCC00 are transferred to the buffer register, the interrupt skipping counter is cleared. Therefore, when the interrupt skipping function is used, interrupt interval may be long temporarily. To avoid this, the interrupt skipping count should be changed with the reload timing being set to the interrupts skipped (TSG3nCTL3.TSG3nRIA = 1).

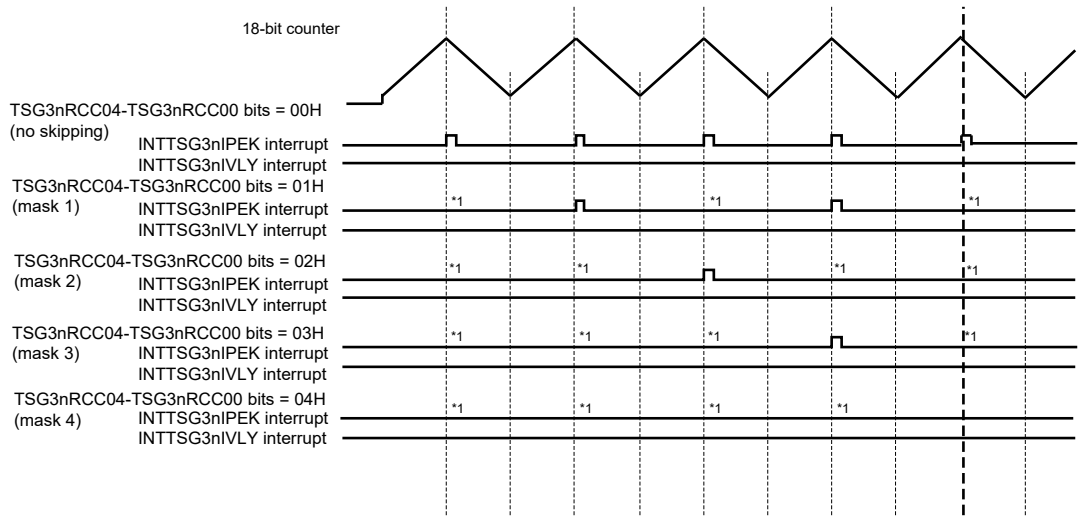
20.4.4.1 Operation of Interrupt Skipping Function

Timing diagram of interrupt skipping function in various conditions are shown in the following figures.



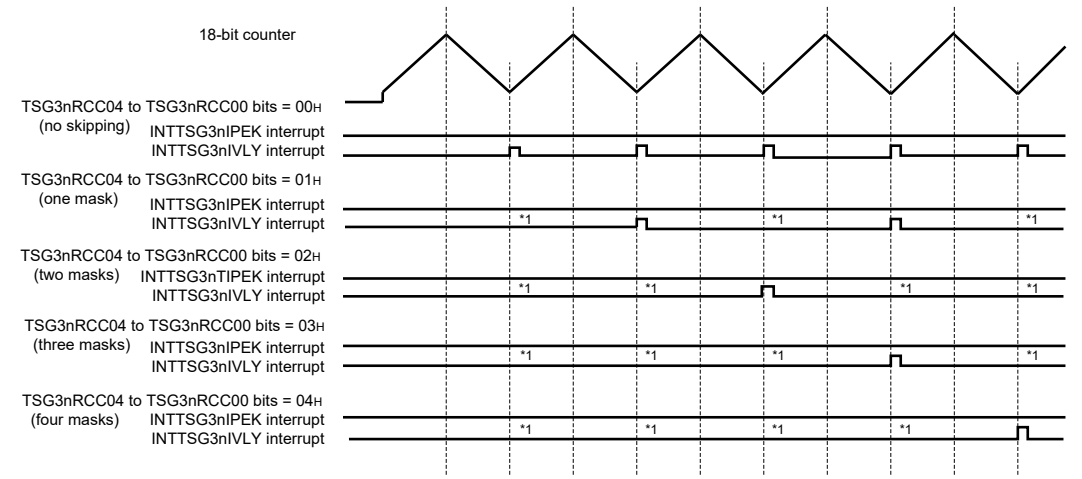
Note 1. Skipped interrupt request

Figure 20.34 Interrupt Skipping Operation when TSG3nPIE = 1 and TSG3nVIE = 1 in TSG3nCTL4 (Peak and Trough Interrupt Generation in HT-PWM Mode)



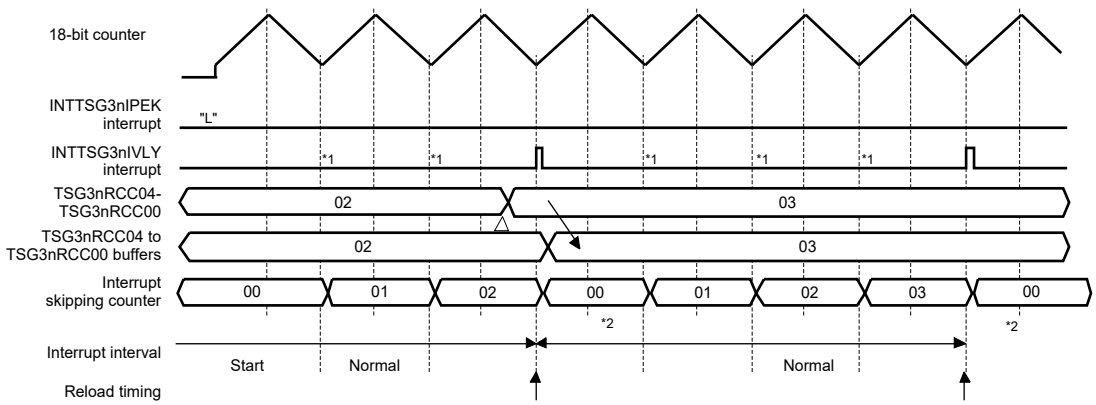
Note 1. Skipped interrupt request

Figure 20.35 Interrupt Skipping Operation when TSG3nPIE = 1 and TSG3nVIE = 0 in TSG3nCTL4 Register (only Peak Interrupt Generation in HT-PWM Mode)



Note 1. Skipped interrupt request

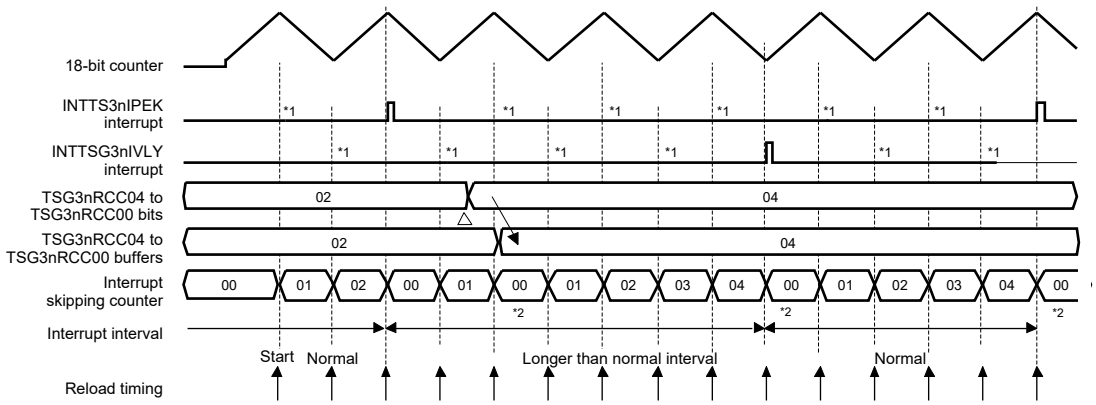
Figure 20.36 Interrupt Skipping Operation when TSG3nPIE = 0 and TSG3nVIE = 1 in TSG3nCTL4 Register (only Trough Interrupt Generation in HT-PWM Mode)



Note 1. Skipped interrupt request

Note 2. Interrupt skipping counter is cleared

Figure 20.37 When TSG3nRMC = 0, TSG3nRIA = 1 in TSG3nCTL3 (with Reload Skipping)



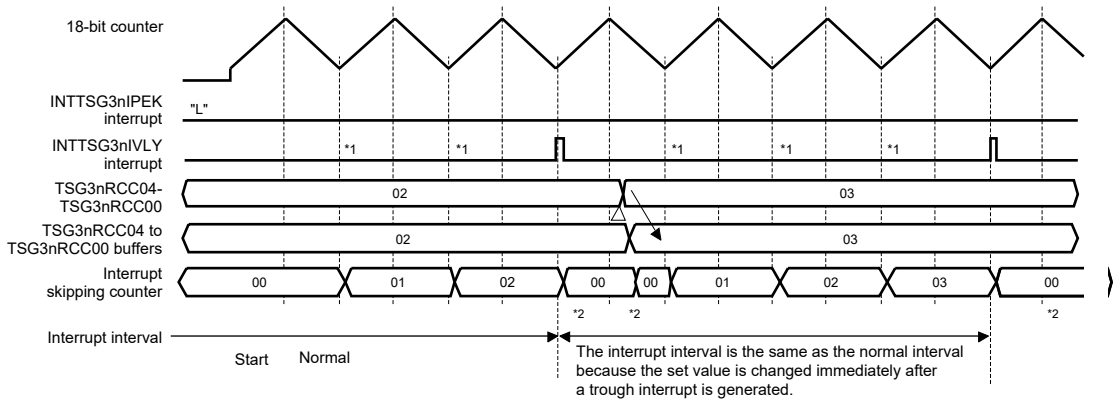
Note 1. Skipped interrupt request

Note 2. Interrupt skipping counter is cleared

Figure 20.38 When TSG3nRMC = 0, TSG3nRIA = 0 in TSG3nCTL3 (without Reload Skipping)

CAUTION

The interrupt interval might be longer.



Note 1. Skipped interrupt request

Note 2. Interrupt skipping counter is cleared

Figure 20.39 When TSG3nRMC = 1 in TSG3nCTL3 (Anytime Rewrite Mode)

NOTE

After rewriting, the value is reflected immediately regardless of the reload timing. The interrupt skipping counter is cleared when the value is transferred to the TSG3nRCC04 to TSG3nRCC00 buffers, not when the pertinent register is rewritten.

20.4.4.2 Example of Operation when Peak Interrupt is Generated (in PWM Mode)

Operation related to the interrupt skipping function in PWM mode is described below.

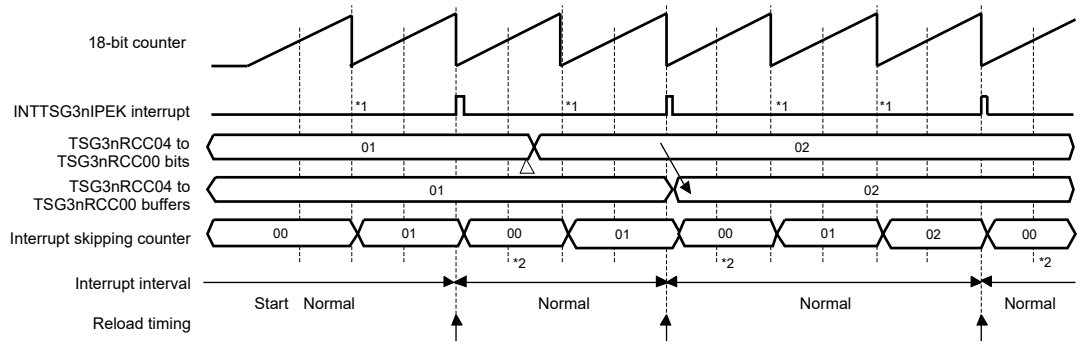
- Trough interrupts (INTTSG3nIPEK) can be skipped. In PWM mode, it is generated by compare match of TSG3nCMP0E buffer register and 18-bit counter.
- TSG3nCTL4.TSG3nPIE enables outputting of the INTTSG3nIPEK interrupt and specifies whether to skip the interrupts.
- The setting of TSG3nCTL4.TSG3nVIE is disabled. At this time, the INTTSG3nIVLY interrupt is not generated.

  When TSG3nCTL3.TSG3nRIA is set to 1 (with reload skipping), reload is executed at the same timing as the interrupt after being skipped.

CAUTION

When a value is written to TSG3nCTL4, and TSG3nRCC04 to TSG3nRCC00 are transferred to the buffer register, the interrupt skipping counter is cleared. Therefore, when the interrupt skipping function is used, interrupt interval may be long temporarily. To avoid this, the interrupt skipping count should be changed with the reload timing being set to the interrupts skipped (TSG3nCTL3.TSG3nRIA = 1).

Example of operation



Note 1. Skipped interrupt request

Note 2. Interrupt skipping counter is cleared.

Figure 20.40 When TSG3nCTL3.TSG3nRMC = 0, TSG3nRIA = 1, TSG3nCTL4.TSG3nPRE = 1 (Recommended Setting)

NOTE

When TSG3nCTL3.TSG3nRIA = 1, reload is executed at the same timing as the interrupt after being skipped.

20.4.5 A/D Conversion Trigger Function

A/D conversion trigger operation is described below.

The TSG3nDCMP0E, TSG3nDCMP1E, and TSG3nDCMP2E registers are used as compare registers of the A/D conversion trigger function.

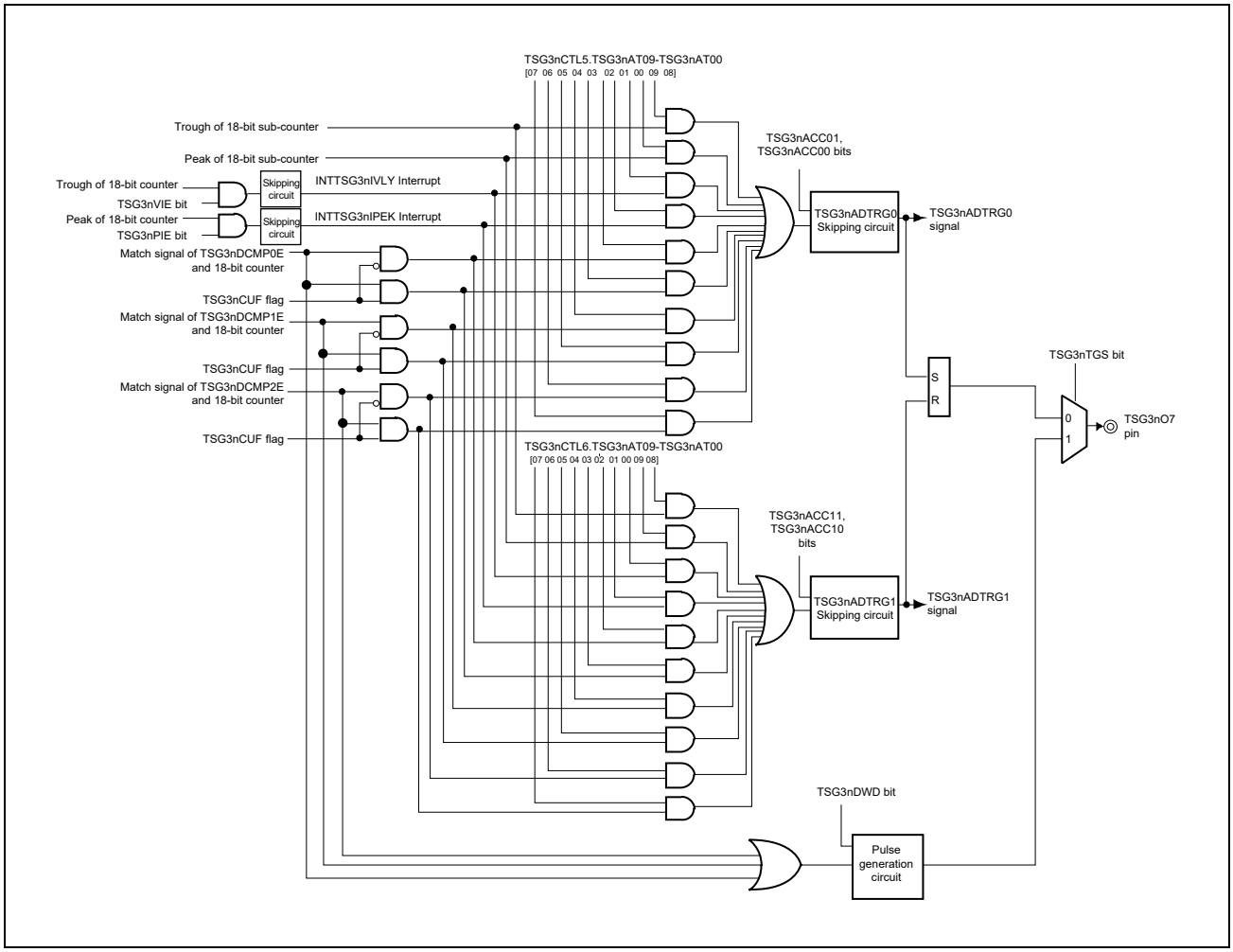


Figure 20.41 A/D Conversion Trigger and Diagnostic Output Control Circuit

As shown in **Figure 20.41**, a logical ORed signal can be generated by selecting the compare match of TSG3nDCMP0E to TSG3nDCMP2E with the 18-bit counter, a peak interrupt (INTTSG3nIPEK) and a trough interrupt (INTTSG3nIVLY) of the 18-bit counter, the peak timing of 18-bit sub-counter, and the trough timing of 18-bit sub-counter.

TSG3n has two channels of the identical A/D conversion trigger control circuits, which can be controlled independently. TSG3n also provides the A/D conversion trigger skipping function with the skipping rate of 1/1, 1/2, 1/4, or 1/8.

20.4.5.1 Operation of A/D Conversion Trigger

TSG3n has a function to generate A/D conversion start triggers (TSG3nADTRG0 and TSG3nADTRG1 signals) by selecting any of ten trigger sources as required. The trigger sources are selected by TSG3nAT09 to TSG3nAT00 in TSG3nCTL5 and TSG3nAT19 to TSG3nAT10 in TSG3nCTL6.

(1) TSG3nADTRG0/TSG3nADTRG1 Signal Output Control (TSG3nCTL5 and TSG3nCTL6)

[Trigger sources]

- TSG3nAT00/TSG3nAT10 = 1 : A trough interrupt (INTTSG3nIVLY) causes an A/D conversion trigger pulse to be generated.
- TSG3nAT01/TSG3nAT11 = 1 : A peak interrupt (INTTSG3nIPEK) causes an A/D conversion trigger pulse to be generated.
- TSG3nAT02/TSG3nAT12 = 1 : While the 18-bit counter is counting up, a TSG3nDCMP0E compare match enables A/D conversion trigger to be generated.
- TSG3nAT03/TSG3nAT13 = 1 : While the 18-bit counter is counting down, a TSG3nDCMP0E compare match enables A/D conversion trigger to be generated.
- TSG3nAT04/TSG3nAT14 = 1 : While the 18-bit counter is counting up, a TSG3nDCMP1E compare match enables A/D conversion trigger to be generated.
- TSG3nAT05/TSG3nAT15 = 1 : While the 18-bit counter is counting down, a TSG3nDCMP1E compare match enables A/D conversion trigger to be generated.
- TSG3nAT06/TSG3nAT16 = 1 : While the 18-bit counter is counting up, a TSG3nDCMP2E compare match enables A/D conversion trigger to be generated.
- TSG3nAT07/TSG3nAT17 = 1 : While the 18-bit counter is counting down, a TSG3nDCMP2E compare match enables A/D conversion trigger to be generated.
- TSG3nAT08/TSG3nAT18 = 1 : A trough timing of the 18-bit sub-counter (at a switch from decrementing to incrementing) enables A/D conversion trigger to be generated.
- TSG3nAT09/TSG3nAT19 = 1 : A peak timing of 18-bit sub-counter (at a switch from incrementing to decrementing) enables A/D conversion trigger to be generated.

[Skipping setting]

- TSG3nACC01, TSG3nACC00 and TSG3nACC11, TSG3nACC10 :
  Set the skipping rate of TSG3nADTRG0/TSG3nADTRG1

All A/D conversion triggers selected by TSG3nAT09 to TSG3nAT00 and TSG3nAT19 to TSG3nAT10 are logically ORed, and the resultant signals are subjected to skipping control specified by TSG3nACC01 and TSG3nACC00, and TSG3nACC11 and TSG3nACC10, and then the TSG3nADTRG0 and TSG3nADTRG1 signals are generated.

A peak interrupt (INTTSG3nIPEK) and a trough interrupt (INTTSG3nIVLY) selected by TSG3nAT00 and TSG3nAT01, and TSG3nAT10 and TSG3nAT11 are interrupt signals obtained after skipped. Therefore, they are output at the timing according to interrupt skipping control. If the interrupt output is not enabled by TSG3nCTL4.TSG3nPIE and TSG3nVIE, A/D conversion trigger is not output.

TSG3nACC01, TSG3nACC00, and TSG3nAT09 to TSG3nAT00, and TSG3nACC11, TSG3nACC10, and TSG3nAT19 to TSG3nAT10 can be rewritten during timer operation.

If A/D conversion trigger setting bits are rewritten during operation, the rewritten values are reflected on the A/D conversion trigger output status immediately. Such control bits are rewritten at anytime regardless of operating modes. If a write access is made to TSG3nCTL5 and TSG3nCTL6 (including a rewrite of the same value), the A/D conversion trigger skipping counter is cleared and starts counting from zero.

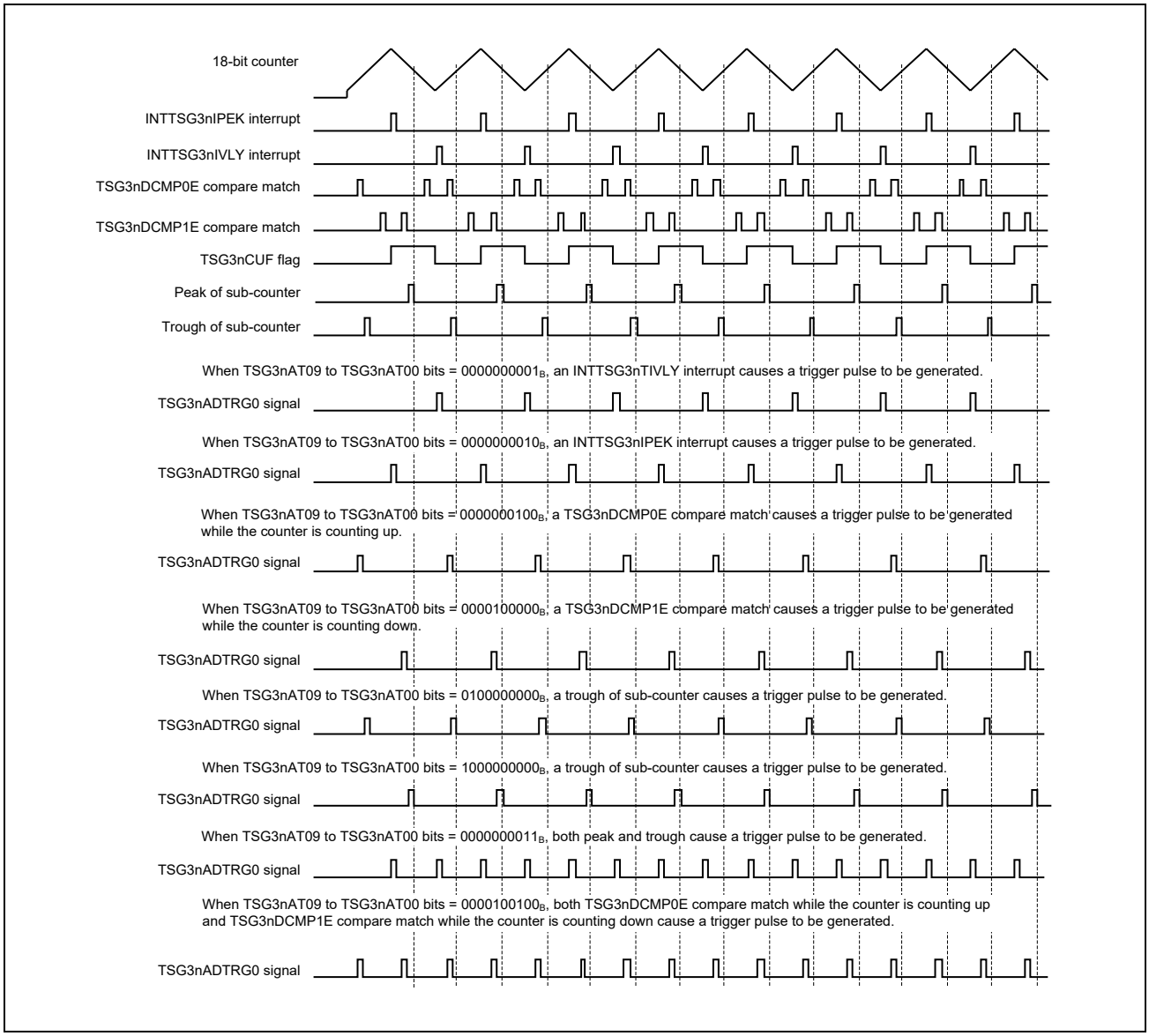
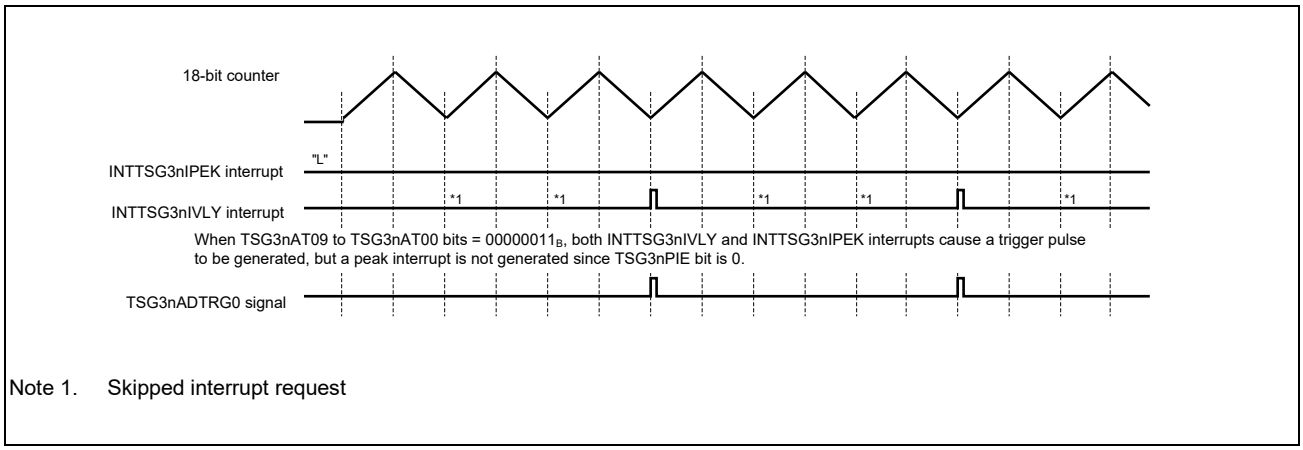
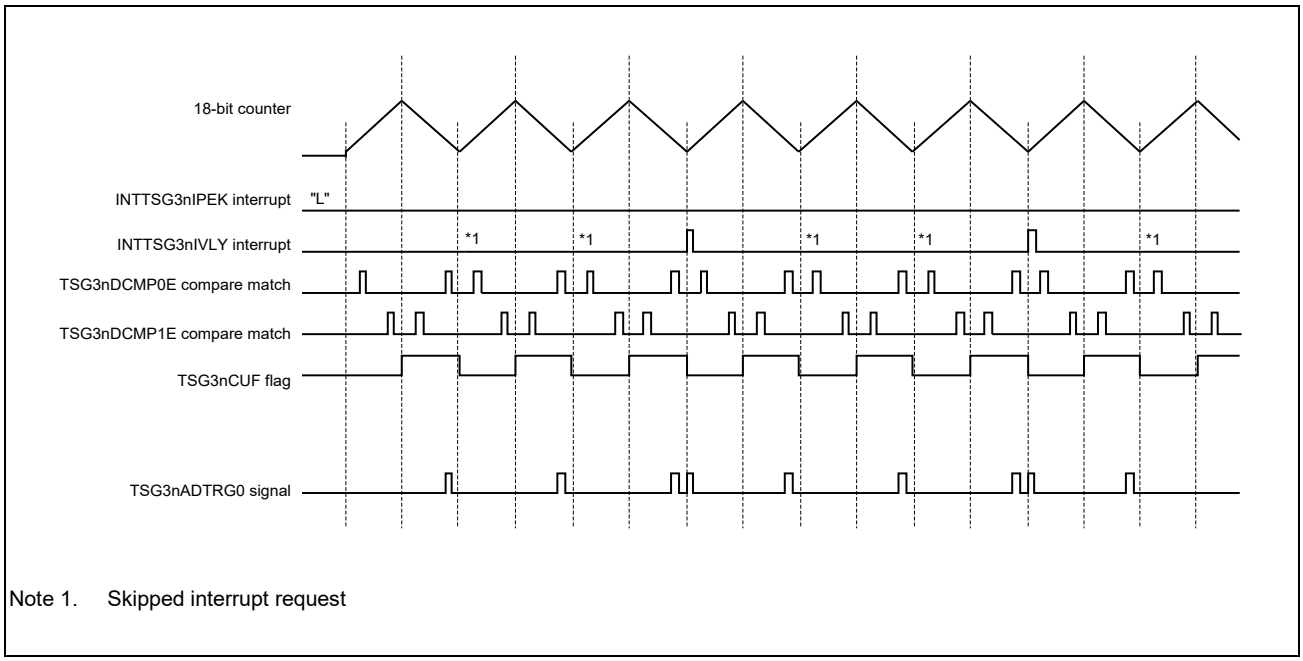


Figure 20.42 When TSG3nPIE = 1, TSG3nVIE = 1, and TSG3nRCC04 to TSG3nRCC00 = 00_H in TSG3nCTL4, and TSG3nACC01 and TSG3nACC00 = 00_B in TSG3nCTL5 (HT-PWM Mode)



Note 1. Skipped interrupt request

Figure 20.43 When TSG3nPIE = 0, TSG3nVIE = 1, and TSG3nRCC04 to TSG3nRCC00 = 02_H in TSG3nCTL4 and TSG3nACC01 and TSG3nACC00 = 00_B in TSG3nCTL5 (HT-PWM Mode)



Note 1. Skipped interrupt request

Figure 20.44 When TSG3nPIE = 0, TSG3nVIE = 1, and TSG3nRCC04 to TSG3nRCC00 = 02_H in TSG3nCTL4 and TSG3nACC01 and TSG3nACC00 = 00_B, and TSG3nAT09 to TSG3nAT00 = 0000 1001_B in TSG3nCTL5 (HT-PWM Mode)

(2) A/D Conversion Trigger Skipping Function

Example of operation of the A/D conversion trigger skipping function is shown in **Figure 20.45.**

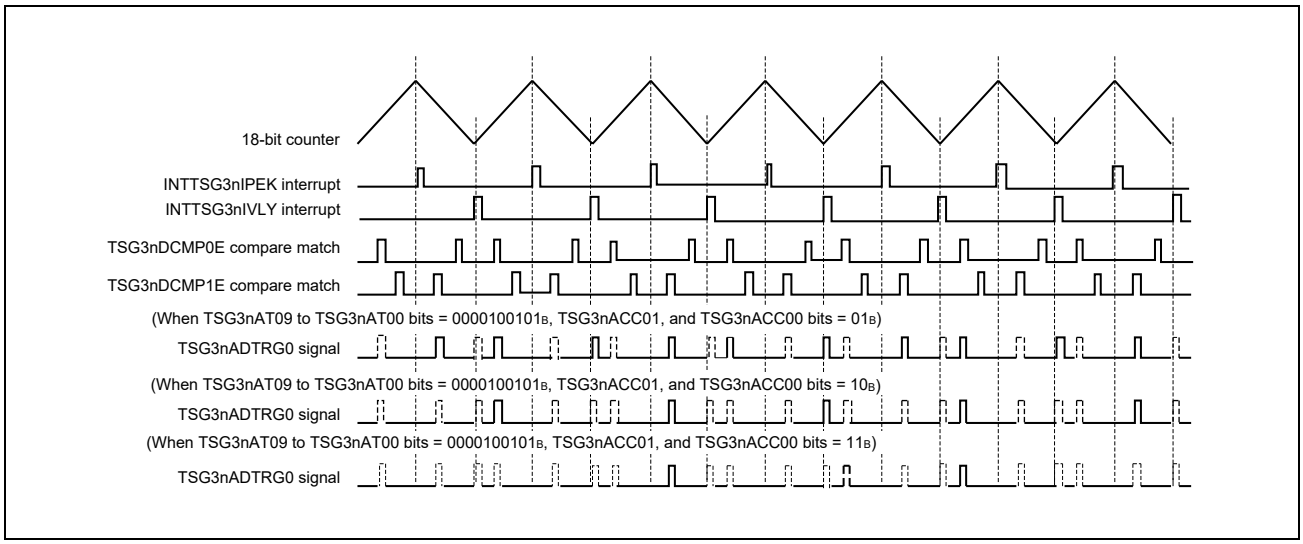


Figure 20.45 Example of Operation of A/D Conversion Trigger Skipping Function

NOTE

Broken-lined pulses indicate A/D conversion trigger pulses skipped by the A/D conversion trigger skipping function.

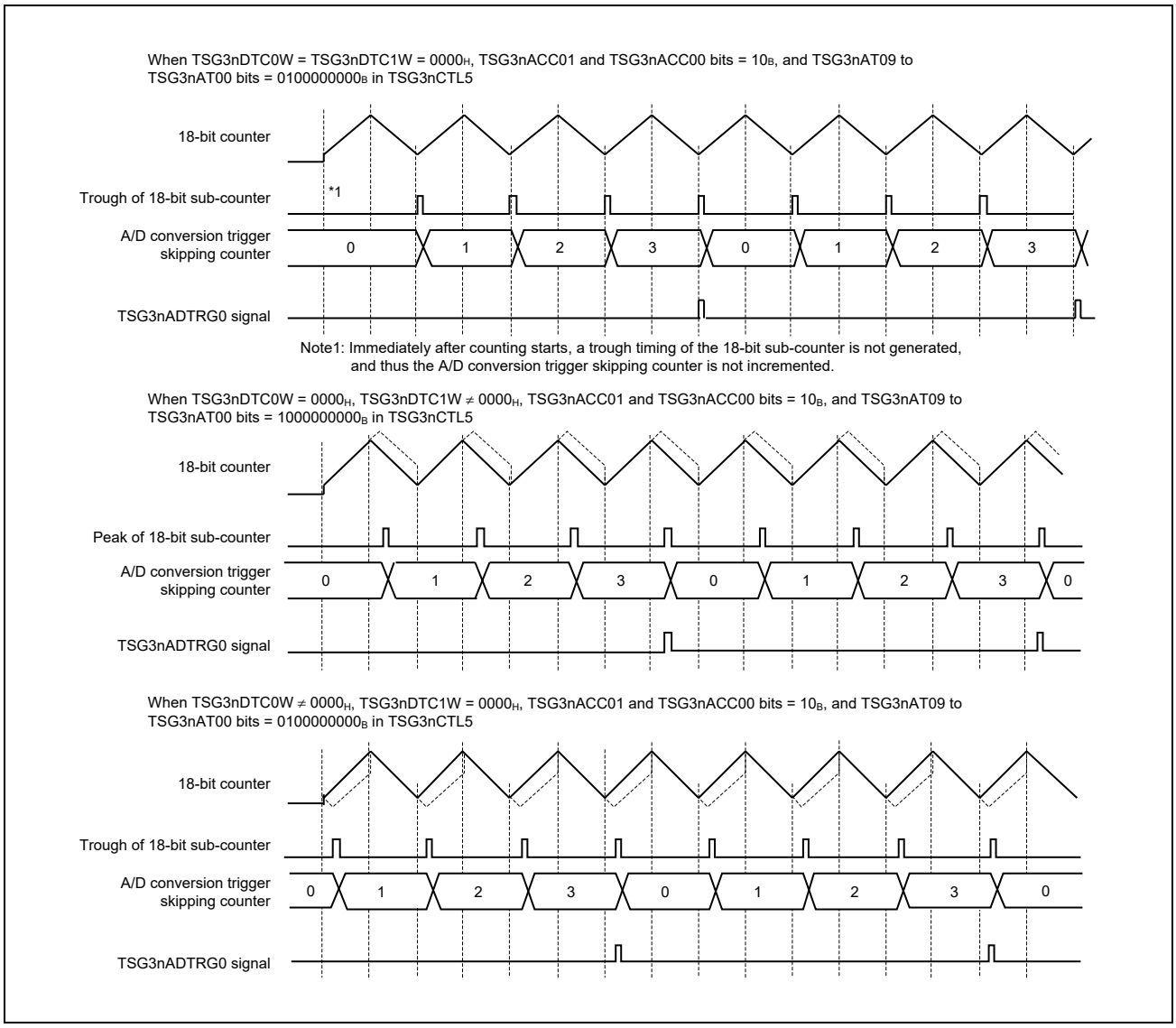


Figure 20.46 Example of Operation of A/D Conversion Trigger Skipping Function

(3) Notes on A/D Conversion Trigger

- If the same value is written to TSG3nDCMP0E and TSG3nDCMP1E or TSG3nDCMP2E, and the same condition (when the 18-bit counter increments or decrements) is set as the valid A/D conversion trigger, A/D conversion trigger skipping counter is incremented by one and one trigger pulse is output upon a match of the 18-bit counter with these registers.
- In PWM mode, SP-PWM mode, 120-DC mode, and HSP-PWM mode, a trough interrupt (INTTSG3nIVLY) is not generated. Only a peak interrupt (INTTSG3nIPEK) is valid.
- In 120-DC mode, when TSG3nS120DCO is set to 0, the 18-bit counter may be cleared during the carrier period due to switch of the output pattern. The A/D conversion trigger is not generated if TSG3nDCMP2E to TSG3nDCMP0E values do not match with the 18-bit counter value and a peak interrupt (INTTSG3nIPEK) is not generated.

20.4.6 Error/Warning Interrupt

20.4.6.1 Error Interrupt Function

If the simultaneous active state of the positive phase and inverse phase is detected after the error interrupt function is enabled (TSG3nIOC1.TSG3nEOC = 1), TSG3nSTR2.TSG3nTBF is set, and an error interrupt (INTTSG3nIER) of TSG3n is generated. Whether or not to detect an error of each phase (TSG3nO1 and TSG3nO2, TSG3nO3 and TSG3nO4, and TSG3nO5 and TSG3nO6 pins) can be selected by TSG3nCTL1.TSG3nTBA2 to TSG3nTBA0, respectively.

When an error occurs, outputs of the TSG3nO1 to TSG3nO6 pins can be set to high-impedance. For details, see **Section 21.4.1, Asynchronous Hi-Z Control Function**.

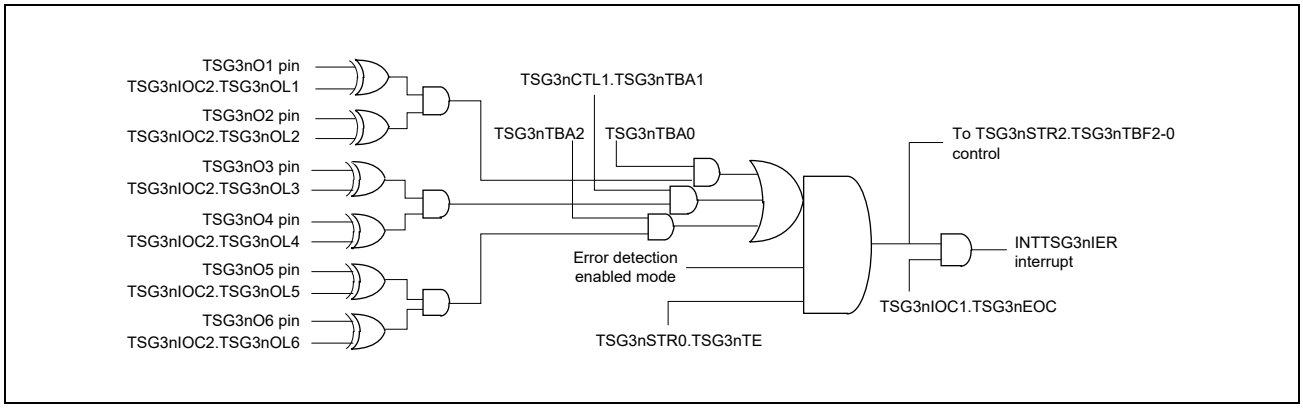


Figure 20.47 Error Interrupt (INTTSG3nIER) Generation Control Circuit

CAUTION

When an error interrupt is generated, the error status should be canceled (write 1 to TSG3nSTC.TSG3nTBR2 to TSG3nSTC.TSG3nTBR0) during an error interrupt handling. Otherwise, subsequent error interrupts are not generated.

(1) PWM Mode, 120-DC Mode and HSP-PWM Mode

In PWM mode, and HSP-PWM Mode if TSG3nCMP1E and TSG3nCMP2E, and TSG3nCMP3E and TSG3nCMP4E are set so that the TSG3nO1 and TSG3nO2 pins output the active level simultaneously, an error interrupt (INTTSG3nIER) is generated. With the same setting, if TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP9E, TSG3nCMP10E, TSG3nCMP11E, and TSG3nCMP12E are set so that the TSG3nO3 and TSG3nO4, and TSG3nO5 and TSG3nO6 pins output the active level simultaneously, an error interrupt (INTTSG3nIER) is generated.

In 120-DC mode, if TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, TSG3nCMP10E, TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, TSG3nCMP12E, TSG3nPAT0W, and TSG3nPAT1W are set so that the TSG3nO1 and TSG3nO2 pins output the active level simultaneously, an error interrupt (INTTSG3nIER) is generated. With the same setting, the TSG3nO3 and TSG3nO4, and TSG3nO5 and TSG3nO6 pins also output the active level simultaneously, an error interrupt (INTTSG3nIER) is generated.

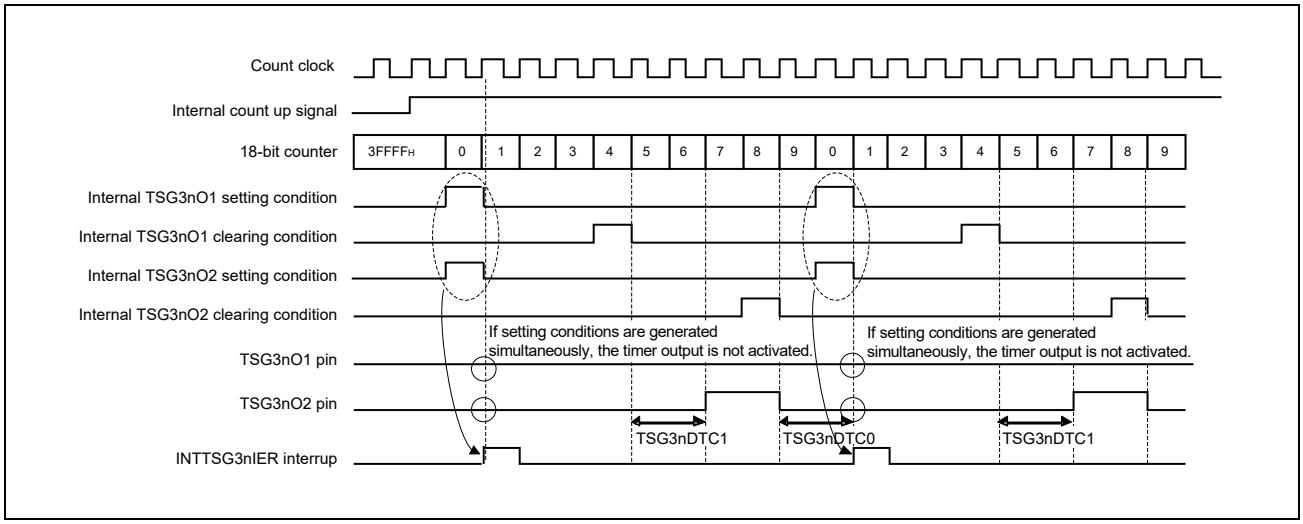


Figure 20.48 Example of Error Interrupt (INTTSG3nIER) Generation (PWM Mode)

NOTE

TSG3nO3 and TSG3nO4, and TSG3nO5 and TSG3nO6 behave the same.

When the active level of output is switched by operation TSG3nIOC2.TSG3nOL1 and TSG3nOL2, an error interrupt is generated as shown in the following figure.

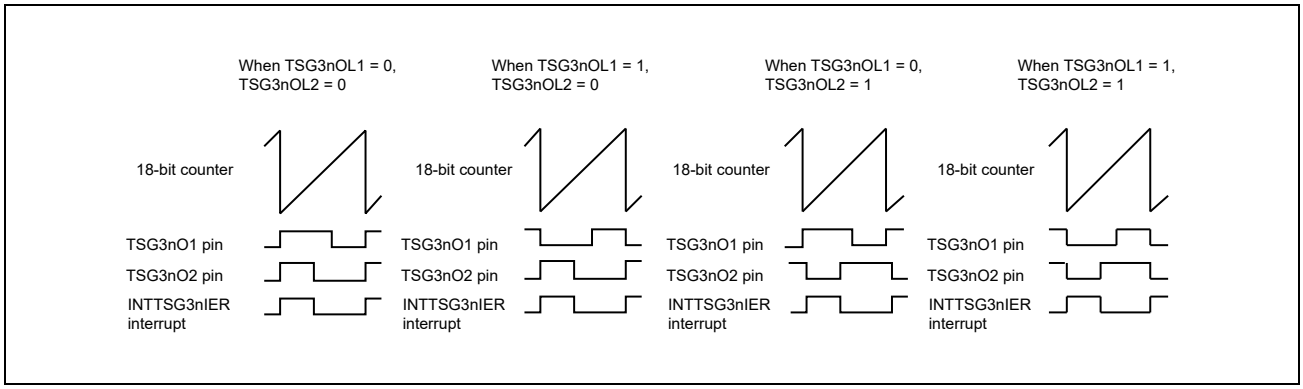


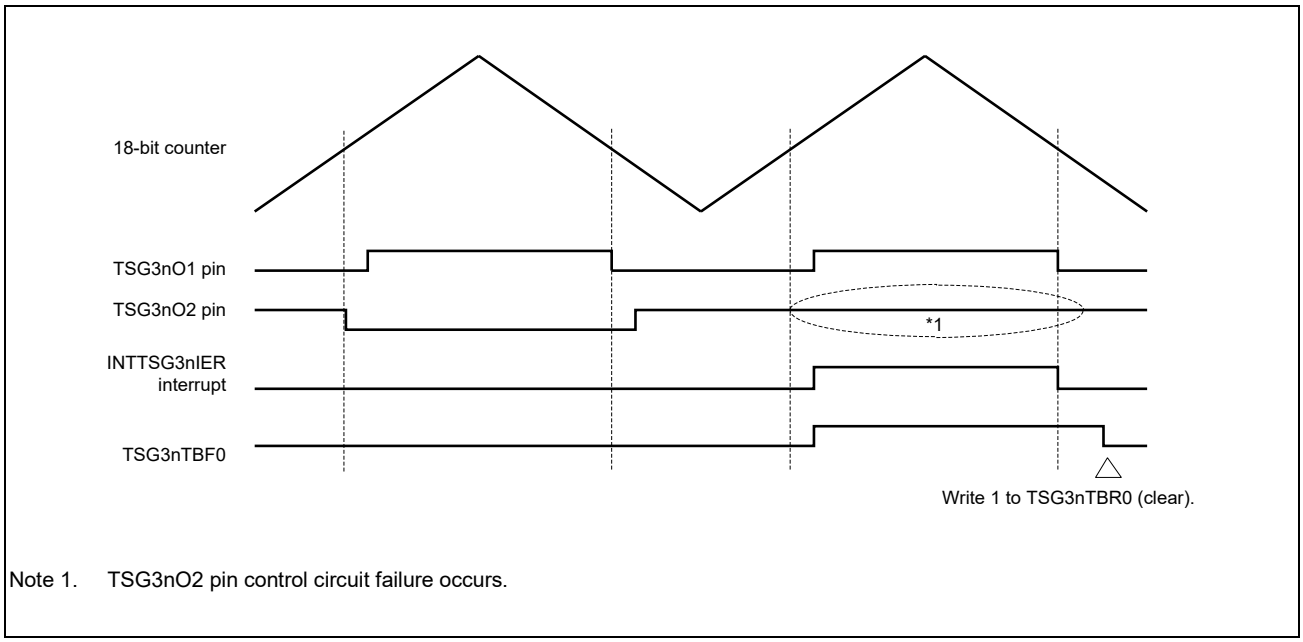
Figure 20.49 Example of Error Interrupt (INTTSG3nIER) Generation for each Active Level

(2) HT-PWM Mode and SP-PWM Mode

Either TSG3n dead time setting register 0 or 1 (TSG3nDTC0W or TSG3nDTC1W) is 0000_H, an error may occur.

NOTE

If an error occurs when the dead time control function is used (both TSG3nDTC0W and TSG3nDTC1W are not 0000_H), internal circuit failure may occur.



Note 1. TSG3nO2 pin control circuit failure occurs.

Figure 20.50 Example of Error Interrupt Operation

20.4.6.2 Warning Interrupt Function

TSG3n has a warning interrupt (INTTSG3nIWN).

Warning interrupt (INTTSG3nIWN) is generated when any of the following conditions is detected. For details, see Section **20.4.3, Flags**.

- When simultaneous change in two or more pins of TSG3nPTSI2 to TSG3nPTSI0 is detected:
  See **Section 20.4.3.4, Noise Detection Flag (TSG3nNDF)**.
- When reversal is detected of the TSG3nPTSI2 to TSG3nPTSI0 pins:

  See **Section 20.4.3.7, Pattern Reversal Detection Flag (TSG3nPRF)**.
- When 000 or 111 is detected from the TSG3nPTSI2 to TSG3nPTSI0 pins:

  See **Section 20.4.3.6, Pattern Error Detection Flag (TSG3nPEF)**.
- When a toggle of the TSG3nPTSI2 to TSG3nPTSI0 pins is generated three times or more between TSG3nOPCI0 and TSG3nOPCI1 signal triggers:

  See **Section 20.4.3.8, TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag**.
- When the TSG3nOPCI0 and TSG3nOPCI1 signal triggers are detected simultaneously:

  See **Section 20.4.3.9, TSG3nOPCI0, TSG3nOPCI1 Signal Simultaneous Trigger Detection Flag**.
- When the phase difference between the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) and output pattern (TSG3nOPF2 to TSG3nOPF0) is detected:

  See **Section 20.4.3.10, Pattern Phase Difference Detection Flag (TSG3nPPF)**.

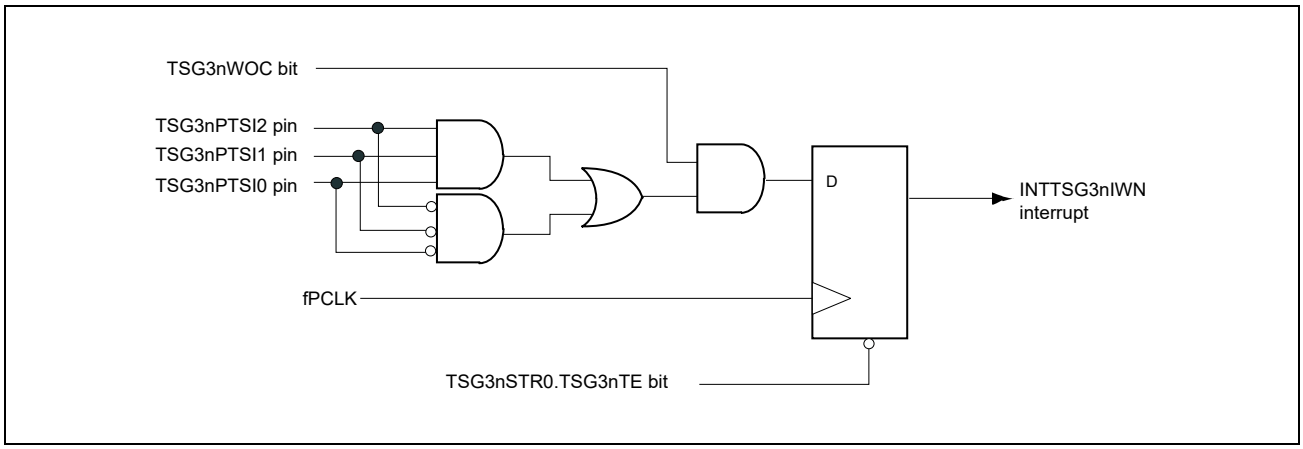


Figure 20.51 Detection of Abnormality of TSG3nPTSI2-TSG3nPTSI0 Pins

20.4.7 Operating Modes

20.4.7.1 PWM Mode

Overview

A PWM signal is output at the TSG3nO1 to TSG3nO6 pins according to set timing/clear timing of TSG3nCMP1E to TSG3nCMP12E registers with the PWM period set in the TSG3nCMP0E register.

Prerequisites

- Set the set timing to the compare register with an even number:

  TSG3nCMP2E (set timing of the TSG3nO1 output), TSG3nCMP4E (set timing of the TSG3nO2 output), TSG3nCMP6E (set timing of the TSG3nO3 output), TSG3nCMP8E (set timing of the TSG3nO4 output), TSG3nCMP10E (set timing of the TSG3nO5 output) and TSG3nCMP12E (set timing of the TSG3nO6 output)

- Set the clear timing to the compare register with an odd number:

  TSG3nCMP1E (clear timing of the TSG3nO1 output), TSG3nCMP3E (clear timing of the TSG3nO2 output), TSG3nCMP5E (clear timing of the TSG3nO3 output), TSG3nCMP7E (clear timing of the TSG3nO4 output), TSG3nCMP9E (clear timing of the TSG3nO5 output) and TSG3nCMP11E (clear timing of the TSG3nO6 output)

Functional description

Set the PWM period and set/clear timing of the TSG3nO1 to TSG3nO6 outputs. Set TSG3nTRG0.TSG3nTS = 1 to start the timer counter.

The TSG3nO1 to TSG3nO6 outputs are set to the inactive state at the same time counting begins. The outputs are set to the active state by the match of the buffer registers TSG3nCMP2E, TSG3nCMP4E, TSG3nCMP6E, TSG3nCMP8E, TSG3nCMP10E, and TSG3nCMP12E with the 18-bit counter.

Next, the TSG3nO1 to TSG3nO6 outputs are set to the inactive state by the match of the buffer registers TSG3nCMP1E, TSG3nCMP3E, TSG3nCMP5E, TSG3nCMP7E, TSG3nCMP9E, and TSG3nCMP11E with the 18-bit counter.

During counting, a compare match interrupt (INTTSG3nI0 to INTTSG3nI12) is generated by the match of the buffer register TSG TSG3nCMP0E-TSG3nCMP12E.

CAUTION

Reload is executed when writing to the TSG3nCMP1E register at TSG3nCTL3.TSG3nRMC = 0. Therefore, even when it is needed to rewrite only the value of the TSG3nCMP0E register, a write operation to the TSG3nCMP1E register is necessary. When only the TSG3nCMP0E register is rewritten, reload is not done.

NOTE

The PWM mode is set when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 000_B

(1) When TSG3nCMP0E and TSG3nCMP1E to TSG3nCMP12E are Not Rewritten during Timer Operation

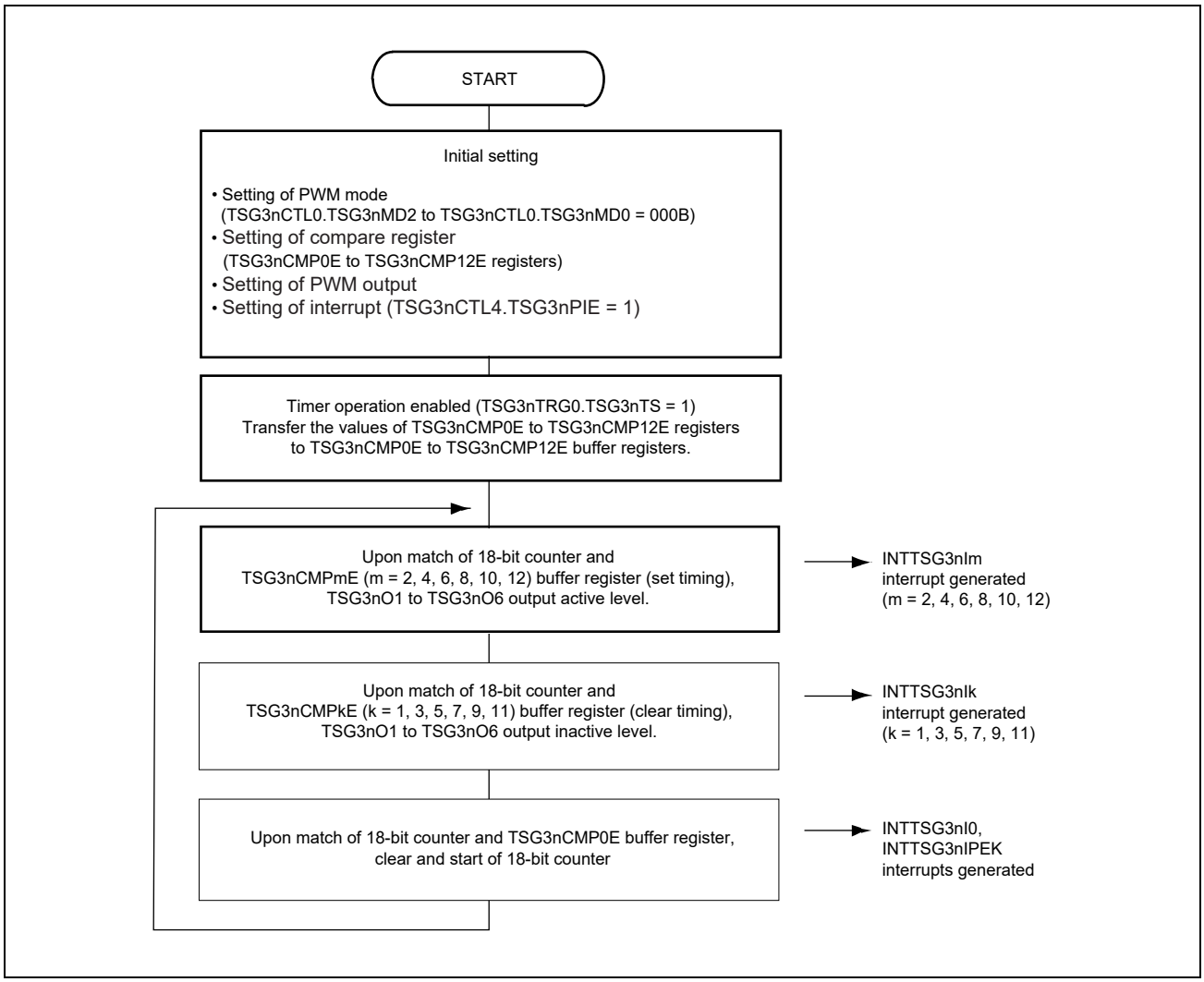
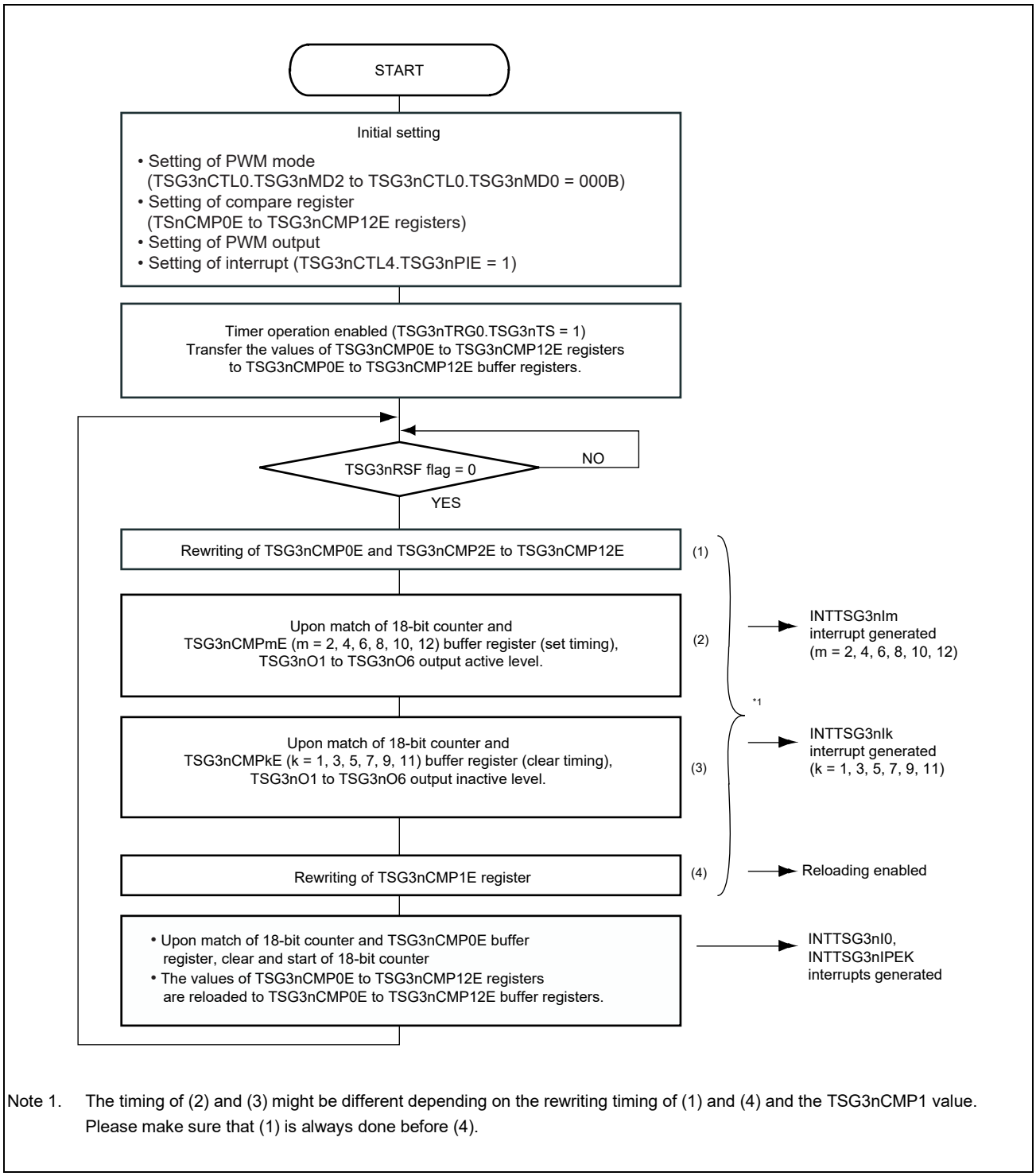


Figure 20.52 Basic Operation Flow of PWM Mode (1/2)

(2) When TSG3nCMP0E and TSG3nCMP1E to TSG3nCMP12E are Rewritten during Timer Operation



Note 1. The timing of (2) and (3) might be different depending on the rewriting timing of (1) and (4) and the TSG3nCMP1 value. Please make sure that (1) is always done before (4).

Figure 20.52 Basic Operation Flow of PWM Mode (2/2)

CAUTION

Please rewrite compare registers after confirming that the reload request flag TSG3nRSF is 0.

(3) List of Operations in PWM Mode

Table 20.55 Counter Functions in PWM Mode

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1, or a simultaneous start trigger
	Clear	Compare match of TSG3nCMP0E buffer register and 18-bit counter
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 20.56 Functions of Compare Registers and Dead Time Setting Register in PWM Mode

Register	Rewriting Methods	Rewrite During Operation	Function
TSG3nCMP0E	Reload/Anytime rewrite	Possible	Setting period
TSG3nCMPmE (m = 1 to 12)	Reload/Anytime rewrite	Possible	Setting set/clear timing
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload/Anytime rewrite	Possible	Diagnostic output or A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Possible*1	Setting dead time

Note 1. For details, see **(5), Controlling Dead Time in PWM Mode**.

Table 20.57 Timer Output in PWM Mode

Pin	Function
TSG3nOm (m = 1 to 6)	PWM output by compare match of TSG3nCMPkE buffer register and 18-bit counter (k = 1 to 12)
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger

Table 20.58 Interrupt Requests in PWM Mode

Interrupt	Function
INTTSG3nIm (m = 0 to 12)	Compare match of TSG3nCMPmE buffer register and 18-bit counter (m = 0 to 12)
INTTSG3nIER	Error (simultaneous active state of TSG3nO1 and TSG3nO2, or TSG3nO3 and TSG3nO4, or TSG3nO5 and TSG3nO6)
INTTSG3nIVLY	—
INTTSG3nIPEK	Peak interrupt (generated at the same timing as INTTSG3nI0)
INTTSG3nIWN	Warning interrupt

Note: "—": Not available in PWM mode

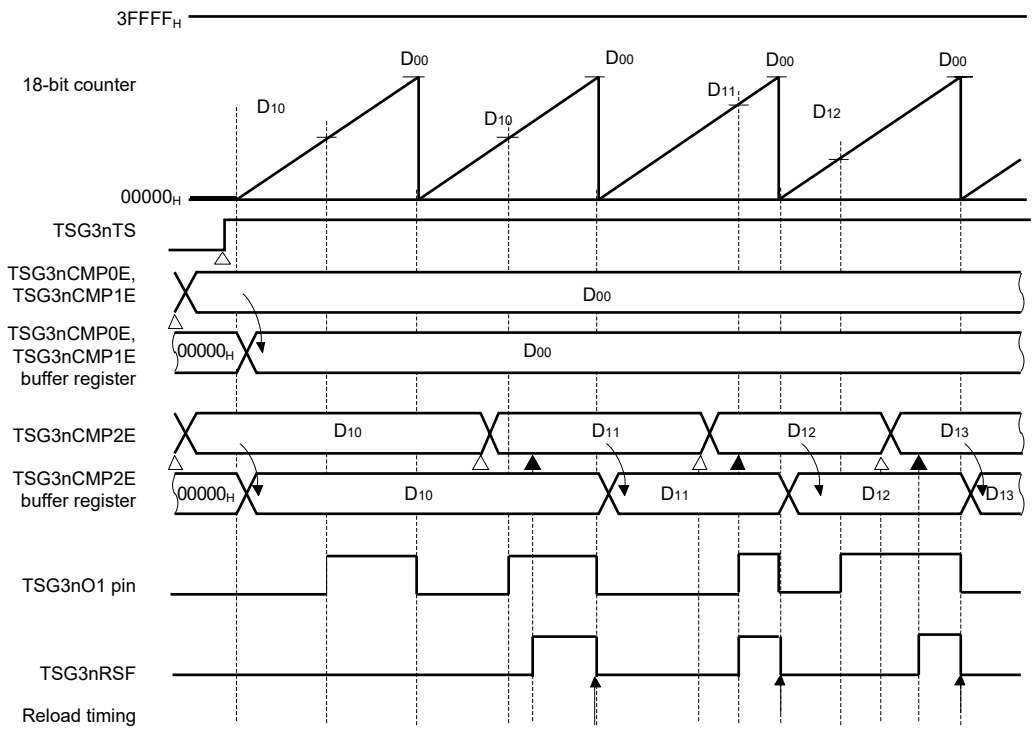
Table 20.59 Compare Match Timing in PWM Mode

Compare Match	Timing
TSG3nCMP0E	When 18-bit counter changes from TSG3nCMP0E to 00000_H
TSG3nCMPmE (m = 1 to 12)	After detecting the match of 18-bit counter and TSG3nCMPmE (m = 1 to 12)

Table 20.60 Example of Setting each Timer Output Condition in PWM Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSG3nOm (m = 1 to 6)	PWM output	(TSG3nCMP0E + 1) × count clock	Output an inactive level throughout one period (duty cycle 0%)	TSG3nCMPmE = TSG3nCMP (m + 1)E or TSG3nCMP (m + 1) E> TSG3nCMP0E (m = 1, 3, 5, 7, 9, 11)
			Output an active level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m + 1)E + 1 TSG3nCMP (m + 1)E = TSG3nCMPmE - 1 (m = 1, 3, 5, 7, 9, 11)
			Output an inactive level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m + 1)E - 1 SG3nCMP (m + 1)E = TSG3nCMPmE + 1 (m = 1, 3, 5, 7, 9, 11)
			Output an active level throughout one period (duty cycle 100%)	TSG3nCMPmE > TSG3nCMP0E TSG3nCMP (m + 1) E ≤ TSG3nCMP0E (m = 1, 3, 5, 7, 9, 11)

- When only TSG3nCMP2E is rewritten and the TSG3nO1 pin output is used (TSG3nIOC0.TSG3nTOE1 = 1, TSG3nIOC2.TSG3nOL1 = 0)



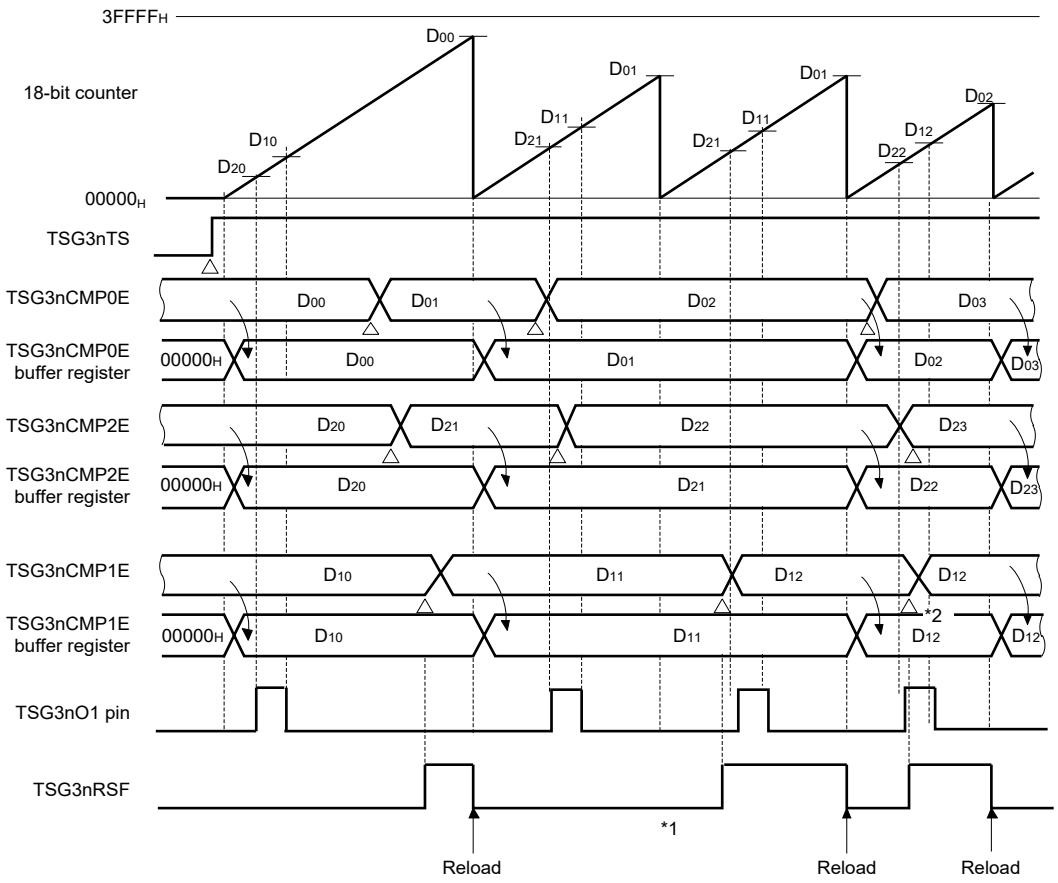
Note: Δ: Write access
▲: TSG3nCMP1E write access (equivalent)

Figure 20.53 Example of Basic Operation Timing in PWM Mode (1/2)

NOTES

1. D00: Set values of TSG3nCMP0E and TSG3nCMP1E (00000_H-$3FFFF_H$)
   D10, D11, D12 and D13: Set values of TSG3nCMP2E (00000_H-$3FFFF_H$)
2. TSG3nO1 (PWM) duty cycle = (TSG3nCMP1E-TSG3nCMP2E) × (count lock)
   TSG3nO1 (PWM) period = (Set value of TSG3nCMP0E + 1) × (count lock)
3. TSG3nO2-TSG3nO6 pins behave similarly to the TSG3nO1 pin

- When TSG3nCMP0E-TSG3nCMP2E are rewritten, and the TSG3nO1 pin output is used (TSG3nIOC0.TSG3nTOE1 = 1, TSG3nIOC2.TSG3nOL1 = 0)



Note 1. The TSG3nCMP0E buffer register is not reloaded, because the TSG3nCMP1E register was not rewritten.

Note 2. Equivalent writing

Note: Δ: Write access

Figure 20.53 Example of Basic Operation Timing in PWM Mode (2/2)

NOTES

1. D00, D01, D02, D03: Set point of TSG3nCMP0E (00000_H-$3FFFF_H$)
   D10, D11, D12, D13: Set point of TSG3nCMP1E (00000_H-$3FFFF_H$)
   D20, D21, D22, D23: Set point of TSG3nCMP2E (00000_H-$3FFFF_H$)
2. Outputs from TSG3nO2 to TSG3nO6 behave similarly to the TSG3nO1 pin.

(4) Interrupt/Reload Skipping Function in PWM Mode

By setting TSG3nCTL4.TSG3nPRE and TSG3nPIE to 1 and setting TSG3nRCC04 to TSG3nRCC00 and TSG3nCTL3.TSG3nRIA, the reload and interrupt skipping function can be used.

By setting TSG3nPRE to 1 and setting TSG3nRCC04 to TSG3nRCC00, the interrupt skipping function can be used.

(5) Controlling Dead Time in PWM Mode

By setting the dead time value in the TSG3nDTC0W and TSG3nDTC1W registers in PWM mode, it is possible to control the dead time. The dead time is controlled according to the switch timing of the TSG3nO1 and TSG3nO2 pin outputs, the TSG3nO3 and TSG3nO4 pin outputs, and the TSG3nO5 and TSG3nO6 pin outputs.

Table 20.61 Dead Time in PWM Mode

Switch Timing	Dead Time
TSG3nO1: High level to low level TSG3nO2: Low level to high level	Value of TSG3nDTC1W register
TSG3nO2: High level to low level TSG3nO1: Low level to high level	Value of TSG3nDTC0W register
TSG3nO3: High level to low level TSG3nO4: Low level to high level	Value of TSG3nDTC1W register
TSG3nO4: High level to low level TSG3nO3: Low level to high level	Value of TSG3nDTC0W register
TSG3nO5: High level to low level TSG3nO6: Low level to high level	Value of TSG3nDTC1W register
TSG3nO6: High level to low level TSG3nO5: Low level to high level	Value of TSG3nDTC0W register

NOTE

Dead time counter keeps operating even when operation stop (TSG3nTE = 0) setting collided with dead time insert timing, and the dead time set in TSG3nO1 and TSG3nO2 are always inserted.

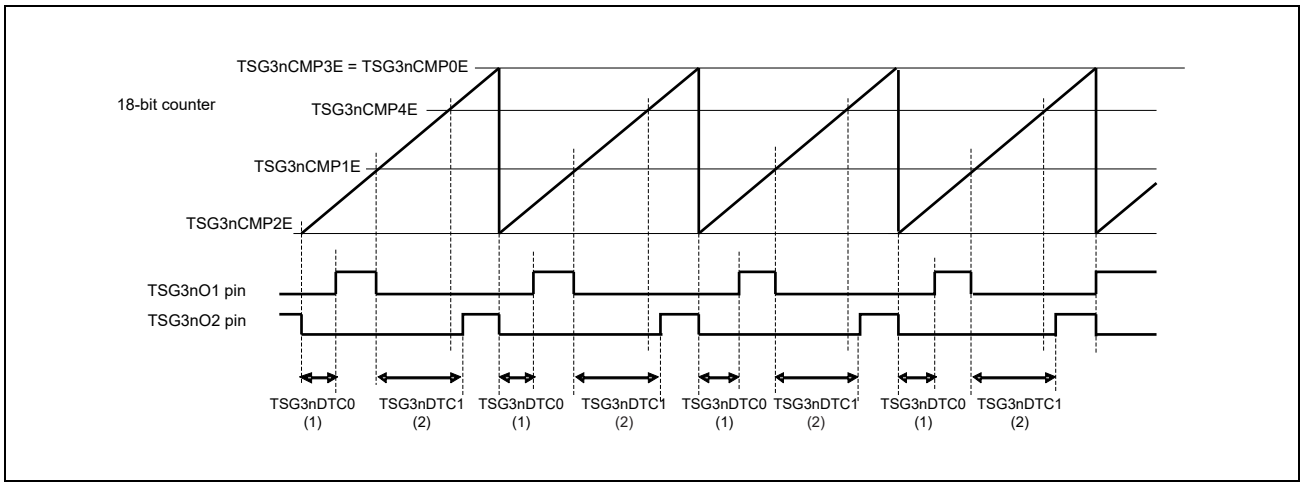


Figure 20.54 Example of Dead Time Control between TSG3nO1 and TSG3nO2 Outputs (1/2)

During (1), the dead time counter starts counting at the falling edge of the TSG3nO2 output. At this time, even after the 18-bit counter reaches 00000_H, the TSG3nO1 output stays inactive because the dead time counter is still operating. The TSG3nO1 output becomes active at the timing when the dead time count operation ends.

At (2), the dead time counter starts counting at the falling edge of the TSG3nO1 output. Even after the match of the 18-bit counter and TSG3nCMP4E, the TSG3nO2 output stays inactive because the dead time counter is still operating. The TSG3nO2 output becomes active at the timing when the dead time count operation ends.

NOTES

1. The TSG3nO1 and TSG3nO2 pins are set to active high
2. The TSG3nO3 and TSG3nO4 pins and the TSG3nO5 and TSG3nO6 pins outputs behave similarly.

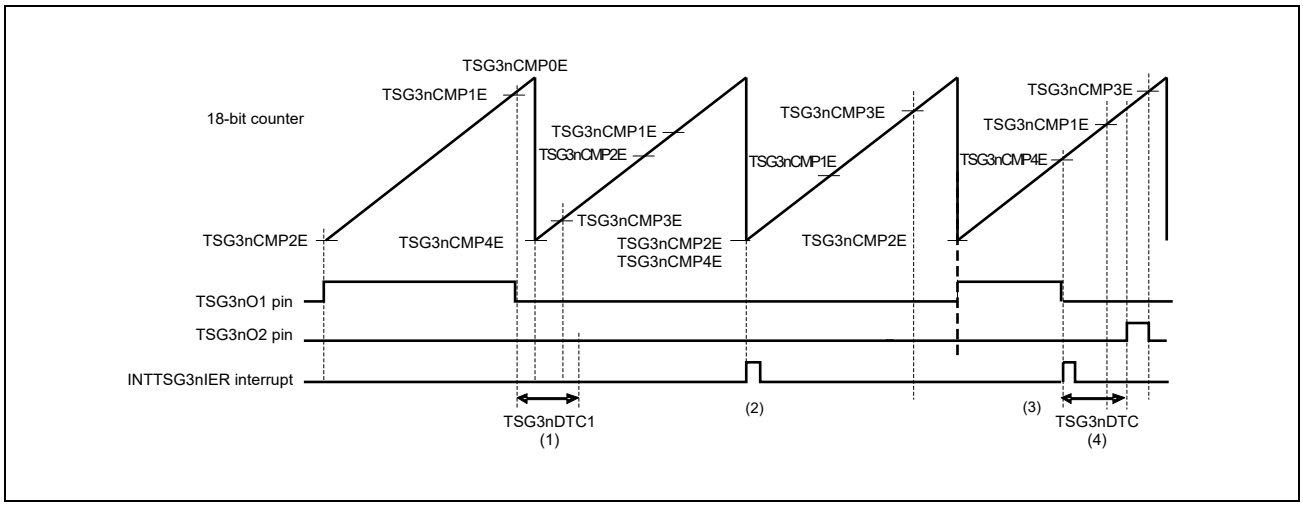


Figure 20.54 Example of Dead Time Control between TSG3nO1 and TSG3nO2 Outputs (2/2)

During (1), the dead time counter starts counting at the falling edge of the TSG3nO1 output. Even after the 18-bit counter reaches 00000_H and the match occurs between the 18-bit counter and TSG3nCMP4E, the TSG3nO2 output stays inactive because the dead time counter is still operating. Moreover, since the TSG3nCMP3E register compare match occurs before the operation of the dead time counter ends, the TSG3nO2 output stays inactive.

- TSG3nCMP1E + TSG3nDTC1 ≥ TSG3nCMP0E + TSG3nCMP2E (TSG3nO2 stays inactive)
- TSG3nCMP2E + TSG3nDTC0 ≥ TSG3nCMP0E + TSG3nCMP1E (TSG3nO1 stays inactive)

At (2), the INTTSG3nIER interrupt occurs because the TSG3nCMP2E register and the TSG3nCMP4E register are set so that the TSG3nO1 and TSG3nO2 outputs rise simultaneously. Here, the TSG3nO1 output and the TSG3nO2 output are inactive.

At (3), compare match with the TSG3nCMP4E register generates an INTTSG3nIER interrupt and both TSG3nO1 and TSG3nO2 outputs become inactive.

At (4), the falling edge (inactive) of the TSG3nO1 output is caused by the simultaneous active state detected and the dead time counter starts counting. After the end of the dead time counter operation, the TSG3nO2 output becomes active.

NOTES

1. The TSG3nO1 and TSG3nO2 pins are set to active high.
2. The TSG3nO3 and TSG3nO4 pins and the TSG3nO5 and TSG3nO6 pins outputs behave similarly.

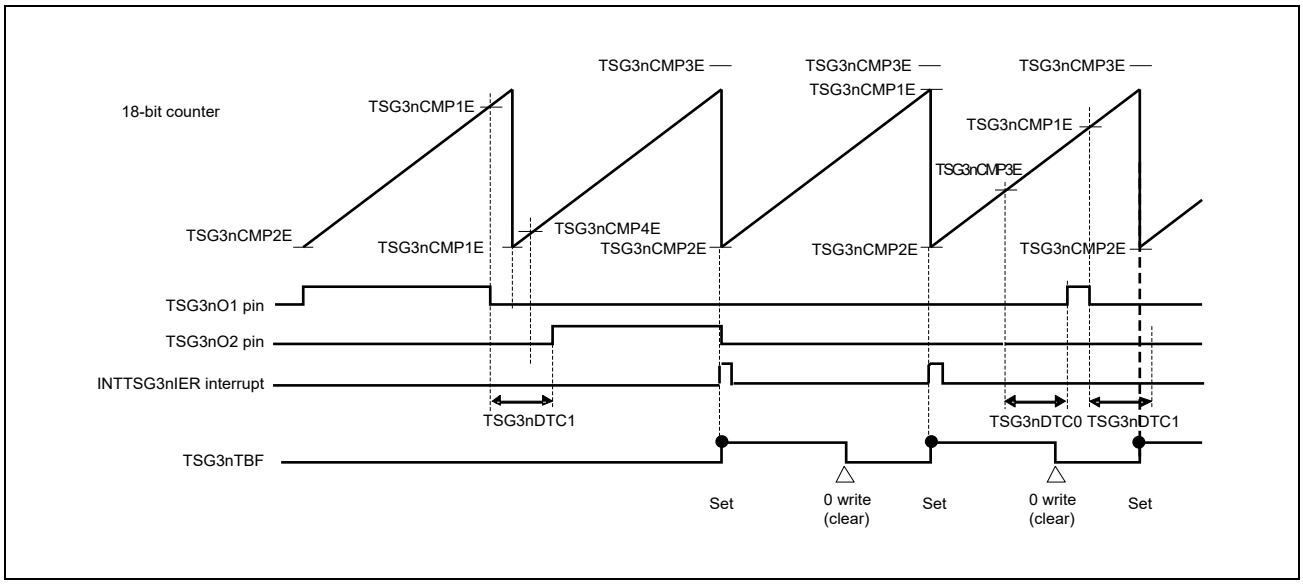


Figure 20.55 Example of 100% Duty Output at Dead Time Control

When the TSG3nO2 pin is set to duty cycle of 100% (TSG3nCMP3E ≥ TSG3nCMP0E + 1), the output of the TSG3nO1 pin is fixed to a low level. This control is intended to mask the active condition of TSG3nO1 output since the TSG3nO2 output is active before the TSG3nO1 output becomes active. In this case, the INTTSG3nIER interrupt is also generated because TSG3nO1 and TSG3nO2 outputs become high simultaneously.

NOTES

1. The TSG3nO1 and TSG3nO2 pins are set to active high.
2. The TSG3nO3 and TSG3nO4 pins and the TSG3nO5 and TSG3nO6 pins outputs behave similarly.

(6) Dead Time Rewriting during Timer Operation in PWM Mode

In PWM mode, it is possible to rewrite TSG3n dead time setting registers TSG3nDTC0W and TSG3nDTC1W while counting. The new settings are active at reload timing. It is not possible to change the dead time setting by rewriting at any time.

Please enable reloading by writing to the TSG3nCMP1E register.

20.4.7.2 HT-PWM mode (High Accuracy Triangular Pulse Width Modulation Mode)

Overview

In this mode, the 18-bit counter (up/down count by ± 2 bits, practically 17 bits) and the 18-bit compare registers (LSB is used to control additional pulse) are used to generate a 6-phase PWM signal.

Prerequisites

- Set the carrier wave period to TSG3nCMP0E.
- Set the duty cycle of the voltage data signals of the U phase, V phase, and W phase with TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE. (The values set to TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE are reflected immediately to the corresponding TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10)).
- Symmetric triangular wave control is described in this section. Please refer to **Section (10), Asymmetric Triangular Wave Control in HT-PWM Mode**, for asymmetric triangular wave control.

Functional description

Set the period of carrier wave and the duty cycle of the U phase, the V phase, and the W phase. Counting up begins when TSG3nTRG0.TSG3nTS is set to 1.

The 18-bit counter counts up from TSG3nDTC0 as the minimum value, and counts down upon the match of the maximum value of TSG3nCMP0E + TSG3nDTC0.

The dead time is set with TSG3nDTC0 and TSG3nDTC1. TSG3nDTC0 sets the dead time of inverse phase (OFF) to positive phase (ON) switch and TSG3nDTC1 sets the dead time of positive phase (OFF) to inverse phase (ON) switch. The 10-bit counters (TSG3nDTT1 to TSG3nDTT3) for dead time generation load the set values of TSG3nDTC0 and TSG3nDTC1 by the compare match of the 18-bit counter and the TSG3nCMPm buffer register (m = 1, 2, 5, 6, 9, 10), and start down-counting.

The INTTSG3nIm interrupts (m = 1, 2, 5, 6, 9, 10) are generated by the compare match of the 18-bit counter with the TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E buffer registers.

INTTSG3nIm interrupts (m = 3, 4, 7, 8, 11, 12) are generated by the compare match of the 18-bit counter with the TSG3nCMP3E, 7E, and 11E buffer registers when counting down (TSG3nCUF = 1), and with the TSG3nCMP4E, 8E, and 12E buffer registers when counting up (TSG3nCUF = 0).

NOTE

The HT-PWM mode is enabled when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 001_B.

(1) Block Diagram and Basic Timing Chart

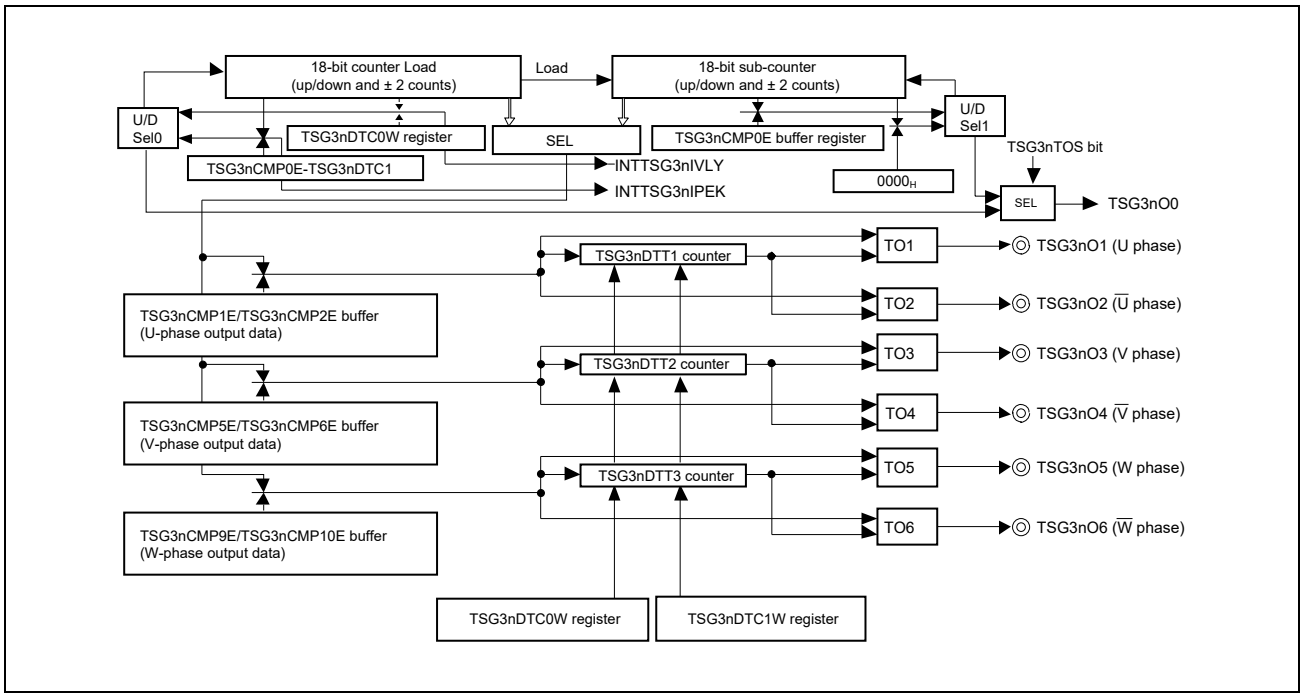


Figure 20.56 Block Diagram in HT-PWM Mode

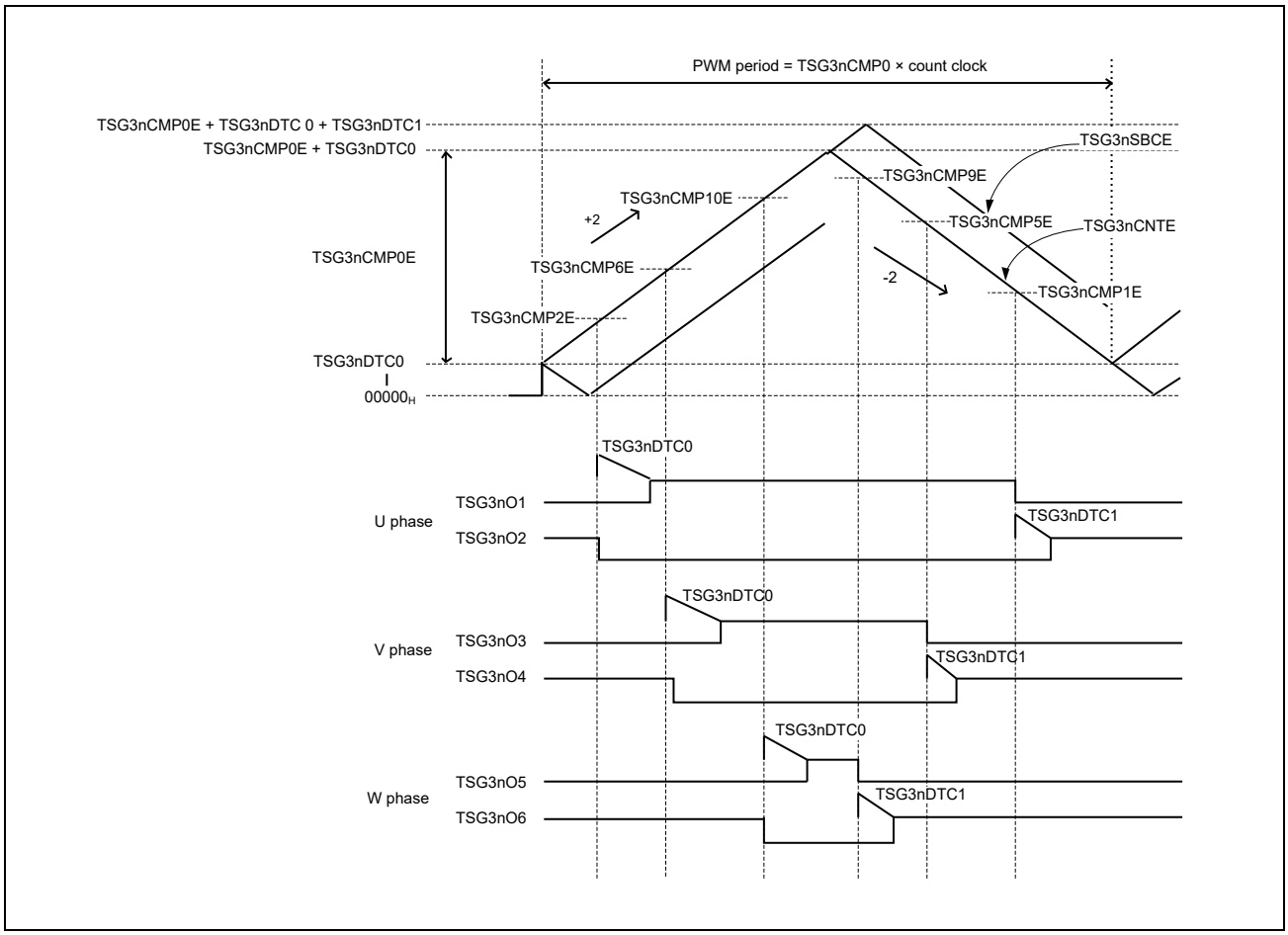
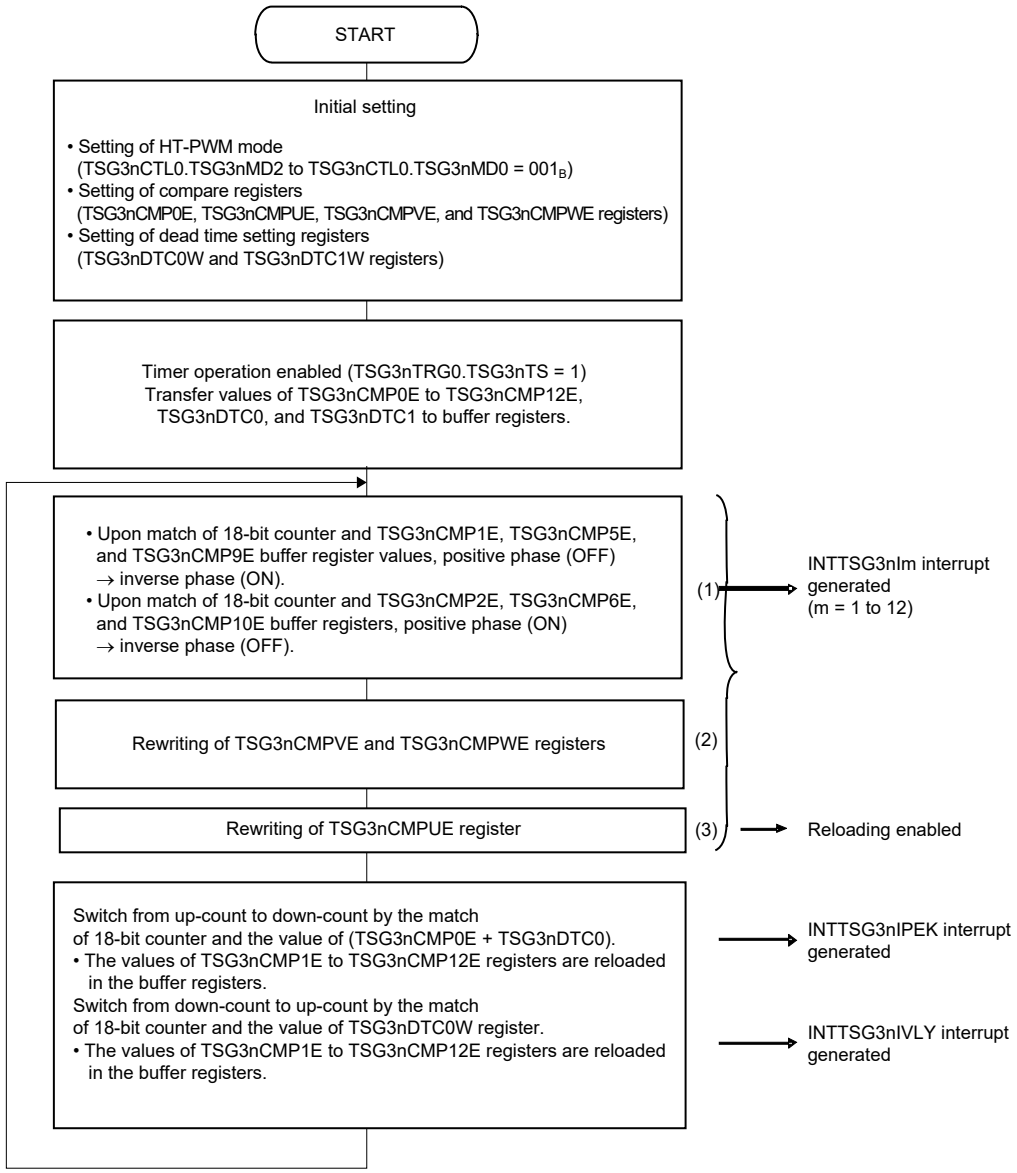


Figure 20.57 Basic Timing in HT-PWM Mode



Note: Write access to TSG3nCMPUE (TSG3nCMP1E) includes the operation to enable reloading. Therefore, (3) must be done after (2).

The INTTSG3nIPEK interrupt is generated only when TSG3nCTL4.TSG3nPRE = 1.

The INTTSG3nIVLY interrupt is generated only when TSG3nCTL4.TSG3nVRE = 1.

INTTSG3nI3, INTTSG3nI7, and INTTSG3nI11 outputs an interrupt at the match timing of TSG3nCMP3E, 7E, and 11E with TSG3nCNTE when counting down (TSG3nCUF = 1). INTTSG3nI4, INTTSG3nI8, and INTTSG3nI12 outputs an interrupt at the match timing of TSG3nCMP4E, 8E, and 12E with TSG3nCNTE when counting up (TSG3nCUF = 0).

Figure 20.58 Basic Operation Flow in HT-PWM Mode

(2) List of HT-PWM Mode Operations

Table 20.62 Counter Function in HT-PWM Mode

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger (counting up from TSG3nDTC0)
	Up count	Compare match of TSG3nDTC0 buffer register and 18-bit counter
	Down count	Compare match of TSG3nCMP0E + TSG3nDTC0 buffer register and 18-bit counter
	Clear	—
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1
18-bit sub-counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger (counting down from TSG3nDTC0)
	Up count	Underflow
	Down count	Compare match of TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 buffer register and 18-bit sub-counter
	Load	• TSG3nCMP0E + TSG3nDTC0: When value of 18-bit counter matches the value of buffer register TSG3nCMP0E + TSG3nDTC0 • TSG3nDTC0: When value of 18-bit counter matches the value of the buffer register TSG3nDTC0
	Clear	—
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 20.63 Compare Register and Dead Time Setting Register Functions in HT-PWM Mode

Register	Rewrite Method	Rewrite during Operation	Function
TSG3nCMP0E	Reload/Anytime rewrite	Possible	Setting period
TSG3nCMPUE	—	Possible	PWM control for U phase
TSG3nCMP1E, TSG3nCMP2E	Reload/Anytime rewrite		
TSG3nCMPVE	—	Possible	PWM control for V phase
TSG3nCMP5E, TSG3nCMP6E	Reload/Anytime rewrite		
TSG3nCMPWE	—	Possible	PWM control for W phase
TSG3nCMP9E, TSG3nCMP10E	Reload/Anytime rewrite		
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload/Anytime rewrite	Possible	Diagnostic signal output or A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Possible conditionally	Period and dead time setting

NOTE

- The rewritten values of TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE are set to TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E and TSG3nCMP10E.
- For rewriting method of TSG3nDTC0 and TSG3nDTC1, see **Section 20.4.7.2(8)(a), TSG3nDTC0 and TSG3nDTC1 Rewriting**.

Table 20.64 Timer Output Function in HT-PWM Mode

Pin	Function
TSG3nO0	Outputs an inactive level while the 18-bit counter or the 18-bit sub-counter counts up and an active level while the 18-bit counter or the 18-bit sub-counter counts down.
TSG3nO1	Outputs a PWM signal with dead time on a compare match of the TSG3nCMP1E buffer register with the 18-bit counter (counting down) or the TSG3nCMP2E buffer register with the 18-bit counter (counting up). Outputs a PWM signal on a compare match of the TSG3nCMP1E buffer register with the 18-bit sub-counter (counting down) or the TSG3nCMP2E buffer register with the 18-bit sub-counter (counting up) while TSG3nCMP1E < DTC0.
TSG3nO2	Outputs an inverse phase to TSG3nO1
TSG3nO3	Outputs a PWM with dead time on a compare match of the TSG3nCMP5E buffer register and the 18-bit counter (counting down) or the TSG3nCMP6E buffer register and the 18-bit counter (counting up). Outputs a PWM on a compare match of the TSG3nCMP3E buffer register and the 18-bit sub-counter (counting down) or the TSG3nCMP6E buffer register and the 18-bit sub-counter (counting up) while TSG3nCMP5E < DTC0.
TSG3nO4	Outputs an inverse phase to TSG3nO3
TSG3nO5	Outputs a PWM signal with dead time on a compare match of the TSG3nCMP9E buffer register and the 18-bit counter (counting down) and the TSG3nCMP10E buffer register and the18-bit counter (counting up). Outputs a PWM signal on a compare match of the TSG3nCMP5E buffer register and the 18-bit sub-counter (counting down) or the TSG3nCMP10E buffer register and the 18-bit sub-counter (counting up) while TSG3nCMP9E < DTC0
TSG3nO6	Outputs an inverse phase to TSG3nO5
TSG3nO7	Outputs a diagnostic signal or a pulse in response to an AD conversion trigger

NOTES

1. State of TSG3nO0 output can be switched by using the TSG3nIOC1.TSG3nTOS bit.
2. When the peak and trough values of the 18-bit sub-counter are set in TSG3nCMP1E and TSG3nCMP2E, clearing takes precedence.

Table 20.65 Interrupt Request in HT-PWM Mode

Interrupt	Function
INTTSG3nI0	Compare match of the TSG3nDTC0 buffer register with the 18-bit counter (periodic interrupt)
INTTSG3nIm (m = 1, 2, 5, 6, 9, 10)	Compare match of the TSG3nCMPmE buffer register with the 18-bit counter (m = 1, 2, 5, 6, 9, 10)
INTTSG3nIER	Error interrupt
INTTSG3nIVLY	Trough interrupt
INTTSG3nIPEK	Peak interrupt
INTTSG3nIWN	Warning interrupt

Table 20.66 Compare Match Timing in HT-PWM Mode

Compare Match	Timing
TSG3nCMP0E	When the 18-bit counter changes from TSG3nDTC0 to TSG3nDTC0 + 2
TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10)	When 18-bit counter changes from TSG3nCMPmE to TSG3nCMPmE ± 2 (m = 1, 2, 5, 6, 9, 10)

Table 20.67 Example of Setting Each Timer Output Condition in HT-PWM Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSG3nO0	Toggle output	TSG3nCMP0E × count clock	Outputs an inactive level when counting up, and an active level when counting down.	—
TSG3nO1, TSG3nO3, TSG3nO5	PWM output	TSG3nCMP0E × count clock	Outputs an inactive level throughout one period (0% duty)	TSG3nCMP0E ≤ TSG3nCMPmE ≤ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 (m = U, V, W)
			Outputs an active level of one count clock in one period	TSG3nCMPmE = TSG3nCMP0E – 1 (m = U, V, W)
			Outputs an inactive level of one count clock in one period	TSG3nCMPmE = 0001_H (m = U, V, W)
			Outputs an active level throughout one period (100% duty)	TSG3nCMPmE = 0000_H (m = U, V, W)
TSG3nO2, TSG3nO4, TSG3nO6	PWM output		Outputs an inactive level throughout one period (0% duty)	TSG3nCMPmE ≤ TSG3nDTC0 + TSG3nDTC1 (m = U, V, W)
			Outputs an active level of one count clock in one period	TSG3nCMPmE = TSG3nDTC0 + TSG3nDTC1 + 1 (m = U, V, W)
			Outputs an inactive level of one count clock in one period	TSG3nCMPmE = TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 – 1 (m = U, V, W)
			Outputs an active level throughout one period (100% duty)	TSG3nCMPmE = TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 (m = U, V, W)
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger	TSG3nCMP0E × count clock	See **Section 20.4.5, A/D Conversion Trigger Function**.	

(3) Settings of HT-PWM Mode

Mode setting

HT-PWM mode is entered by setting TSG3nCTL0.TSG3nMD2-TSG3nMD0 to 001_B.

Setting timer output

The output pins TSG3nO1 to TSG3nO6 are controlled by setting TSG3nIOC0, TSG3nIOC2, and TSG3nIOC3.

The output pin TSG3nO0 indicates the state of the 18-bit counter or the 18-bit sub-counter, whether counting up or down. The states between these counters are switched by using the TSG3nIOC1.TSG3nTOS bit.

The TSG3nO7 pin outputs pulses of the diagnostic output or the A/D conversion trigger. Please set it as required.

Enabling error interrupt generation

Error interrupt (INTTSG3nIER) due to the detection of the simultaneous active state of the positive phase and inverse phase is enabled by setting TSG3nIOC1.TSG3nEOC to 1. In HT-PWM mode, with any value set in the compare register, the simultaneous active state of the positive phase and inverse phase is not possible. For the detail, see **Section 20.4.6, Error/Warning Interrupt**.

Setting register rewriting timing with reload function

With TSG3nCTL3.TSG3nRMC, reload (simultaneous rewrite) or rewrite (anytime) is specified for the registers with reload function. The default setting is 0 (reload). To reload, set TSG3nCTL4.TSG3nPRE or TSG3nVRE to 1.

The reload timing is not generated if both the TSG3nPRE bits and TSG3nVRE bits are set to 0.

When “anytime rewrite” is specified, the unintended output may be generated depending on the rewrite timing.

Setting interrupts and skipping function

Interrupts and the skipping function are set with TSG3nCTL4. TSG3nPIE should be set to 1 when peak interrupt (INTTSG3nIPEK) is necessary and TSG3nVIE to 1 when trough interrupt (INTTSG3nIVLY) is necessary. To use the skipping function for peak/trough interrupts, the TSG3nRCC4 to TSG3nRCC0 must be set.

Setting A/D conversion trigger output

To set A/D conversion trigger 0 (TSG3nADTRG0 signal), use TSG3nCTL5.TSG3nAT09 to TSG3nAT00.

With TSG3nAT09 to TSG3nAT00, A/D conversion trigger output is enabled or disabled at the match of the 18-bit counter (during up count) with TSG3nDCMP2E to TSG3nDCMP0E, the match of the 18-bit counter (during down count) with TSG3nDCMP2E to TSG3nDCMP0E, the 18-bit counter peak interrupt (INTTSG3nIPEK), the 18-bit counter trough interrupt (INTTSG3nIVLY), the 18-bit sub-counter peak timing, and 18-bit sub-counter trough timing.

To set A/D conversion trigger 1 (TSG3nADTRG1 signal), use TSG3nCTL6.TSG3nAT19 to TSG3nAT10.

To set the match timing of 18-bit counter and TSG3nDCMP2E to TSG3nDCMP0E, set the compare value to the pertinent register.

The skipping function can be used for TSG3nADTRG0 and the TSG3nADTRG1 signals. Use TSG3nACC01, TSG3nACC00 of TSG3nCTL5, TSG3nACC10, and TSG3nACC11 of TSG3nCTL6 to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTION

Set TSG3nCTL5, TSG3nCTL6, and TSG3nDCMP2E-TSG3nDCMP0E correctly when using the TSG3nO7 output for the A/D conversion trigger timing pulse.

Setting dead time

The dead time can be set with TSG3nDTC0 and TSG3nDTC1.

The dead time is calculated by the following expressions:

- PCLK × TSG3nDTC0
- PCLK × TSG3nDTC1

TSG3nDTC0 can set the time between a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the inactive state and a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the active state, respectively.

TSG3nDTC1 can set the time between a change of TSG3nO1, TSG3nO3, TSG3nO5 to the inactive state and a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the active state, respectively.

TSG3nDTC0 and TSG3nDTC1 can only be set to an even value.

Carrier period

Set the carrier period with TSG3nCMP0E according to the following expression:

TSG3nCMP0E = Carrier period/count clock period (PCLK)

Satisfy the following requirements when setting the TSG3nCMP0E register regarding the dead time:

- TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 ≤ $3FFFE_H$
- TSG3nCMP0E > TSG3nDTC0
- TSG3nCMP0E > TSG3nDTC1
- TSG3nCMP0E > 3 × MAX (TSG3nDTC0, TSG3nDTC1)
- TSG3nCMP0E: Even number

NOTE

MAX (A, B) indicates the larger value of A and B.

Duty (PWM width) setting

The duty of the U phase, the V phase, and the W phase is set with TSG3nCMPmE (m = U, V, W, or 1, 2, 5, 6, 9, and 10), respectively. The setting range of the compare registers is as follows:

00000_H ≤ TSG3nCMPmE ≤ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1

LSB (least significant bit) of TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE indicates the setting of an additional pulse. When TSG3nCMPUE = 00003_H, the change in the inverse phase (TSG3nO2 output) is done one count clock later compared to the TSG3nCMPUE = 00002_H setting (when the 18-bit counter is up-counting). The additional pulse cannot be set to TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, or TSG3nCMP10E (only even numbers can be set to these registers).

(4) H18-bit counter Operation in HT-PWM Mode

The 18-bit counter is initialized to 00000_H and the value of TSG3nDTC0 is loaded immediately after starting the TSG3n timer operation (TSG3nTRG0.TSG3nTS = 1). Afterwards, counting is done by +2. After 18-bit counter reaches the value of TSG3nCMP0E + TSG3nDTC0, counting is done by -2.

The following figure shows 18-bit counter operation.

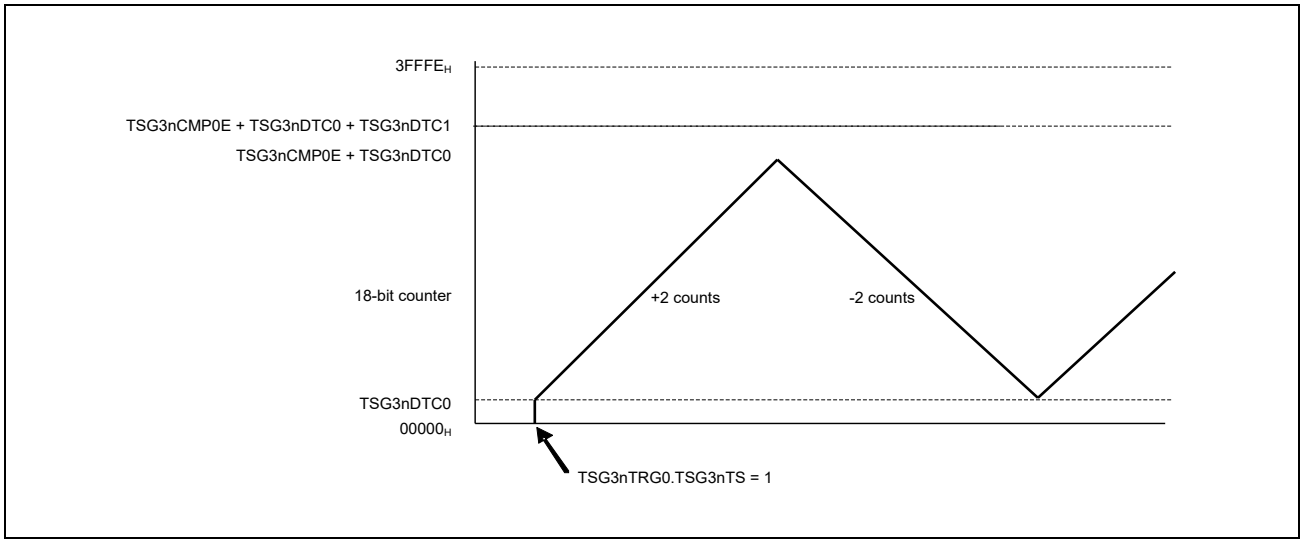


Figure 20.59 Example of 18-bit counter Operation in HT-PWM Mode

NOTE

Minimum 18-bit counter value: TSG3nDTC0
Maximum 18-bit counter value: TSG3nCMP0E + TSG3nDTC0
Carrier period: TSG3nCMP0E × count clock period (PCLK)

The 18-bit sub-counter is initialized to 00000_H and the value of TSG3nDTC0 is loaded immediately after starting the TSG3n timer operation (TSG3nTRG0.TSG3nTS = 1). Afterwards, counting by -2 is done until 00000_H is reached and counting by +2 begins. Next, the value of the 18-bit counter is loaded into the 18-bit sub-counter at a change timing of the 18-bit counter into the down count. Counting up by the 18-bit sub-counter continues until the value reaches the value of TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1, and then counting by -2 begins. Similarly, when the 18-bit counter value matches the TSG3nDTC0 value, the 18-bit counter value is loaded to the 18-bit sub-counter and the down count is continued.

The following figure shows the 18-bit sub-counter operation.

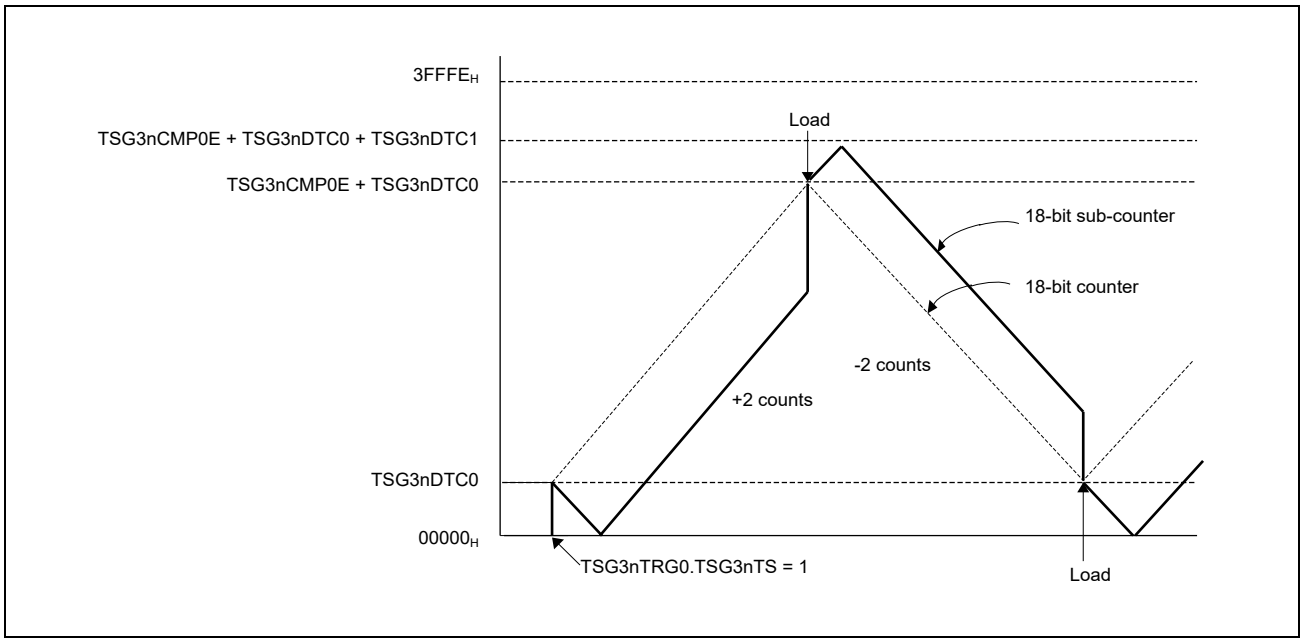


Figure 20.60 Example of 18-bit Sub-Counter Operation in HT-PWM Mode

NOTE

Minimum 18-bit sub-counter value: 00000_H
Maximum 18-bit sub-counter value: TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1

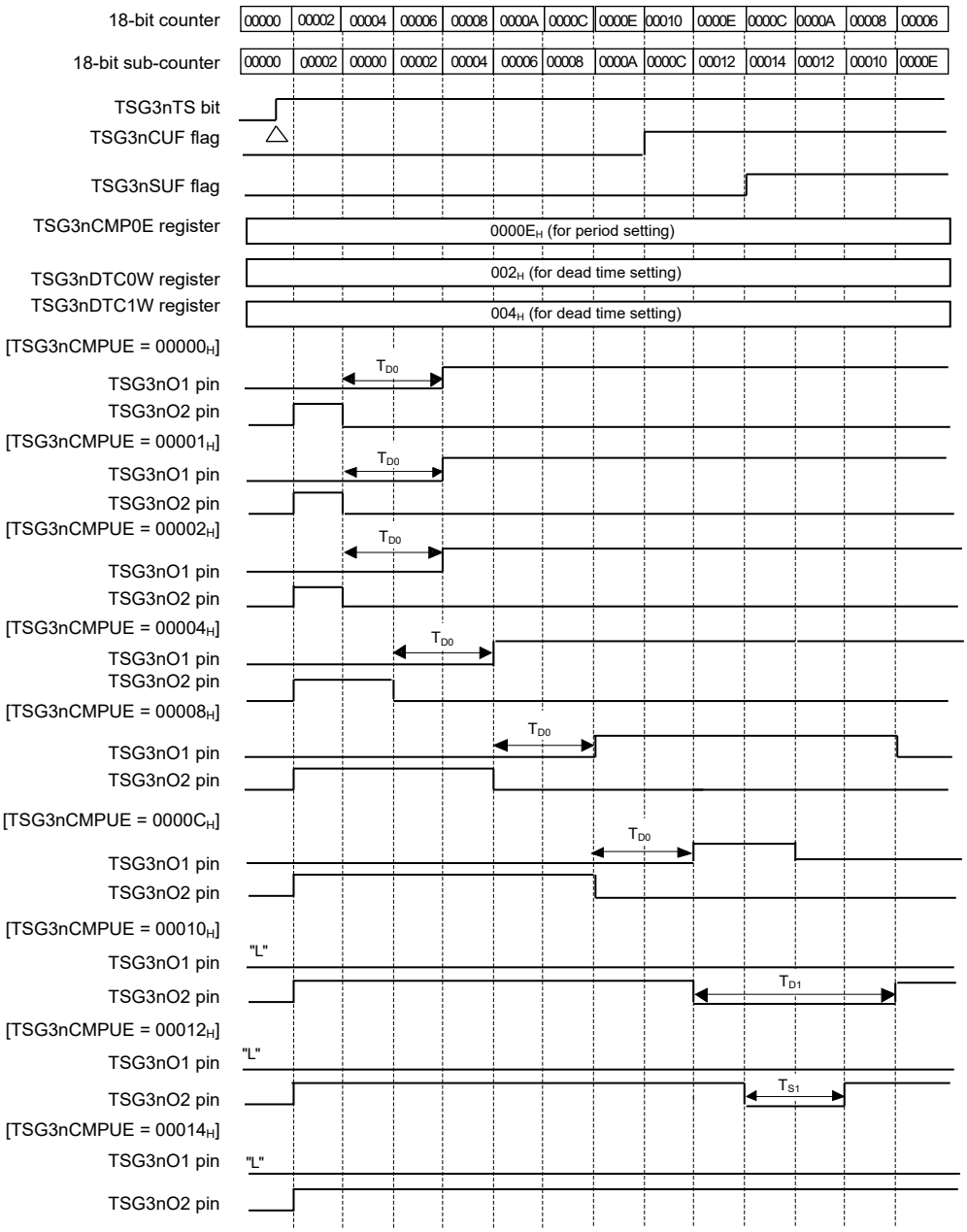
(5) Basic Operation of HT-PWM Mode

(a) Example of Timer Output Immediately after the Start of the TSG3n Timer Operation

The following figure shows the timing chart when TSG3nCMP0E = $0000E_H$, TSG3nDTC0 = 002_H, TSG3nDTC1 = 004_H and TSG3nCMPUE = 00000_H to 00014_H (excerpt). In this example, TSG3nIOC2.TSG3nOL1 to TSG3nOL6 = 000000_B.

When operation starts (TSG3nTRG0.TSG3nTS = 1), the level of the TSG3nO2 pin changes to active. Afterwards, if TSG3nCMPUE ≤ TSG3nDTC0, the TSG3nO2 pin is cleared after 1 count clock cycle.

The TSG3nO2 pin is cleared upon a match of the 18-bit counter and the compare register (TSG3nCMP2E), or a match of the 18-bit sub-counter and the compare register (TSG3nCMP2E) if TSG3nCMPUE ≥ TSG3nDTC0. Afterwards, the TSG3nO1 pin is set after the set dead time period (the TSG3nO1 pin is not set if TSG3nCMPUE ≥ TSG3nCMP0E).



Note: Δ: Write access

Figure 20.61 Example of Timer Output when TSG3nTS is Set to 1 (Initial State) in HT-PWM Mode

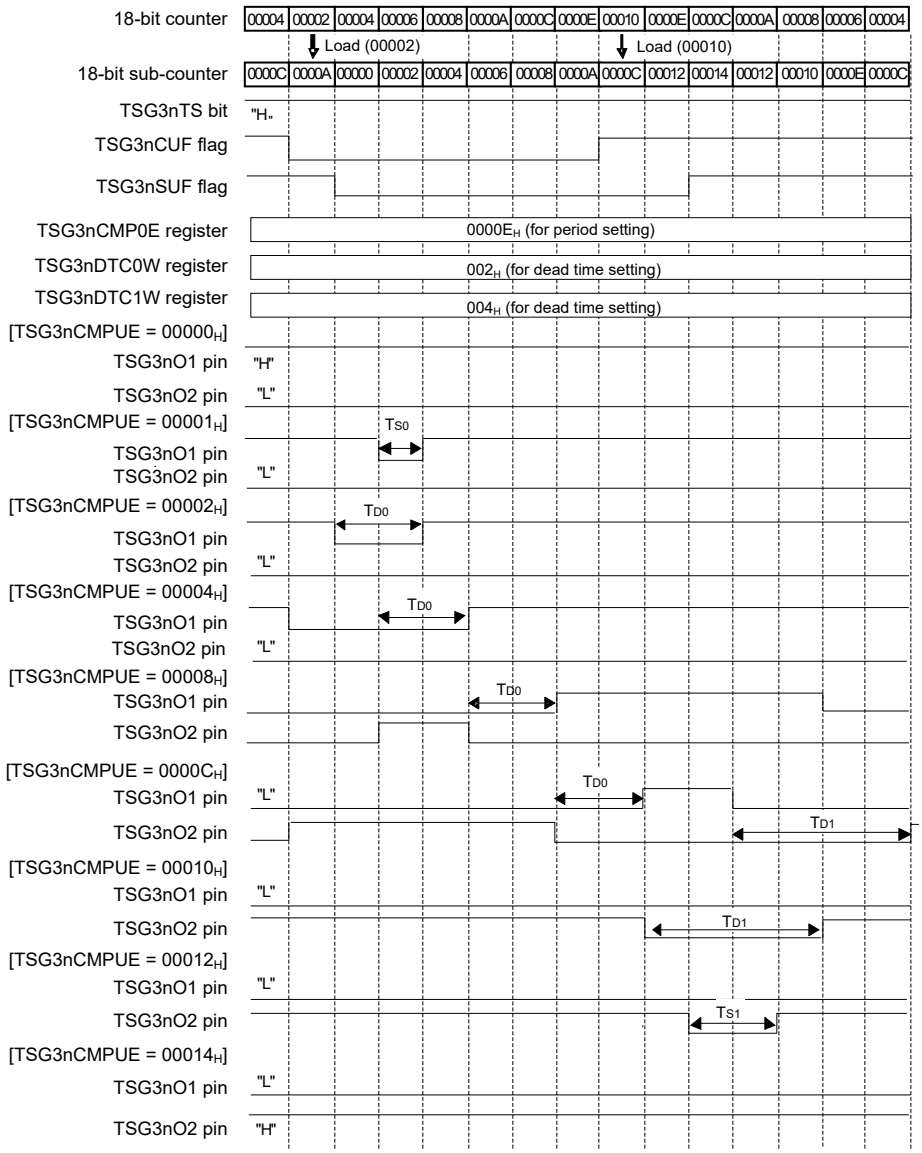
NOTES

1. TSG3nCMP0E = $0000E_H$, TSG3nDTC0 = 002_H, TSG3nDTC1 = 004_H
2. T_{D0}: Time depending on setting of the dead time in the TSG3nDTC0W register
   T_{D1}: Time depending on setting of the dead time in the TSG3nDTC1W register
   T_{S1}: Time decided by compare match of the 18-bit sub-counter and TSG3nCMPUE register, when TSG3nCMPUE >18-bit counter maximum value

(b) Example of Timer Output during TSG3n Timer Operation

The following figure shows the timing chart when TSG3nCMP0E = $0000E_H$, TSG3nDTC0 = 002_H, TSG3nDTC1 = 004_H, and TSG3nCMPUE is set to 00000_H-00014_H (excerpt). In this example, TSG3nIOC2.TSG3nOL1-TSG3nOL6 = 000000_B.

The range of the active (high level) width of a positive phase (TSG3nO1) output is $00000_H \leq$ TSG3nCMPUE $\leq$ TSG3nCMP0E (for the additional pulse). The range of the active (high level) width of an inverse phase (TSG3nO2) output is TSG3nDTC0 + TSG3nDTC1 $\leq$ TSG3nCMPUE $\leq$ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1.



Note 1. TSG3nCMP0E = $0000E_H$, TSG3nDTC0 = 002_H, TSG3nDTC1 = 004_H

Note 2. T_{D0}: Time depending on setting of the dead time in the TSG3nDTC0 register
T_{D1}: Time depending on setting of the dead time in the TSG3nDTC1 register
T_{S0}: Time decided by compare match of 18-bit sub-counter and the TSG3nCMPUE register, when TSG3nCMPUE < 18-bit counter minimum value
T_{S1}: Time decided by compare match of the 18-bit sub-counter and the TSG3nCMPUE register, when TSG3nCMPUE > 18-bit counter maximum value

Figure 20.62 Example of Timer Output during TSG3n Operation in HT-PWM Mode

(6) Additional Pulse Control in HT-PWM Mode

The HT-PWM mode can generate an additional pulse by setting 1 to the LSB of the duty setting registers (TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE). This allows more precise control of the pulse duty than standard pulse control.

The following sections describe two examples of pulse output of TSG3nO1: additional pulse control is used in one example and additional pulse control is not used in another.

(a) Example of Pulse Output when Additional Pulse Control Is Used

Figure 20.63 shows the additional pulse control when an odd value is set to TSG3nCMPUE.

The arrows and numerical values show the width of the duty cycle of the TSG3nO1 output in one period.

When the additional pulse control is used as shown in **Figure 20.63**, the width of the output of the TSG3nO1 (duty cycle) can be set within a range from the width of 12 clock cycles to the width of 0 clock cycles in one-clock-cycle step.

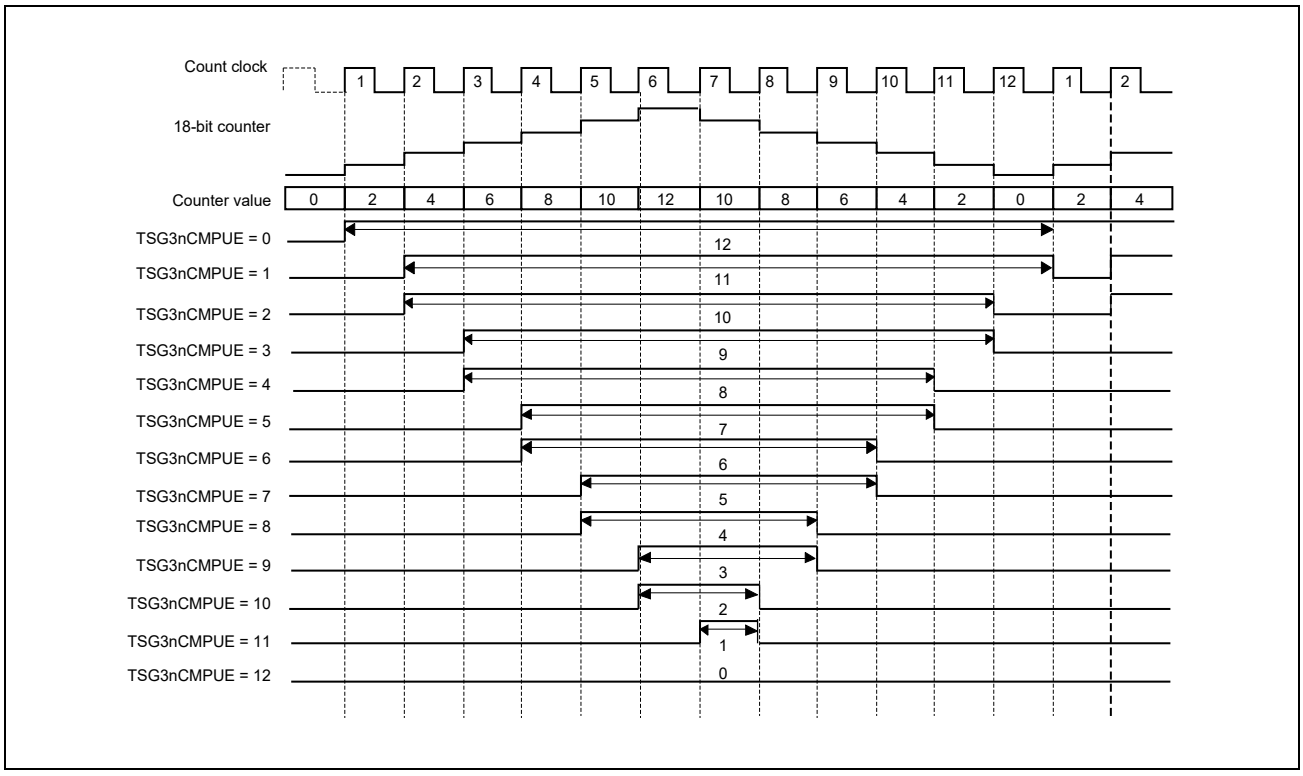


Figure 20.63 Example of TSG3nO1 Output when Additional Pulse Control Is Used in HT-PWM Mode

NOTE

TSG3nCMP0E = 12, TSG3nDTC0 = 0, TSG3nDTC1 = 0

(b) Example of Pulse Output when Additional Pulse Control Is Not Used

The arrows and numerical values in **Figure 20.64** show the width of the duty cycle of the TSG3nO1 output in one period.

When the additional pulse control is not used, the width of the TSG3nO1 output can be set within a range from the width of 12 clock cycles to the width of 0 clock cycles in two-clock-cycle step. In this case, the change in duty cycle is larger than that in the case when the additional pulse control is used.

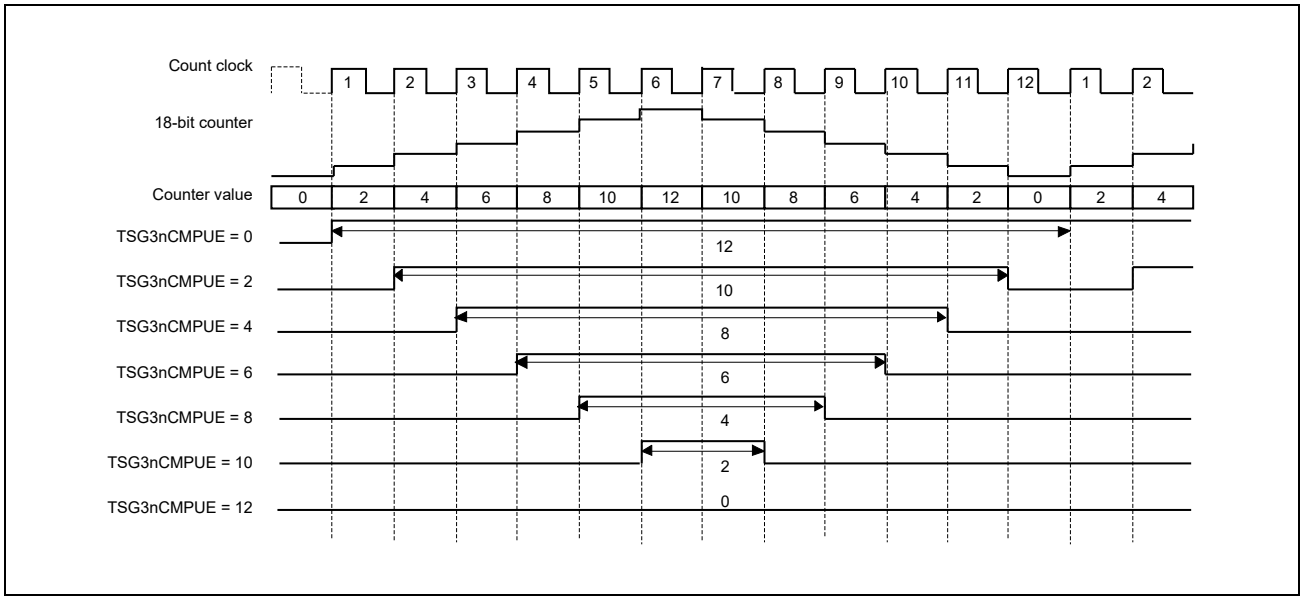


Figure 20.64 Example of Output when Additional Pulse Control is Not Used in HT-PWM Mode

NOTE

TSG3nCMP0E = 12, TSG3nDTC0 = 0, TSG3nDTC1 = 0

(7) Dead Time Control in HT-PWM Mode

Duty setting registers are TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE in HT-PWM mode. The 6-phase PWM waveform of a variable duty is output by using these registers. To achieve the dead time control, there are two dead time setting registers (TSG3nDTC0W and TSG3nDTC1W) and six 10-bit down counters that operate synchronously with the count clock of the 18-bit counter. TSG3nDTC0 is used for setting a dead time from a change of the inverse phase to the inactive state to a change of the positive phase to the active state. TSG3nDTC1 is used for setting a dead time from a change of the positive phase to the inactive state to a change of the inverse phase to the active state.

The following figure shows the output waveform when TSG3nDTC0 = x and TSG3nDTC1 = y.

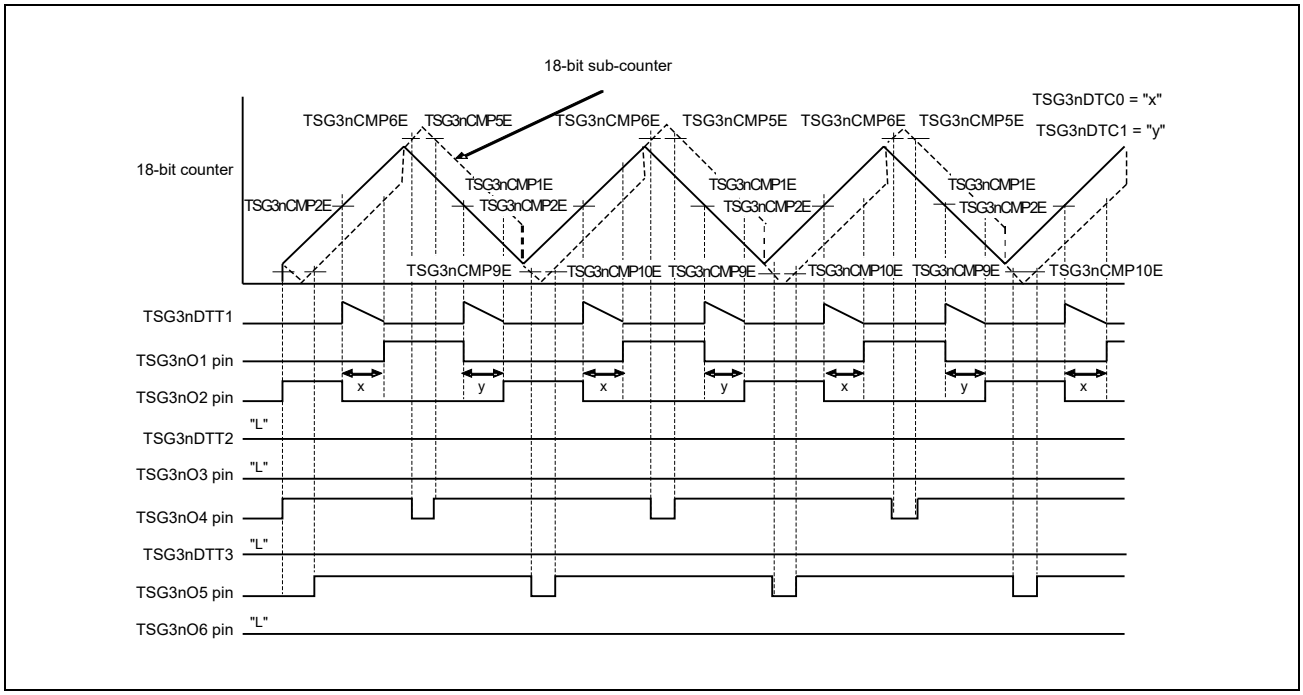


Figure 20.65 Example of Output Waveform with Dead Time in HT-PWM Mode

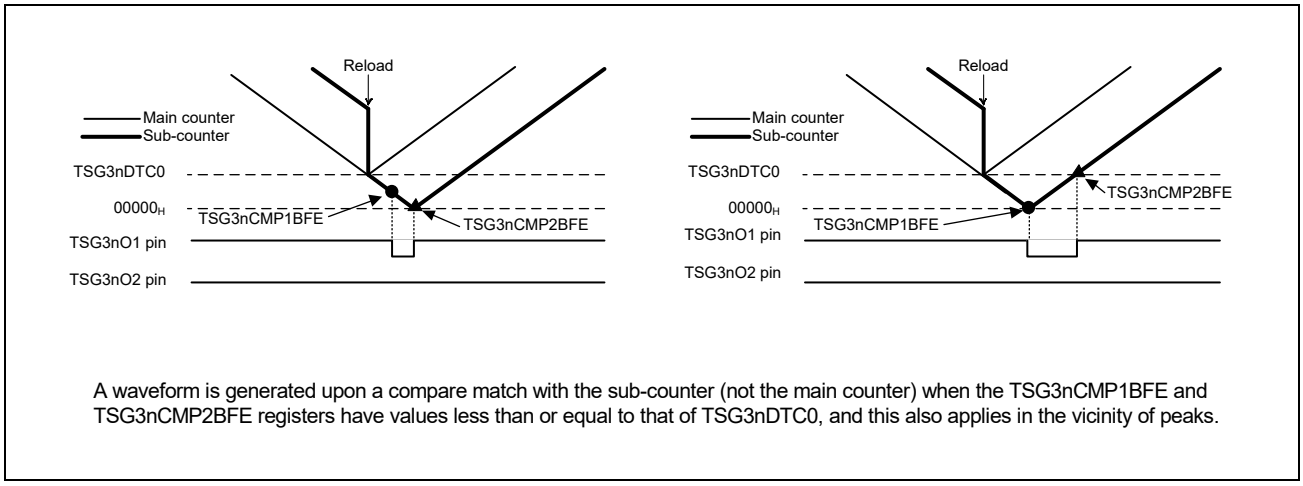


Figure 20.66 Example of Output Waveform near Trough after Reloading

(8) Notes Concerning Dead Time Control in HT-PWM Mode

(a) TSG3nDTC0 and TSG3nDTC1 Rewriting

It is possible to rewrite the dead time setting in TSG3nDTC0 and TSG3nDTC1 registers during timer operation.

CAUTIONS

1. Rewrite TSG3nDTC0 and TSG3nDTC1 when the reload function is used (TSG3nRMC = 0).
2. The write protection code check function is applied when TSG3nDTC0 and TSG3nDTC1 are rewritten. For the detail, see the pertinent register descriptions (**Section 20.3.44, 20.3.45, 20.3.64).**
3. When the TSG3nCMP0E and TSg3nDTC1 are updated at the peak of the 18-bit counter:
   When the set value of TSG3nCMPmE is greater than the updated TSG3nCMP0E + TSG3nDTC0 (new maximum value of the main counter), the match interrupt (INTTSG3nIm) is not generated immediately after reloading (m = 2, 6, or 10).

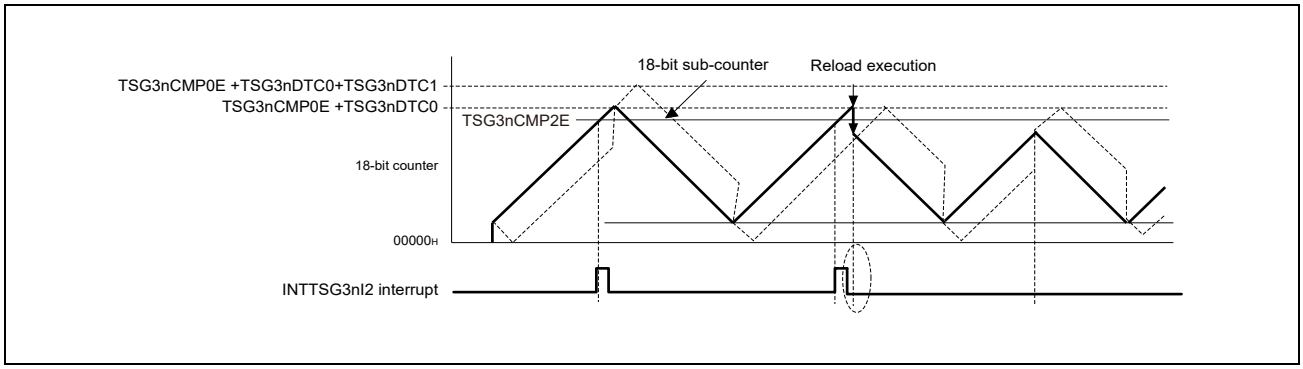


Figure 20.67 Example of Operation During Reloading at 18-Bit Counter Peak Timing

4. When the TSG3nDTC0 is updated at the trough of the 18-bit counter:
   When the TSG3nCMPmE set value is smaller than the updated TSG3nDTC0 (new minimum value of the main counter), the match interrupt (INTTSG3nlm) is not generated immediately after reloading (m = 1, 5, or 9).

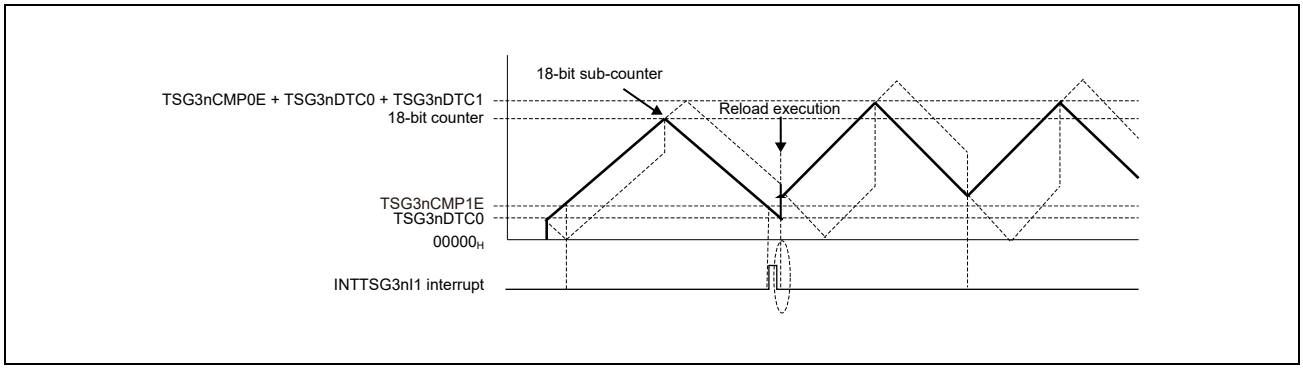


Figure 20.68 Example of Operation During Reloading at 18-Bit Counter Trough Timing

(9) Software Output Control Function in HT-PWM Mode

TSG3nOPT0.TSG3nSOC, TSG3nIDC, and TSG3nOPT1.TSG3nSPC2-TSG3nSPC0 are used in HT-PWM mode for software control of timer output control.

As shown in **Figure 20.69**, with TSG3nSTE = 0, the output control is switched immediately when TSG3nSOC is set to 1. If the dead time is set, the period of the dead time is guaranteed. After that, when TSG3nSOC is set to 0, output control is retained. When the reload timing is generated, output control is switched to HT-PWM mode output control.

For details, refer to **Section 20.4.7.10, Software Output Control Function**.

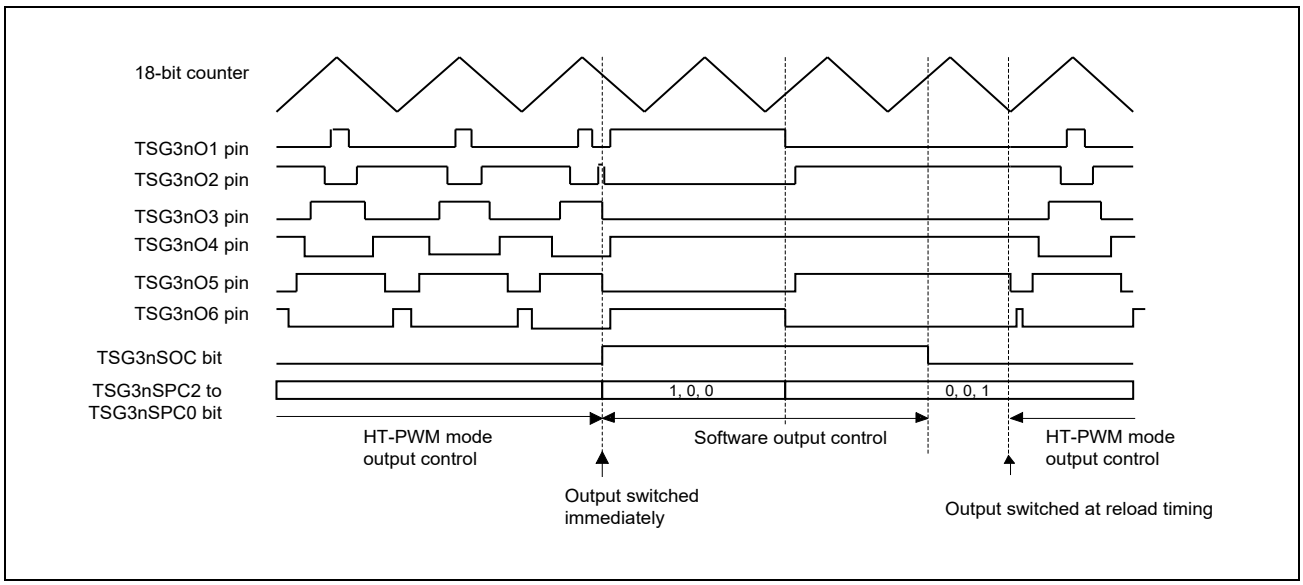


Figure 20.69 Example of Software Output Control Switching in HT-PWM Mode

CAUTION

Use reload (simultaneous rewrite) mode (TSG3nCTL3.TSG3nRMC = 0) when software output control function is used.

(a) Procedure for Software Output Control

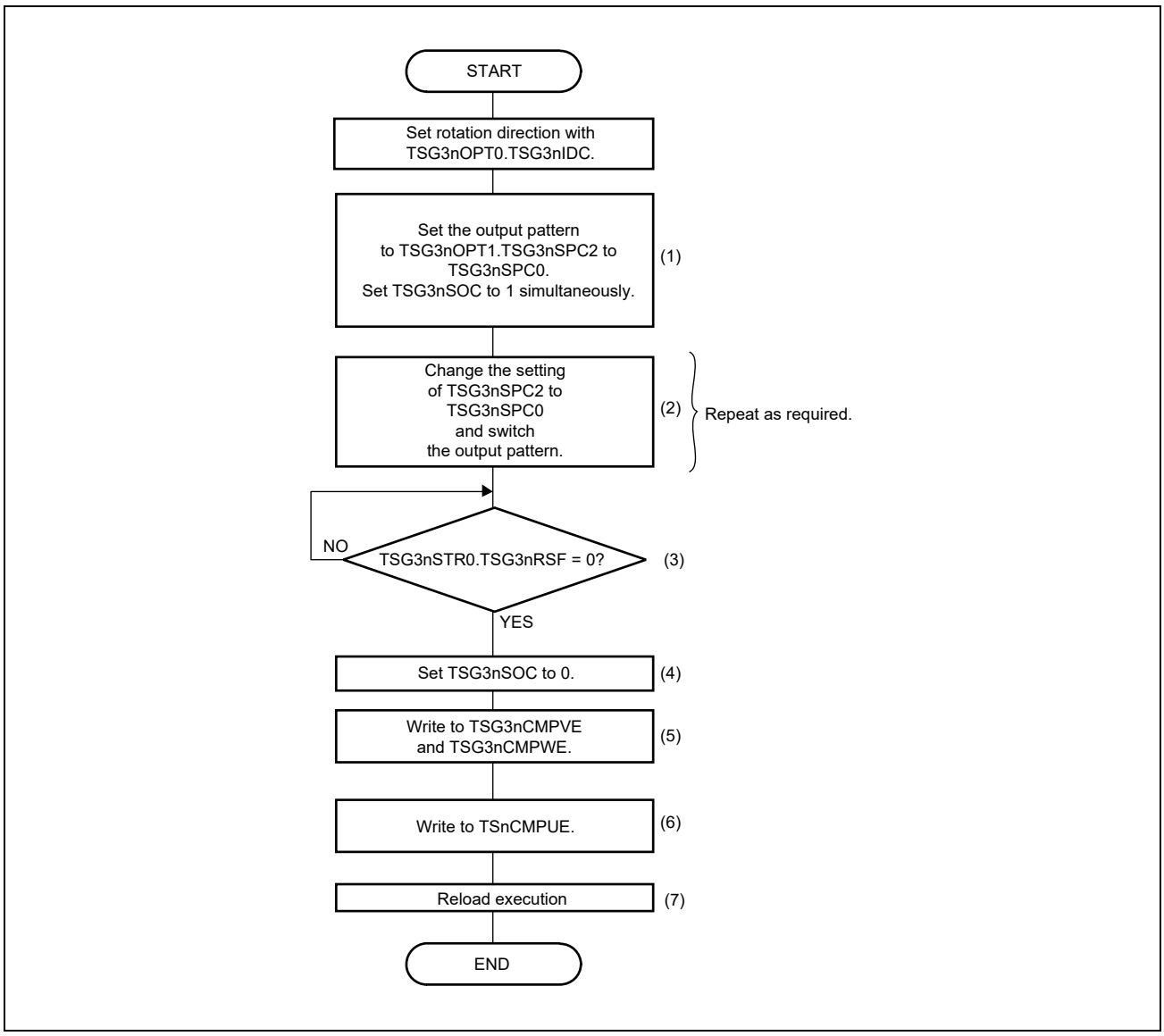


Figure 20.70 Flow of Software Output Control

The procedure for software output control is described below.

(1) Set the output pattern to the TSG3nOPT1.TSG3nSPC2-TSG3nSPC0. To enable software output control, set TSG3nOPT0.TSG3nSOC to 1 simultaneously.

(2) Change the output pattern setting of the TSG3nSPC2-TSG3nSPC0 to change the timer output.

(3) Confirm that reload request flag TSG3nSTR0.TSG3nRSF = 0. In case TSG3nRSF = 1, do not proceed to the following step until TSG3nRSF = 0.

(4) Setting TSG3nSOC to 0 starts releasing of the software control (it is not released here yet).

(5) Make necessary settings of the compare registers that will be used after the software output control is released. Proceed to the following step when the register settings are not required. Here, change the registers with the reload function.

(6) Write to TSG3nCMPUE (TSG3nCMP1E) to start reloading.

(7) Reload is executed and software output is released.

CAUTION

Execute reload after executing steps (3), (4), (5), and (6). When reload cannot be executed, the software output cannot be released.

(10) Asymmetric Triangular Wave Control in HT-PWM Mode

In HT-PWM mode, it is possible to control output by an asymmetric triangular waveform by setting the different timings for setting and clearing the U, V, and W phases.

The following describes the differences of the asymmetric triangular wave control from the symmetric triangular wave control.

(a) PWM Setting

When a symmetric triangular wave is used, the output control of each of the U phase, V phase, and W phase is done by setting the set timing and the clear timing to the same value to the TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE. When an asymmetric triangular wave is used, the output control of each phase is done by setting TSG3nCMPmE as follows (m = 1, 2, 5, 6, 9, 10).

Prerequisites

- The clear timing of PWM of the voltage data signal of U, V and W phases is set with TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E.
- The set timing of PWM of the voltage data signal of U, V, and W phases is set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E.
- The set and clear timings of each phase can also be set with TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP5E, TSG3nCMP9E, and TSG3nCMP10E.
- TSG3nCMPmE can only be set to an even value (m = 1, 2, 5, 6, 9, 10).

(b) Timer Output

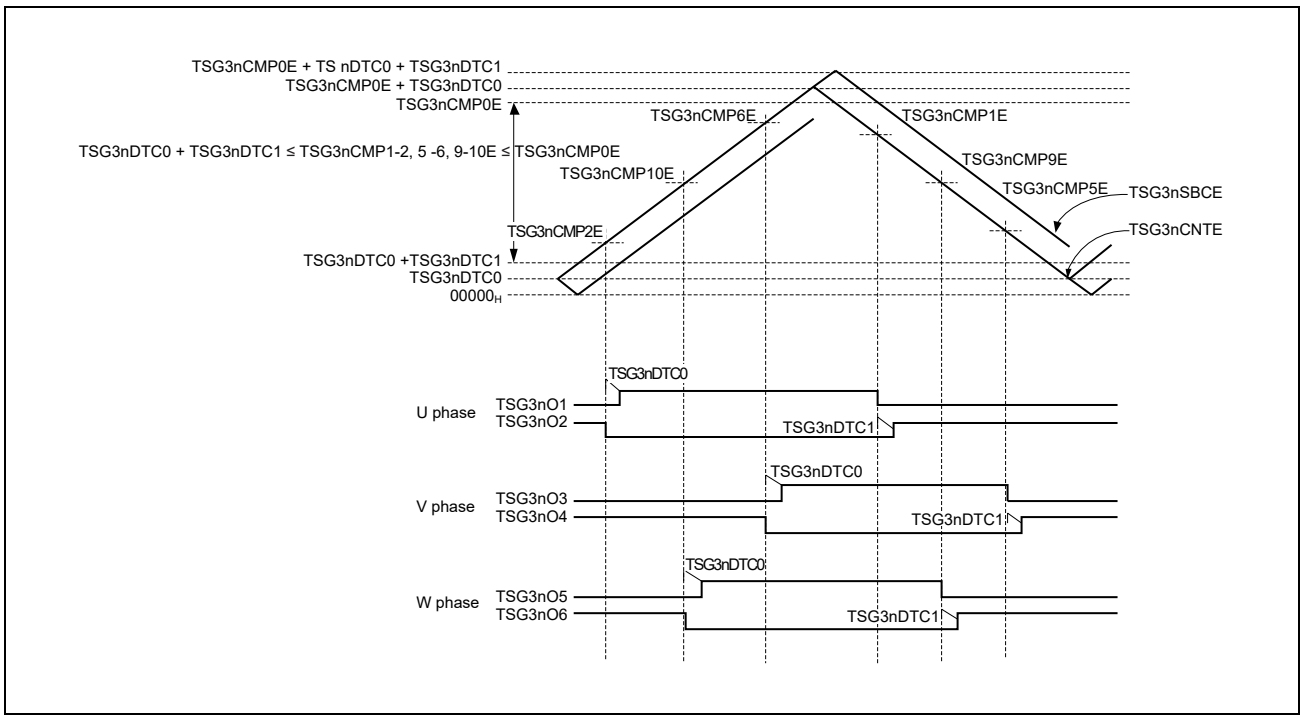


Figure 20.71 Example of Timer Output Waveform in HT-PWM Mode

NOTE

When output is controlled by the asymmetric triangular waveform, the following conditions apply to setting of TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10).

- TSG3nDTC0+TSG3nDTC1 ≤ TSG3nCMPmE ≤ TSG3nCMP0E
- Only when TSG3nCMPmE = TSG3nCMP (m + 1) E, or TSG3nCMPmE = TSG3nCMP (m+ 1) E + 2, it is possible to set TSG3nCMPmE under the condition 00000_H ≤ TSG3nCMPmE ≤ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1, which also applies to the case where the symmetric triangular wave is used.

20.4.7.3 Data Transfer from EMU3

TSG3 can directly reflect the values of carrier cycle set in EMU3 and the U, V, and W phase duty calculated by EMU3 to the compare registers TSG3nCMP0E, 1E, 2E, 5E, 6E, 9E, and 10E.

When the carrier cycle value EMU3nCARR from EMU3 and the compare values EMU3nPWMUIP, EMU3nPWMVIP, and EMU3nPWMWIP of U, V, and W phases are input to TSG3 and the EMU3n register write signal is set to 1, the TSG3 compare registers TSG3nCMP0E, 1E, 2E, 5E, 6E, 9E, and 10E are updated.

When the calculation of the PWM IP in EMU3n is completed, the EMU3n register write signal is set to 1. In addition, the signal can also be controlled by software.

For details, see **Section 25.4.5, PWM IP** in **Section 25, Enhanced Motor Control Unit 3 (EMU3)**.

Note that TSG32 does not support synchronous operation with EMU3.

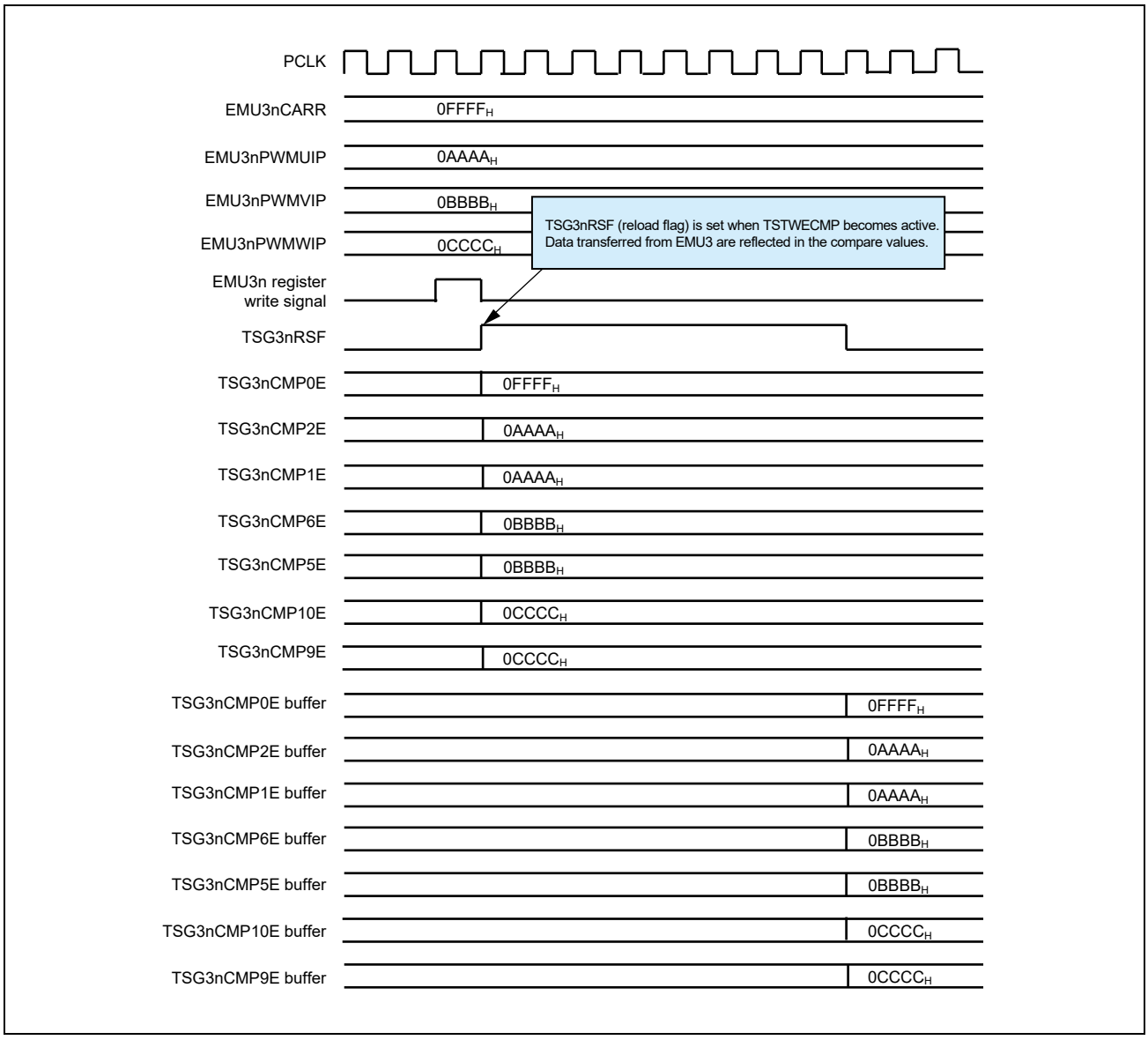
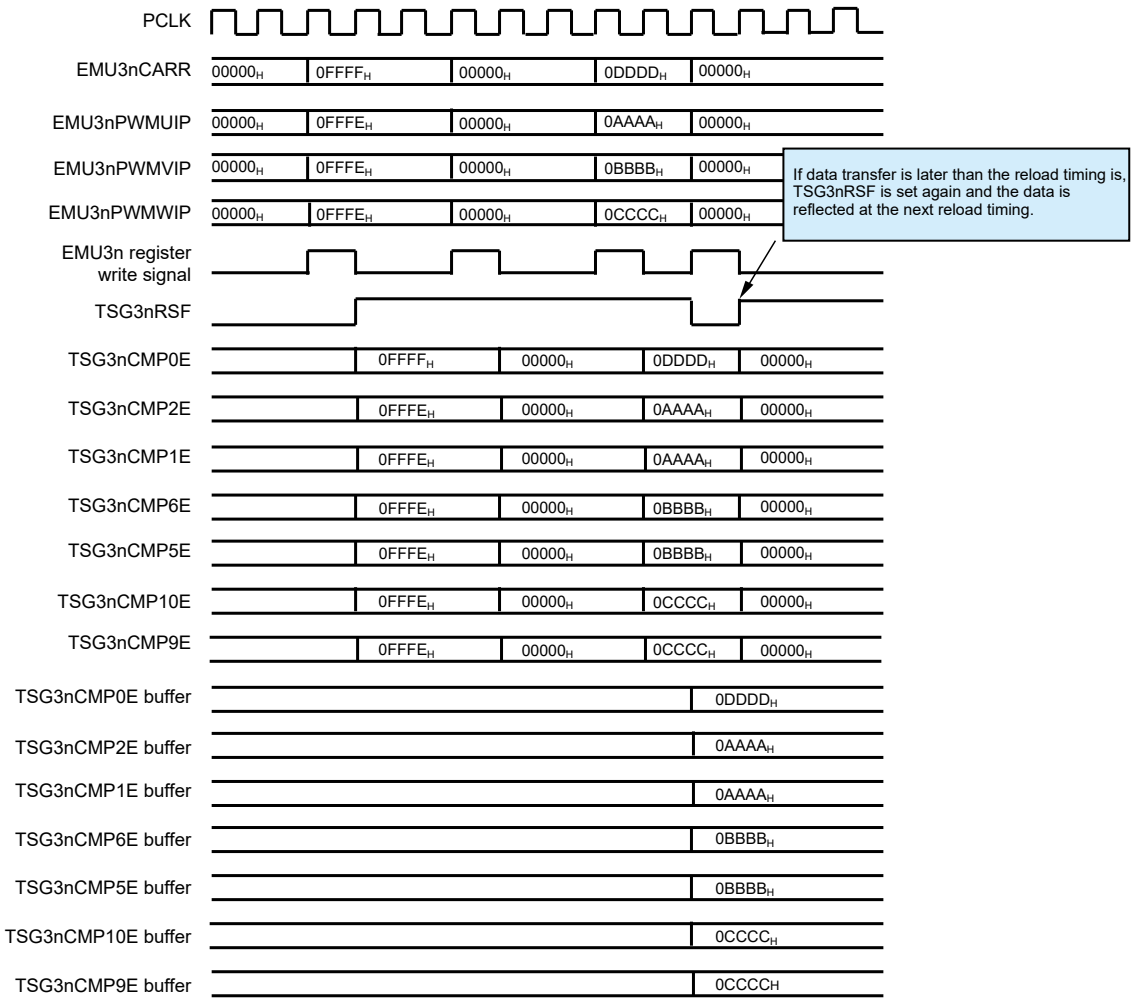


Figure 20.72 Transfer Timing of Data from EMU3

CAUTION

Transfer from EMU3 is enabled only in HT-PWM mode (TSG3nMD2-0 = 001) and in reload mode operation (TSG3nRMC = 0). Do not transfer data in PWM mode, SP-PWM mode, 120-DC mode, HSP-PWM mode or in anytime rewrite mode (TSG3nRMC = 1).

If transfers of data from EMU3 are performed sequentially before a reload timing is generated, the last transferred data is effective.



Note 1. If a transfer of data from EMU3 occurs at the same time as reload timing, TSG3nRSF is not cleared and the transferred data conflicted is reflected at the next reload timing.

Note 2. Transfer of data from EMU3 occurs at the same time as a write access to a register, the write access is effective. For example, in the case of writing to TSG3nCMP0E and transfer of data from EMU3, writing makes effects to the TSG3nCMP0E register and transferred data is reflected in the other registers TSG3nCMP1E, 2E, 5E, 6E, 9E, and 10E.

Figure 20.73 Transfer Timing of Data from EMU3

20.4.7.4 ESW Function

ESW function outputs PWM which is not generated using the counter/compare value of TSG3 but adding the set dead time to the rectangular waveform from EMU3 to TSG3.

CAUTION TSG32 does not support the ESW function.

(1) PWM Output Using ESW Function

Setting 1 to the TSG3nOPT2.TSG3nESSC bit enables the ESW function. PWM output is switched from the one using TSG3 counter/compare values to the one derived from the rectangular waveform input from EMU3.

With the ESW function, the U-phase output pattern value, V-phase output pattern value, and W-phase output pattern value input from EMU3 are positive output (TSG3nO1, 3, and 5), and the inverse of the U-phase output pattern value, V-phase output pattern value, and W-phase output pattern value signals are inverse output (TSG3nO2, 4, and 6).

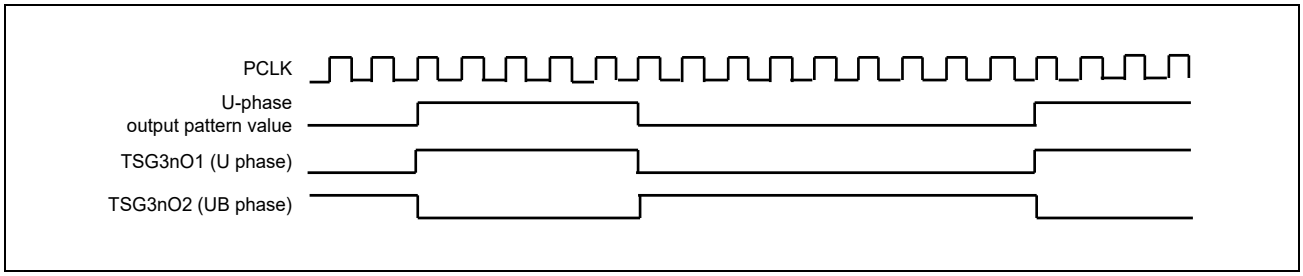


Figure 20.74 PWM Output Using ESW Function (With 0 Dead Time)

The setting of dead time is also effective in ESW function. The dead time set to TSG3nDTC0 is inserted to the positive phase (TSG3nO1, 3, and 5), and the dead time set to TSG3nDTC1 is inserted to the negative phase (TSG3nO2, 4, and 6).

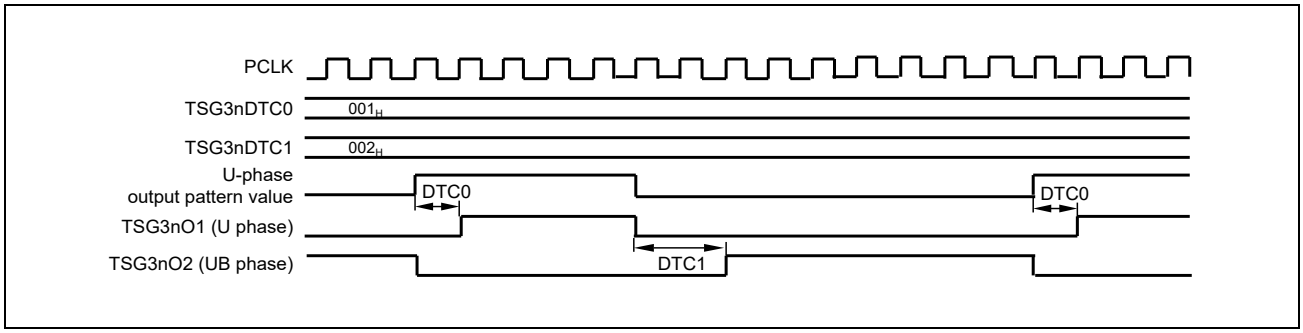


Figure 20.75 PWM Output Using ESW Function (TSG3nDTC0 = 1, TSG3nDTC1 = 2)

(2) Switching Operation to ESW Function

Changing the TSG3nOPT2.TSG3nESSC bit from 0 to 1 immediately enables the ESW function and PWM output is switched from to one derived from rectangular waveform input from EMU3. When the TSG3nESSC bit is changed from 1 to 0, time output is synchronized with the counter and switched to HT-PWM mode at reload timing.

(3) Dead Time Insertion when Switching to ESW function

The set dead time is always inserted when switching to ESW function. If PWM output is switched from positive-phase high-level output to positive-phase high-level output of rectangular waveform, output at high level is continued (not to become inactive by switching to ESW function).

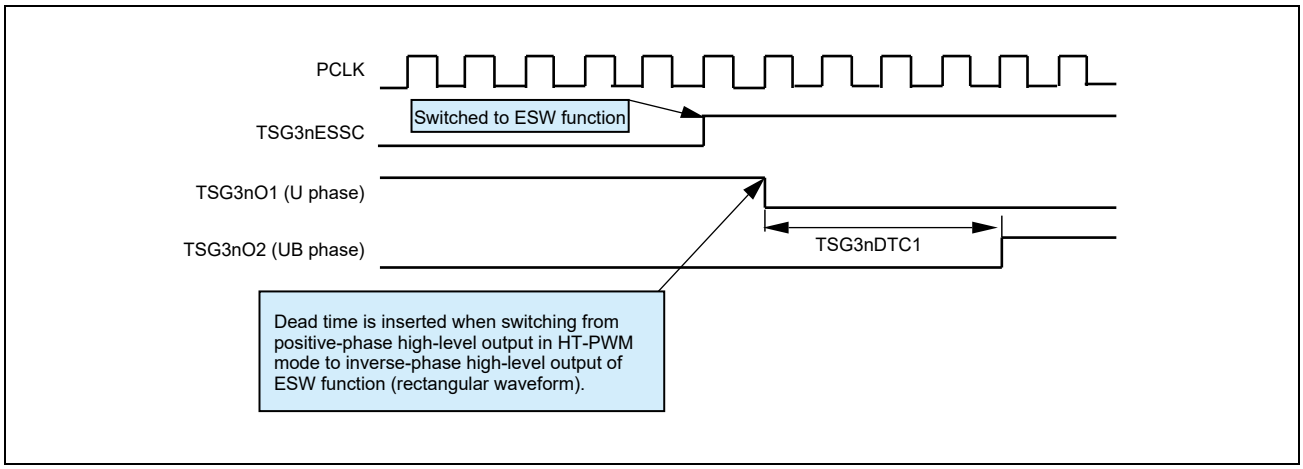


Figure 20.76 Switching Operation to ESW Function (From Positive-Phase High-Level to Inverse-Phase High-Level)

Dead time setting is also effective when ESW function is used. The dead time set in TSG3nDTC0 is inserted to the positive phase (TSG3nO1, 3, and 5), and the dead time set in TSG3nDTC1 is inserted to the inverse phase (TSG3nO2, 4, and 6).

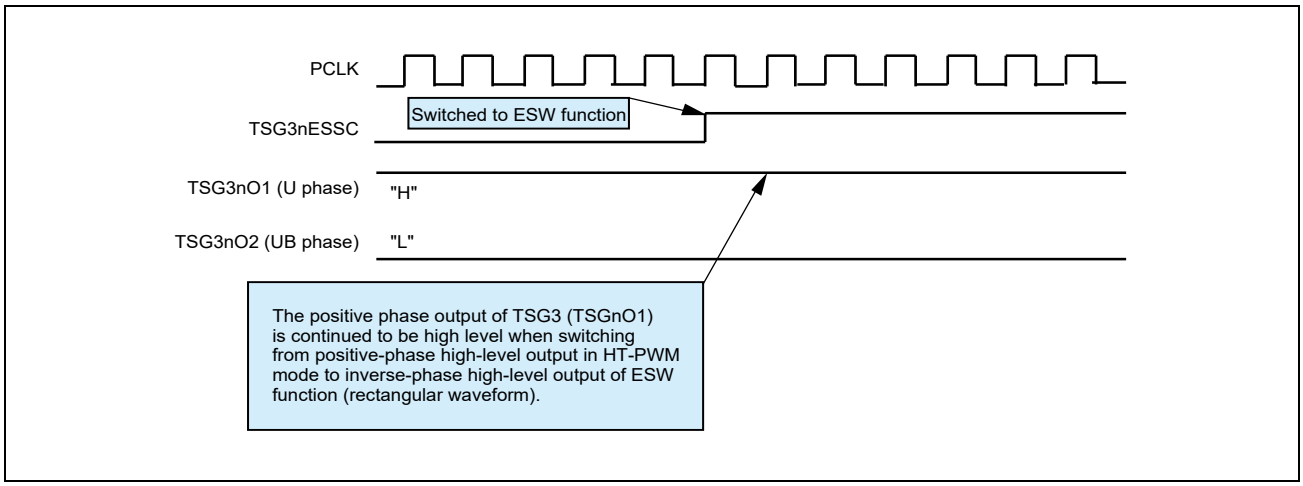


Figure 20.77 Switching Operation to ESW Function (From Positive-Phase High-Level to Positive-Phase High-Level)

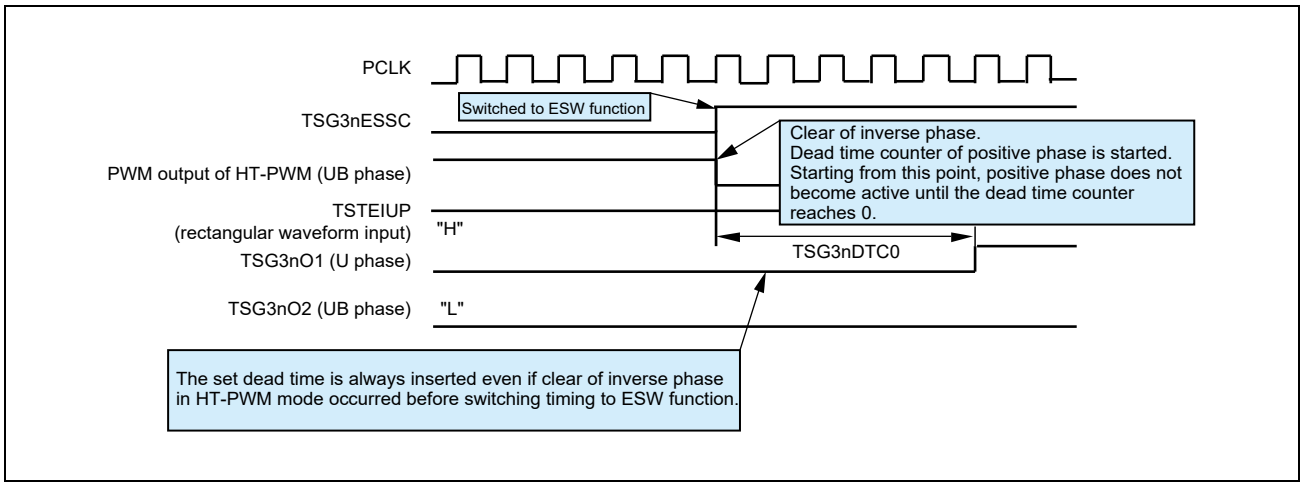


Figure 20.78 Switching Operation to ESW Function (Inverse Phase Cleared Before Switching)

(4) Settings for Operation using ESW Function

Use ESW function with the following settings of bits. Do not modify settings during operation (TSG3nTE = 1).

Table 20.68 List of Settings for Operation using ESW Function

Bit Name	Setting Value	Description
TSG3nCTL0.TSG3nMD2-0	001_B	Switchable only in HT-PWM mode.
TSG3nCTL3.TSG3nRMC	0	Available only in reload mode.
TSG3nIOC3.TSG3nTOL6-1	000000_B	Setting of logical inverse of PWM set/clear is prohibited (HT-PWM mode limitation)
TSG3nOPT0.TSG3nSOC	0	Switching to software control function is prohibited.
TSG3nOPT0.TSG3nSTE	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPOT	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPSS	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nIDC	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPSC	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT1.TSG3nSPC2-0	000_B	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nPAT0W	000000_H	Operation setting of 120-DC mode (value after reset).
TSG3nPAT1W	000000_H	Operation setting of 120-DC mode (value after reset).

20.4.7.5 SP-PWM Mode (Shifted-Pulse - Pulse Width Modulation Mode)

Overview

In this mode, a 6-phase PWM can be generated by using the 18-bit counter and the 18-bit compare registers.

Prerequisites

- The PWM signal cycle is set in TSG3nCMP0E.
- The set timings of the U phase, V phase, and W phase are set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E, while the clear timings of these phases are set with TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E (when set timing and clear timing are used for control).
- The set timings of the U phase, V phase, and W phase are set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E while the active periods of these phases are set with TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE. The sum of the set values of TSG3nCMP2E, TSG3nCMP6E, TSG3nCMP10E, and the set values of TSG3nUPWE, TSG3nVPWE, TSG3nWPWE are set to TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E respectively (when set timing and active period are used for control). The value after addition must be no greater than $3FFFF_H$. Truncate values having 19 or more bits.

Functional description

Set the carrier period and the set timings and duty of U phase, V phase, and W phase. The counting up begins when TSG3nTRG0.TSG3nTS is set to 1.

The 18-bit counter counts up from 00000_H and is cleared by a match with TSG3nCMP0E.

The dead time is set with TSG3nDTC0 and TSG3nDTC1. TSG3nDTC0 sets the dead time of inverse phase (OFF) to positive phase (ON) switch while TSG3nDTC1 sets that of positive phase (OFF) to inverse phase (ON) switch. The 10-bit counters (TSG3nDTT1 to TSG3nDTT3) for dead time generation load the set values of TSG3nDTC0 and TSG3nDTC1 at compare match of the 18-bit counter with the TSG3nCMPmE buffer register (m = 1, 2, 5, 6, 9, 10) and start down-counting.

INTTSG3nIm interrupts (m = 1, 2, 5, 6, 9, 10) are generated by the compare match of the 18-bit counter with the TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E buffer registers.

NOTE

SP-PWM mode is enabled when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 010_B.

(1) Basic Timing Chart

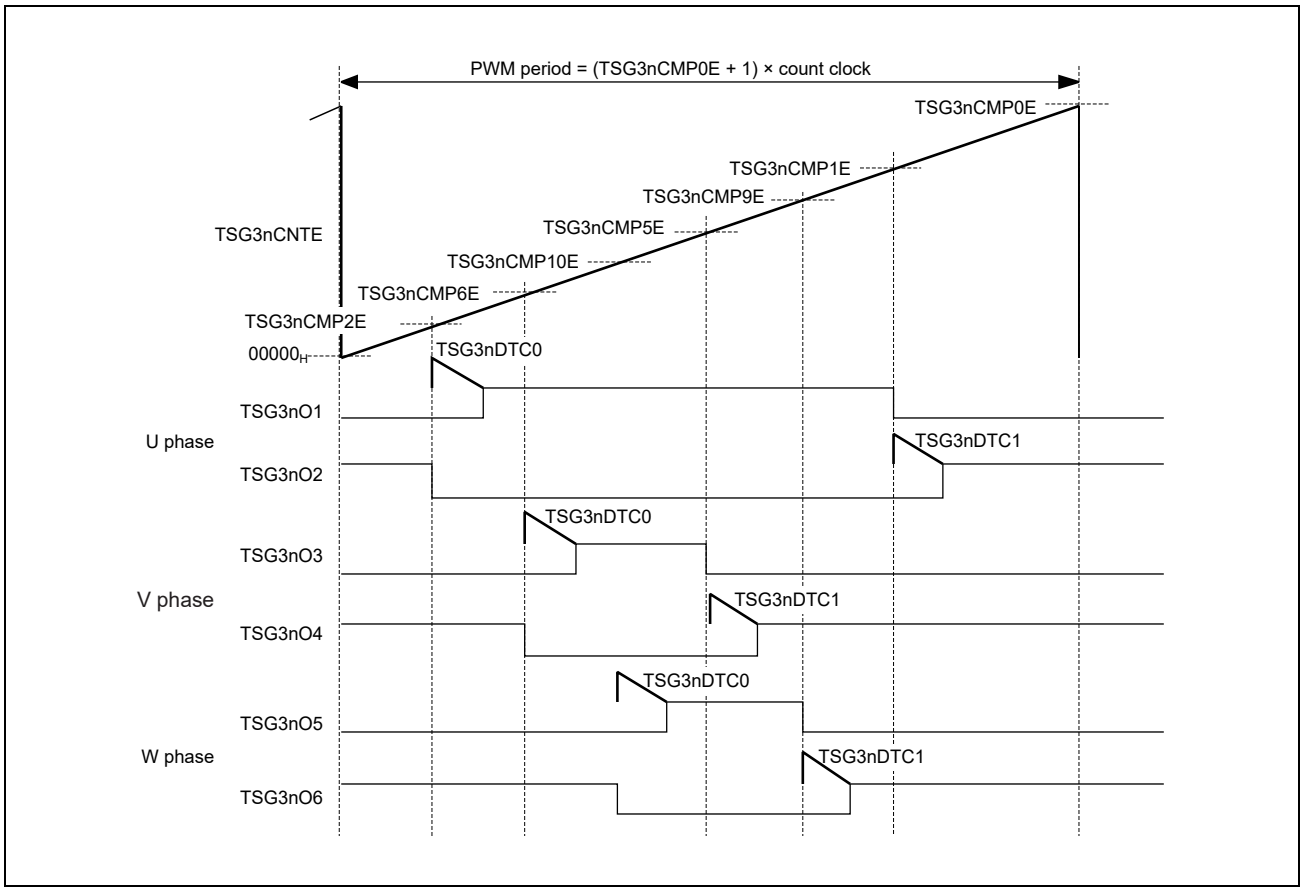
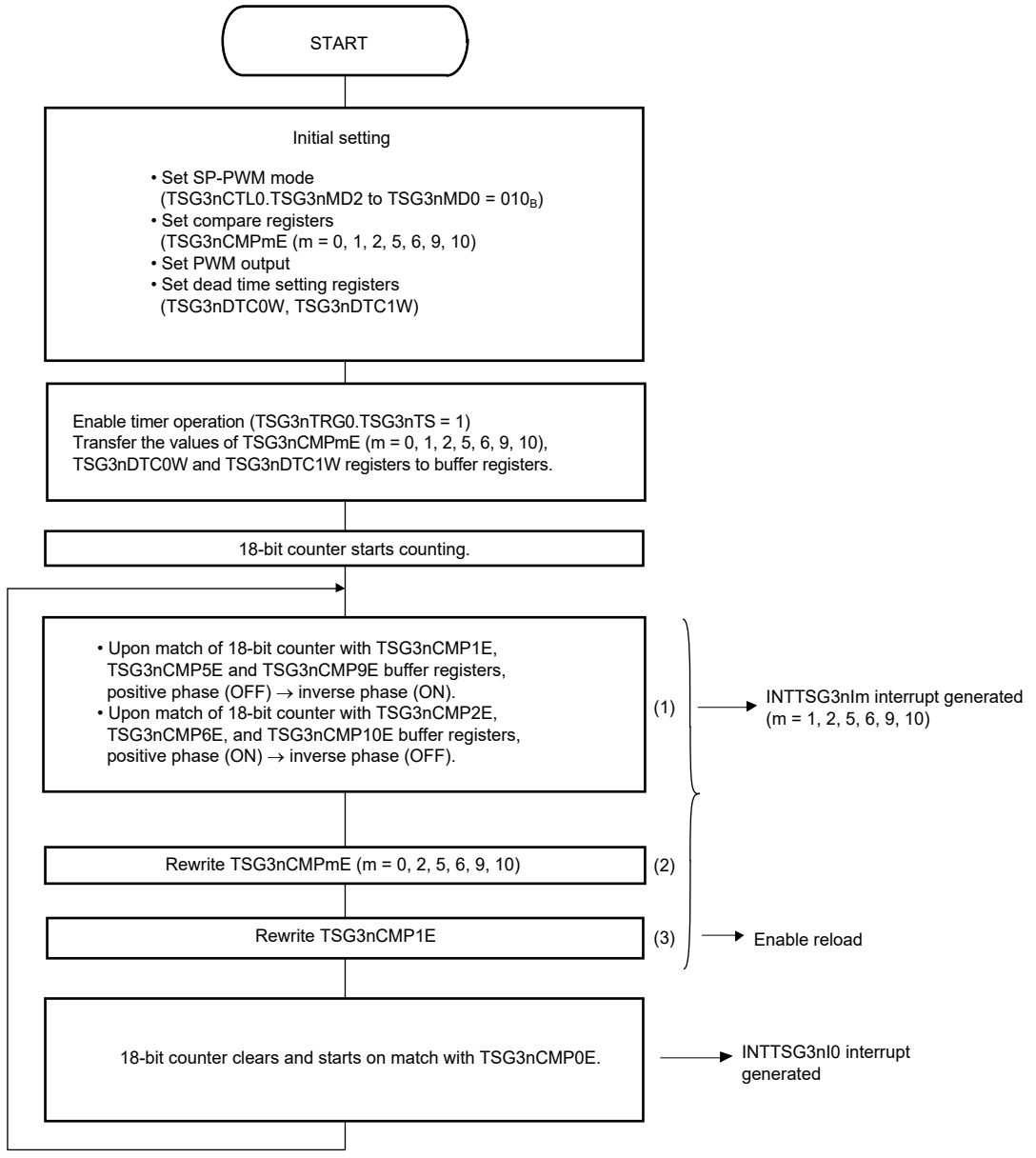


Figure 20.79 Basic Timing in SP-PWM Mode



Note: The timing of (1) may be different depending on the rewriting timing of (2) and (3) and the TSG3nCMP1E value. Be sure to rewrite (2) followed by rewriting (3).

Figure 20.80 Basic Operation Flow in SP-PWM Mode

(2) List of SP-PWM Mode Operations

Table 20.69 Counter Functions in SP-PWM Mode

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger
	Clear	Compare match of TSG3nCMP0E buffer register with 18-bit counter
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 20.70 Compare Registers and Dead Time Setting Register Functions in SP-PWM Mode

Register	Rewriting Method	Rewrite During Operation	Function
TSG3nCMP0E	Reload/Anytime rewrite	Possible	Setting period
TSG3nUPWE	Reload/Anytime rewrite	Possible	PWM control for U phase
TSG3nCMP1E, TSG3nCMP2E	Reload/Anytime rewrite		
TSG3nVPWE	Reload/Anytime rewrite	Possible	PWM control for V phase
TSG3nCMP5E, TSG3nCMP6E	Reload/Anytime rewrite		
TSG3nWPWE,	Reload/Anytime rewrite	Possible	PWM control for W phase
TSG3nCMP9E, TSG3nCMP10E	Reload/Anytime rewrite		
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload/Anytime rewrite	Possible	Diagnostic output or A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Possible	Period and dead time

Table 20.71 Output Functions in SP-PWM Mode

Pin	Function
TSG3nO1	PWM output with dead time by compare match of TSG3nCMP1E buffer register (clear timing) or TSG3nCMP2E buffer register (set timing) with 18-bit counter
TSG3nO2	Output inverse phase with respect to TSG3nO1 (with dead time)
TSG3nO3	PWM output with dead time by compare match of TSG3nCMP5E buffer register (clear timing) or TSG3nCMP6E buffer register (set timing) with 18-bit counter
TSG3nO4	Output inverse phase with respect to TSG3nO3 (with dead time)
TSG3nO5	PWM output with dead time by compare match of the TSG3nCMP9E buffer register (clear timing) or the TSG3nCMP10E buffer register (set timing) with the 18-bit counter
TSG3nO6	Output inverse phase with respect to TSG3nO5 (with dead time)
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger

Table 20.72 Interrupt Requests in SP-PWM Mode

Interrupt	Function
INTTSG3nIm (m = 0, 1, 2, 5, 6, 9, 10)	Compare match of TSG3nCMPmE buffer register with 18-bit counter (m = 0, 1, 2, 5, 6, 9, 10)
INTTSG3nIER	Error
INTTSG3nIVLY	—
INTTSG3nIPEK	Peak interrupt (generated at the same timing as INTTSG3nI0 interrupt)
INTTSG3nIWN	Warning

Table 20.73 Compare Match Timing in SP-PWM Mode

Compare Match	Timing
TSG3nCMP0E	When 18-bit counter changes from TSG3nCMP0E to 00000_H
TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10)	After match of 18-bit counter and TSG3nCMPmE is detected (m = 1, 2, 5, 6, 9, 10)

Table 20.74 Example of Setting Each Timer Output Condition in SP-PWM Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSG3nO1, TSG3nO3, TSG3nO5	PWM output	(TSG3nCMP0E + 1) × count clock	Output an inactive level throughout one period (duty 0%)	TSG3nCMPmE = TSG3nCMP (m + 1) E or TSG3nCMP (m + 1) E > TSG3nCMP0E (m = 1, 5, 9)
			Output an active level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m + 1) E + 1 TSG3nCMP (m + 1) E = TSG3nCMPmE – 1 (m = 1, 5, 9)
			Output an inactive level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m + 1) E - 1 TSG3nCMP (m + 1) E = TSG3nCMPmE + 1 (m = 1, 5, 9)
			Output an active level throughout one period (duty 100%)	TSG3nCMPmE > TSG3nCMP0E TSG3nCMP (m + 1) E ≤ TSG3nCMP0E (m = 1, 5, 9)
TSG3nO2, TSG3nO4, TSG3nO6	PWM output	(TSG3nCMP0E + 1) × count clock	Output an inactive level throughout one period (duty 0%)	TSG3nCMP (m – 1) E > TSG3nCMP0E (m = 2, 6, 10)
			Output an active level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m – 1) E - 1 TSG3nCMP (m – 1) E = TSG3nCMPmE + 1 (m = 2, 6, 10)
			Output an inactive level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m – 1) E + 1 TSG3nCMP (m – 1) E = TSG3nCMPmE – 1 (m = 2, 6, 10)
			Output an active level throughout one period (duty 100%)	TSG3nCMPmE = TSG3nCMP (m – 1) E or TSG3nCMPmE > TSG3nCMP0E (m = 2, 6, 10)
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger	(TSG3nCMP0E + 1) × count clock	Please refer to **Section 20.4.5, A/D Conversion Trigger Function**.	

(3) Various Settings of SP-PWM Mode

Mode setting

SP-PWM mode is entered by setting TSG3nCTL0.TSG3nMD2-TSG3nMD0 to 010_B.

Setting timer output

The output pins TSG3nO1-TSG3nO6 are controlled by setting TSG3nIOC0, TSG3nIOC2, and TSG3nIOC3.

The TSG3nO7 pin provides output pulse as diagnostic output or A/D conversion trigger. The pin should be set as necessary.

Enabling error interrupt generation

Error interrupt (INTTSG3nIER) due to the detection of the simultaneous active state of the positive and inverse phases is enabled by setting TSG3nIOC1.TSG3nEOC to 1. For details, refer to **Section 20.4.6, Error/Warning Interrupt.**

Setting rewriting timing of register with reload function

With the TSG3nCTL3.TSG3nRMC, reload (simultaneous rewrite) or rewrite (anytime) is specified for the registers with reload function. The default setting is 0 (reload). To reload, set TSG3nCTL4.TSG3nPRE to 1.

No reload timing is generated when TSG3nPRE = 0.

When “anytime rewrite” is specified, the unintended output may be generated depending on the rewrite timing.

Setting A/D conversion trigger output

The A/D conversion trigger 0 (TSG3nADTRG0 signal) is set with TSG3nCTL5.TSG3nAT09 to TSG3nAT00.

TSG3nAT09 to TSG3nAT00 is used to enable or disable the A/D conversion trigger output on timing match of TSG3nDCMP2E to TSG3nDCMP0E with the 18-bit counter (up count).

TSG3nCTL6.TSG3nAT19 to TSG3nAT10 is used to set the A/D conversion trigger 1 (TSG3nADTRG1 signal).

To set the match timing of the 18-bit counter and TSG3nDCMP2E to TSG3nDCMP0E, set the compare value to each register.

The skipping function can be used for TSG3nADTRG0 and TSG3nADTRG1 signals. TSG3nACC00 and TSG3nACC01 of TSG3nCTL5, and TSG3nACC10 and TSG3nACC11 of TSG3nCTL6 can be used to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTION

- Be sure to set TSG3nCTL5, TSG3nCTL6, and TSG3nDCMP2E to TSG3nDCMP0E correctly when the timing pulse of A/D conversion trigger is output to TSG3nO7.
- In SP-PWM mode, no trough interrupt (INTTSG3nIVLY) is generated. Therefore, TSG3nCTL5.TSG3nAT00 and TSG3nCTL6.TSG3nAT10 must be set to 0.
- In SP-PWM mode, the 18-bit sub-counter does not operate. Therefore, TSG3nAT09, and TSG3nAT08 in TSG3nCTL5, and TSG3nAT19, and TSG3nAT18 in TSG3nCTL6 must be set to 0.
- In SP-PWM mode, down counting by the 18-bit counter is not generated. Therefore, TSG3nAT07, TSG3nAT05, and TSG3nAT03 in TSG3nCTL5 and TSG3nAT17, TSG3nAT15, and TSG3nAT13 in TSG3nCTL6 should be set to 0.

Setting dead time

The dead time can be set with TSG3nDTC0 and TSG3nDTC1. The dead time is calculated by the following expressions:

- PCLK × TSG3nDTC0
- PCLK × TSG3nDTC1

TSG3nDTC0 can set the time between a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the inactive state and a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the active state, respectively.

TSG3nDTC1 can set the time between a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the inactive state, and a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the active state, respectively.

Carrier period

Set the carrier period with TSG3nCMP0E according to the following expression:

- TSG3nCMP0E = (carrier period/count clock period) – 1

CAUTION

PWM output with 100% duty cannot be produced when TSG3nCMP0E = $3FFFF_H$.

Setting duty (PWM width)

The duty of U phase, V phase, and W phase is set with TSG3nCMPmE, TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE (m = 1, 2, 5, 6, 9, 10), respectively.

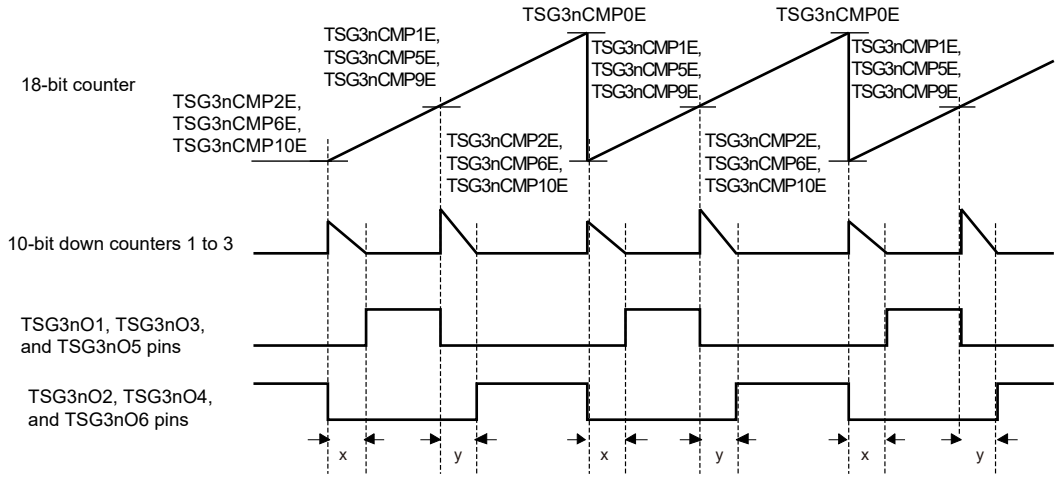
- The set timings of the U phase, V phase, and W phase are set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E, and the clear timings are set with TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E. (The set and clear timing setting is used for control.)
- The set timings of U phase, V phase, and W phase is set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E while the active periods of these phases are set with TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE. The sum of the set values of TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E, and the set values of TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE are set to TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E respectively (when set timing and active period are used for control).

(4) Dead Time Control in SP-PWM mode

Duty setting registers are TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10), TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE and register for setting the period is TSG3nCMP0E. The 6-phase PWM waveform of a variable duty is output by using these registers. To achieve the dead time control, there are six 10-bit down counters that operate synchronously with the count clock of the 18-bit counter and two dead time setting registers (TSG3nDTC0W and TSG3nDTC1W). TSG3nDTC0W is used for setting a dead time from a change of the inverse phase to the inactive state to a change of the positive phase to the active state. TSG3nDTC1W is used for setting a dead time from a change of the positive phase to the inactive state to a change of the inverse phase to the active state.

Dead time counter keeps operating even when operation stop (TSG3nTE = 0) setting collided with dead time insert timing, and the dead time set in TSG3nO1 and TSG3nO2 are always inserted.

The following figure shows an example of the output waveform.



Note: x: TSG3nDTC0 register values, y: TSG3nDTC1 register values

Figure 20.81 Example of Output Waveform in SP-PWM Mode

(5) Software Output Control Function in SP-PWM Mode

TSG3nOPT0.TSG3nSOC, TSG3nIDC, and TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 are used to control timer output by software.

As shown in **Figure 20.82**, the output control is switched immediately when TSG3nSOC is set to 1. If the dead time is set, the period of the dead time is guaranteed. After that, when TSG3nSOC is set to 0, output control is retained. When the reload timing is generated, output control is switched to SP-PWM mode output control.

For details, refer to **Section 20.4.7.10, Software Output Control Function**.

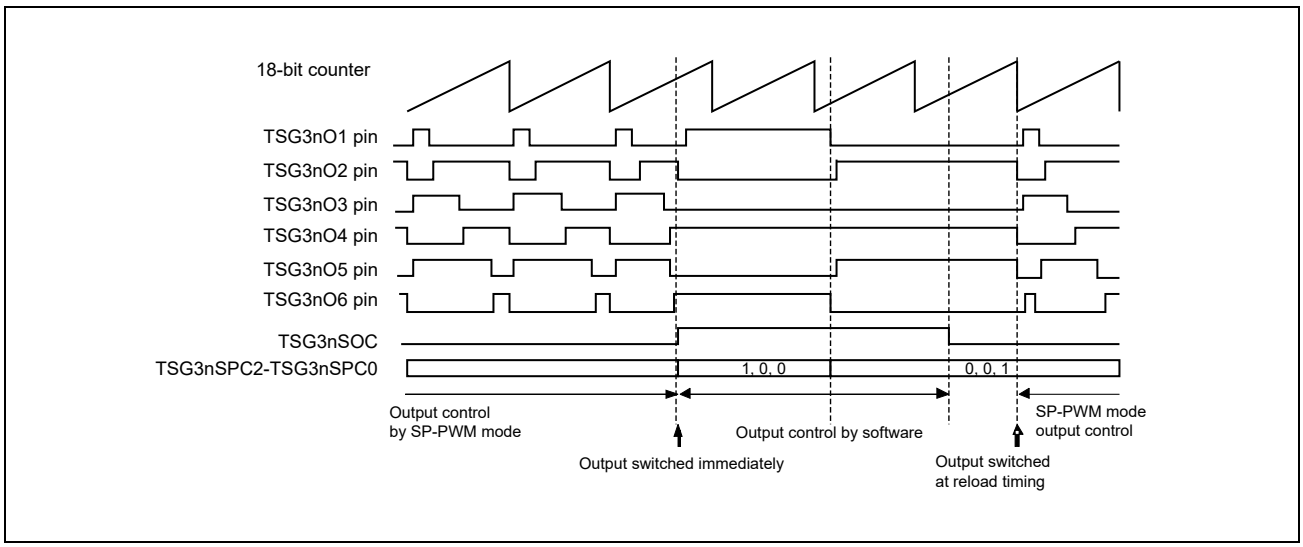


Figure 20.82 Example of Output Control Switching from SP-PWM Mode Control to Software Control

(a) Procedure on Software Output Control Processing

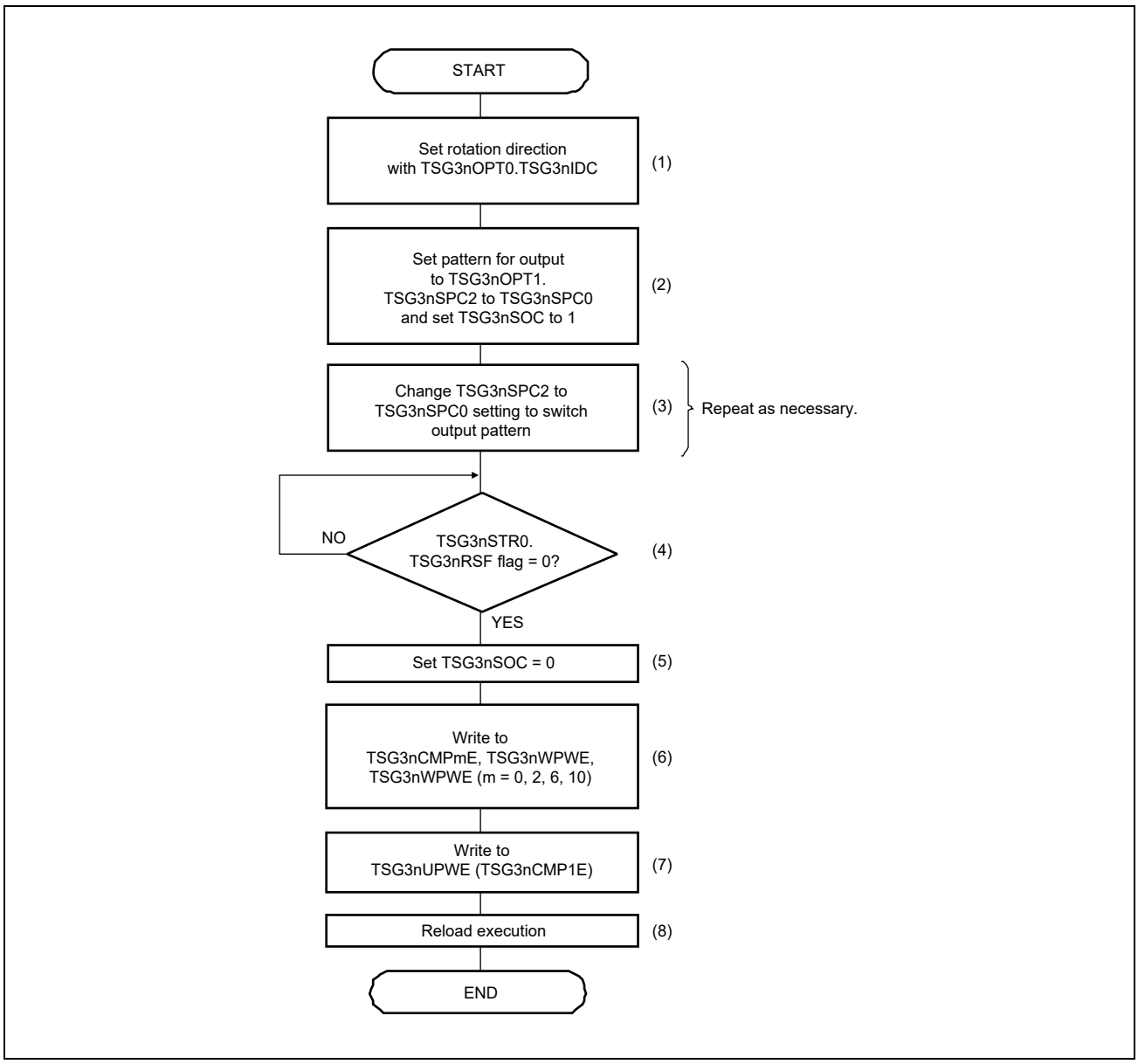


Figure 20.83 Flow of Software Output Control

The procedure for software output control is described below.

(1) Set TSG3nIDC to determine the electric current direction. The timer output has a 180-degreephase shift between when TSG3nIDC = 0 and when TSG3nIDC = 1. When this bit is rewritten with the software output control function, the output pattern will change according to the new setting at the next timer cycle.

(2) Set the pattern for output to TSG3nSPC2-0. At the same time, set TSG3nSOC to 1 to switch to software output control mode.

(3) Change the output pattern setting for TSG3nSPC2-0 to change the timer output.

(4) Ensure that the reload request flag TSG3nRSF is 0. If TSG3nRSF is 1, do not shift to the following procedure until it goes 0.

(5) Clearing TSG3nSOC to 0 starts releasing of software control (it is not released here yet).

(6) Make necessary settings of the compare registers that will be used after the software output control is released. Proceed to the following step when the register settings are not required. Here, change the registers with the reload function.

(7) Write TSG3nUPWE (TSG3nCMP1E) to start reloading.

(8) Reload is executed and software output control is released.

CAUTION

Be sure to execute reload after execution of steps (4), (5), (6), and (7). Unless reload can be executed, software output control cannot be released

20.4.7.6 120-DC Mode

Overview

In this mode, PWM output period set to TSG3nCMP0E and timer output (TSG3nO1 to TSG3nO6) according to the duty cycle set to TSG3nCMP1E to TSG3nCMP12E are controlled with three types of pattern inputs (software output control method, pattern switch method, and trigger switch method) to perform 120-DC control.

Note that pattern switch method and trigger switch method cannot be used with TSG32 because it does not support the functions of external pattern input and linkage with ENCA.

Prerequisites

- Set the PWM period to TSG3nCMP0E.
- Set the PWM duty to TSG3nCMP1E to TSG3nCMP12E and set the output pattern to TSG3nPAT0W and TSG3nPAT1W.

Functional description

Set the PWM period, set the duty cycle to individual compare register, and set the pattern to be output to the pattern register. Setting TSG3nTRG0.TSG3nTS to 1 starts counting.

The 18-bit counter starts counting from 00000_H, and is cleared by the match with TSG3nCMP0E.

INTTSG3nI1 to INTTSG3nI12 interrupts are generated by a compare match of the 18-bit counter and TSG3nCMP1E to TSG3nCMP12E buffer registers, respectively.

NOTE

120-DC mode is valid when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 011_B.

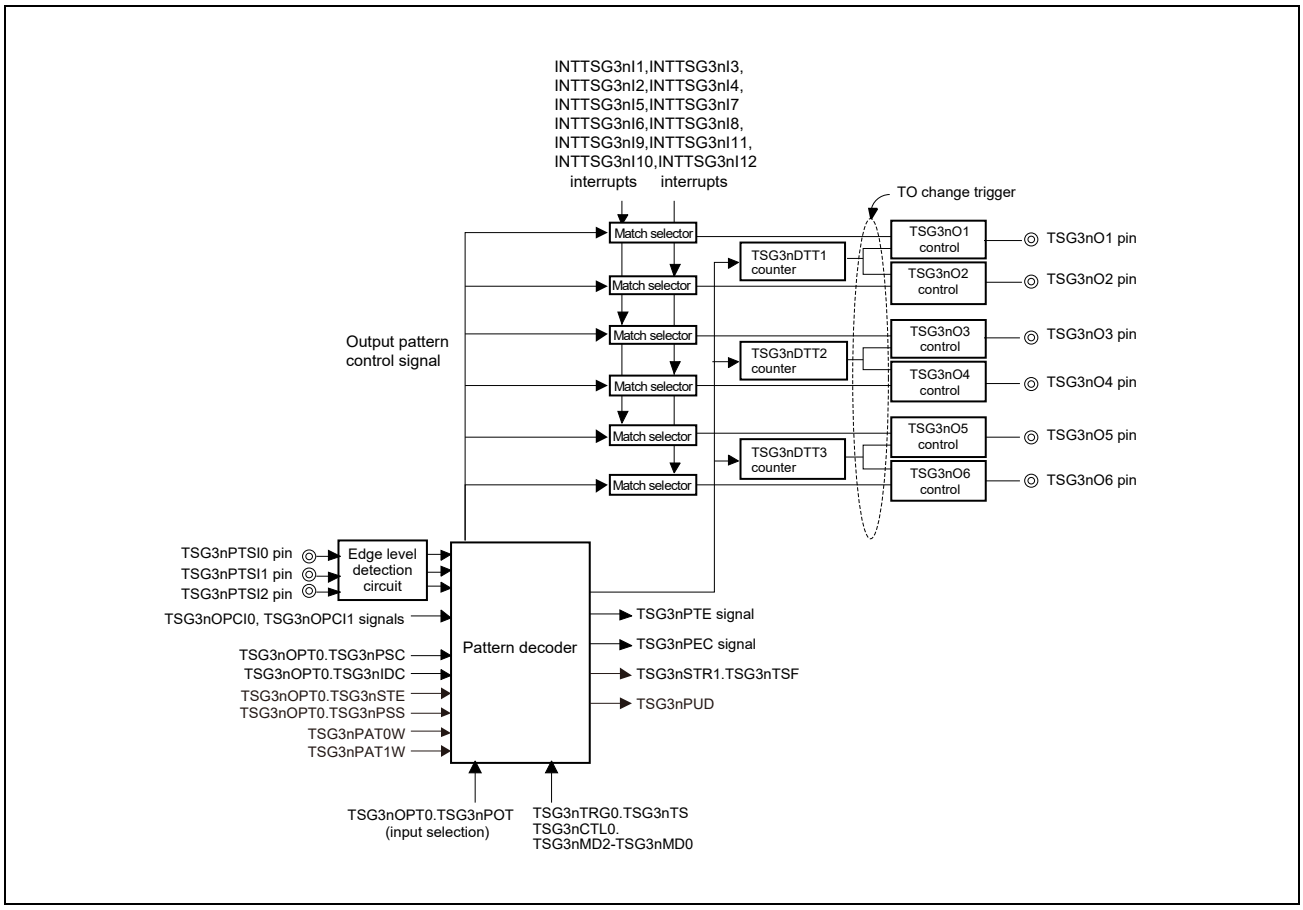
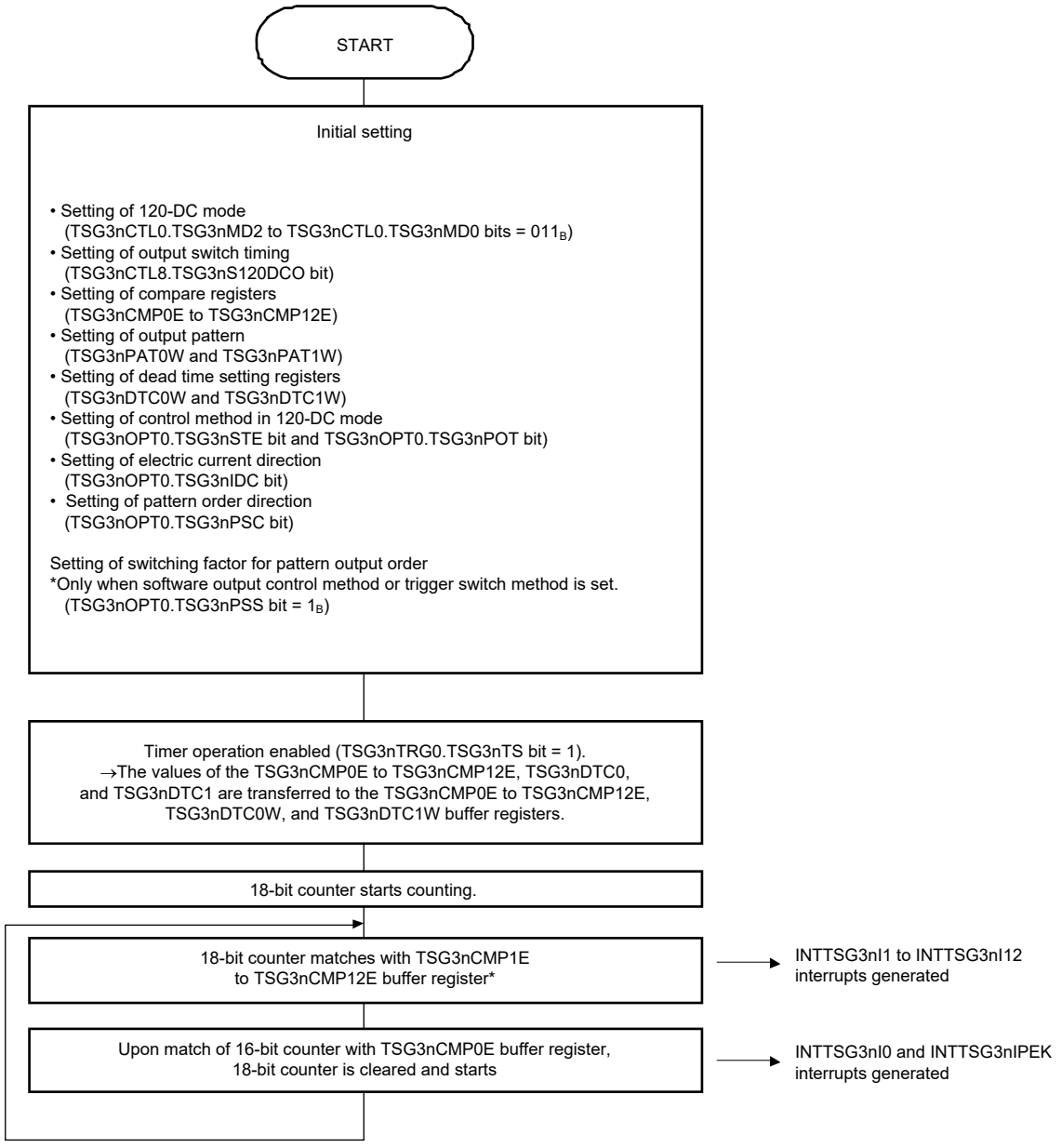


Figure 20.84 Block Diagram in 120-DC Mode



Note: The 18-bit counter is not cleared by match of the 18-bit counter with the TSG3nCMP1E to TSG3nCMP12E buffer registers

Figure 20.85 Basic Operation Flow in 120-DC Mode

(1) List of Operations in 120-DC Mode

Table 20.75 Counter Functions in 120-DC Mode

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger
	Clear	When TSG3nCTL8.TSG3nS120DCO is 0: timing of a match of TSG3nCMP0E value and 18-bit counter value or output pattern switch When TSG3nCTL8.TSG3nS120DCO is 1: timing of a match of TSG3nCMP0E value with 18-bit counter value
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 20.76 Functions of Compare Registers and Dead Time Setting Registers in 120-DC Mode

Register	Rewrite Method	Rewrite during Operation	Function
TSG3nCMP0E	Reload	Possible	Setting period
TSG3nCMPmE (m = 1 to 12)	Reload	Possible	Setting PWM duty
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload	Possible	Outputting diagnostic signal or setting A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Possible	Setting dead time

Table 20.77 Timer Input Function in 120-DC Mode

Pin/Signal	Function
TSG3nPTSI2-TSG3nPTSI0 pins	Pattern input (3 phases)
TSG3nOPCI0, TSG3nOPCI1 signals	Trigger input

Table 20.78 Timer Output Function in 120-DC Mode

Pin/Signal	Function
TSG3nO1 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 1, 2, 5, 6, 9, 10) with the 18-bit counter and output pattern selected by TSG3nPAT0W setting.
TSG3nO2 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 3, 4, 7, 8, 11, 12) with the 18-bit counter and output pattern selected by TSG3nPAT1W setting.
TSG3nO3 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 1, 2, 5, 6, 9, 10) with the 18-bit counter and output pattern selected by TSG3nPAT0W setting.
TSG3nO4 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 3, 4, 7, 8, 11, 12) with the 18-bit counter and output pattern selected by TSG3nPAT1W setting.
TSG3nO5 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 1, 2, 5, 6, 9, 10) with the 18-bit counter and output pattern selected by TSG3nPAT0W setting.
TSG3nO6 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 3, 4, 7, 8, 11, 12) with the 18-bit counter and output pattern selected by TSG3nPAT1W setting.
TSG3nO7 pin	Diagnostic signal output or pulse output by A/D conversion trigger
TSG3nPTE signal	Toggle signal by change in input pattern

Table 20.79 Interrupt Requests in 120-DC Mode

Interrupt	Function
INTTSG3nIm (m = 0 to 12)	Compare match of TSG3nCMPmE buffer register and 18-bit counter (m = 0 to 12)
INTTSG3nIER	Error
INTTSG3nIVLY	—
INTTSG3nIPEK	Peak interrupt (generated at the same timing as INTTSG3nI0)
INTTSG3nIWN	Warning interrupt

Table 20.80 Compare Match Timing in 120-DC Mode

Compare Match	Timing
TSG3nCMP0E	When 18-bit counter changes from TSG3nCMP0E to 00000_H
TSG3nCMPmE (m = 1 to 12)	After detecting the match of 18-bit counter and TSG3nCMPmE (m = 1 to 12)

Table 20.81 Example of Setting Each Timer Output Condition in 120-DC Mode

			Output Duty	
Pin	Item	Output Period	Output Condition	Setting Condition
TSG3nOm (m = 1 to 6)	PWM output	(TSG3nCMP0E + 1) × count clock	See **Section (6), List of Output Patterns in 120-DC Mode**.	—
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger	(TSG3nCMP0E + 1) × count clock	See **Section 20.4.5, A/D Conversion Trigger Function**.	—

(2) Various Settings of 120-DC Mode

Mode setting

120-DC mode is entered by setting TSG3nCTL0.TSG3nMD2-TSG3nMD0 to 011_B.

Setting timer output

The output pins TSG3nO1 to TSG3nO6 are controlled by setting TSG3nIOC0, TSG3nIOC2, and TSG3nIOC3.

The TSG3nO7 pin outputs pulses of the diagnostic output or the A/D conversion trigger. Please set it as required.

Enabling error interrupt generation

With TSG3nIOC1.TSG3nEOC = 1, the error interrupt (INTTSG3nIER) generation is enabled when the simultaneous active state of the positive phase and inverse phase is detected. For details, see **Section 20.4.6, Error/Warning Interrupt**.

Setting register rewrite timing

Reloading the registers with the reload function is activated with TSG3nCTL3.TSG3nRMC (simultaneous rewrite; default setting is 0 = reload). Set TSG3nCTL4.TSG3nPRE to 1 when reload is used.

The reload timing is not generated if TSG3nPRE is 0.

Setting A/D conversion trigger output

To set A/D conversion trigger 0 (TSG3nADTRG0 signal), use TSG3nCTL5.TSG3nAT09 to TSG3nAT00.

With TSG3nAT09 to TSG3nAT00, A/D conversion trigger output is enabled or disabled at the match of the 18-bit counter and TSG3nDCMP2E to TSG3nDCMP0E (during up count).

To set A/D conversion trigger 1 (TSG3nADTRG1 signal), use TSG3nCTL6.TSG3nAT19 to TSG3nAT10.

To set the match timing of the 18-bit counter and TSG3nDCMP2E to TSG3nDCMP0E, set the compare value to the pertinent register.

The skipping function can be used for TSG3nADTRG0 and TSG3nADTRG1 signals. Use TSG3nACC01, TSG3nACC00 of TSG3nCTL5 and TSG3nACC11, and TSG3nACC10 in TSG3nCTL6 to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTION

- Set TSG3nCTL5, TSG3nCTL6, and TSG3nDCMP2E to TSG3nDCMP0E correctly when using the TSG3nO7 output for the A/D conversion trigger timing pulse.
- In 120-DC mode, a trough interrupt (INTTSG3nIVLY) is not generated. Therefore, set TSG3nAT00 and TSG3nAT10 in TSG3nCTL5 and TSG3nCTL6 to 0.
- In 120-DC mode, the 18-bit sub-counter does not operate. Therefore, set TSG3nAT09, TSG3nAT08, TSG3nAT19, and TSG3nAT18 in TSG3nCTL5 and TSG3nCTL6 to 0.
- In 120-DC mode, the 18-bit counter does not decrement. Therefore, set TSG3nAT07, TSG3nAT05, TSG3nAT03, TSG3nAT17, TSG3nAT15, and TSG3nAT13 in TSG3nCTL5 and TSG3nCTL6 to 0.

Setting dead time

The dead time can be set with TSG3nDTC0 and TSG3nDTC1.

The dead time is calculated by the following expressions

- PCLK × TSG3nDTC0
- PCLK × TSG3nDTC1

TSG3nDTC0 can set the time between a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the inactive state and a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the active state, respectively.

TSG3nDTC1 can set the time between a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the inactive state to a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the active state.

Carrier period

Set the carrier period with TSG3nCMP0E according to the following expression:

- TSG3nCMP0E = (carrier period/count clock period) – 1

Duty (PWM width) setting

The duty of PWM output is set with TSG3nCMP1E to TSG3nCMP12E. The setting range of the compare registers is as follows:

- $00000_H \leq$ TSG3nCMPmE $\leq$ TSG3nCMP0E + 1

CAUTION

Do not set TSG3nCMPmE = TSG3nCMP0E + 1 (m = 1 to 12) only when TSG3nCMP0E + 1 < TSG3nCMPmE, and TSG3nCMP0E = $3FFFF_H$.

Output PWM setting

In 120-DC mode, the output pins TSG3nO1, TSG3nO3, and TSG3nO5 are controlled by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E, and the output pins TSG3nO2, TSG3nO4, TSG3nO6 are controlled by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E. The duty cycle of a PWM period (TSG3nCMP0E) can be set with TSG3nCMP1E to TSG3nCMP12E. Setting TSG3nCMP1E to TSG3nCMP12E to 00000_H sets the PWM duty cycle to 0%. Setting TSG3nCMP1E to TSG3nCMP12E to TSG3nCMP0E + 1 value sets the PWM duty cycle to 100%. This allows chopping output control and rectangular wave output control.

(3) Control Methods in 120-DC Mode

Control methods in 120-DC mode are listed below.

Table 20.82 Control Method in 120-DC Mode

Control Method	Function
Software output control method	Switches the output pattern according to the TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 setting made by software.
Pattern switch method	Switches the output pattern by the pattern input signal of TSG3nPTSI0 to TSG3nPTSI2.
Trigger switch method	Switches the output pattern by the trigger switch method using the trigger input signals TSG3nOPCI0 and TSG3nOPCI1 or by the pattern input setting of TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 in the constant order.

Switch timing of timer output when changing input pattern of 120-DC mode can be set by TSG3nCTL8.TSG3nS120DCO.

Table 20.83 Setting of TSG3nS120DCO and Operation in 120-DC Mode

TSG3nS120DCO	Function
0	When input patterns are changed, the main counter (TSG3nCNTE) is cleared and the change of patterns is immediately reflected to timer output.
1	When input patterns are changed, the change of input patterns is reflected to timer output from the next timer cycle (after a match of the main counter (TSG3nCNTE) with TSG3nCMP0E).

Setting software output control method

Setting TSG3nOPT0.TSG3nSTE = 0 switches the output pattern by software output control. The TSG3nO1 to TSG3nO6 pin output is switched according to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0.

The pattern output order at the beginning of operation is set with TSG3nOPT0.TSG3nIDC and TSG3nOPT0.TSG3nPSC.

Operation of software output control method

The PWM output of TSG3nO1 to TSG3nO6 pins (PWM output defined by TSG3nCMP1E to TSG3nCMP12E) is selected by TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 by software. To control the dead time, the dead time counter is activated at the falling edge of the signals in each phase and the dead time is inserted.

The 18-bit counter counts based on the carrier period set in TSG3nCMP0E. The 18-bit counter is cleared by match of the 18-bit counter and TSG3nCMP0E or by a write access to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 (when TSG3nS120DCO = 0).

In this method, the pattern is output, which is decoded using information on the output pattern (TSG3nSPC2 to TSG3nSPC0), the electric current direction control bit (TSG3nOPT0.TSG3nIDC), and order detection control bit (TSG3nOPT0.TSG3nPSC). **Figure 20.106** shows the timer output when the output pattern is changed by software output control.

Immediately after the operation starts (TSG3nTRG0.TSG3nTS = 1), the output pattern of TSG3nSPC2 to TSG3nSPC0 and the pattern set with TSG3nIDC and TSG3nPSC (TSG3nOPT0.TSG3nPSS = 1) are output.

Setting pattern switch method

Setting TSG3nOPT0.TSG3nSTE to 1 and TSG3nPOT to 0 selects the pattern switch method. The TSG3nO1 to TSG3nO6 pin output pattern is changed at the change timing of the TSG3nPTSI2 to TSG3nPTSI0 pins.

The output pattern at the beginning of operation is set with TSG3nOPT0.TSG3nIDC and with TSG3nOPT0.TSG3nPSC. However, after determining the rotation direction (after the value is set to TSG3nSTR1.TSG3nTSF), the setting of TSG3nPSC is disabled.

Operation of pattern switch method

After level detection is performed for the pins (three inputs from the hall sensor), the level-detected signals are decoded. From the decoding result, the PWM output of TSG3nO1 to TSG3nO6 pins (PWM output defined by TSG3nCMP1E to TSG3nCMP12E) is selected. To control the dead time, the dead time counter is activated at the falling timing of signals in each phase and the dead time is inserted.

The 18-bit counter is cleared by a match of the 18-bit counter value with the TSG3nCMP0E value or by a change of the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) while TSG3nS120DCO is 0.

In this method, the pattern, which is decoded by using information on input pattern (TSG3nPTSI2 to TSG3nPTSI0), the electric current direction control bit (TSG3nOPT0.TSG3nIDC), and TSG3nPTSI2 to TSG3nPTSI0 pattern order detection flag (TSG3nSTR1.TSG3nTSF), is output. **Figure 20.88 to Figure 20.91** show the timer output when TSG3nPTSI2 to TSG3nPTSI0 pin inputs change.

Immediately after the operation starts (TSG3nTRG0.TSG3nTS = 1), the output pattern set by the levels input on the TSG3nPTSI2 to TSG3nPTSI0 pins and by the TSG3nIDC and TSG3nPSC bits is produced. If a level on any of the TSG3nPTSI2 to TSG3nPTSI0 pins is changed, TSG3nTSF is determined by the direction of the change to the sequence. After the TSG3nTSF value is determined, the pattern set by the TSG3nTSF bit replaces that set by the TSG3nPSC bit.

CAUTION

When connecting the three-phase pulse input signal to the TSG3nPTSI2 to TSG3nPTSI0 pins, confirm that the three-phase pulse input value and the patterns output from the TSG3nO1 to TSG3nO6 pins satisfy the expected conditions.

If the expected conditions are not satisfied, change the connection between the three-phase pulse input signal and the TSG3nPTSI2 to TSG3nPTSI0 pins.

Setting trigger switch method

Setting TSG3nOPT0.TSG3nSTE and TSG3nPOT to 1 selects the trigger switch method. The output patterns of the pins TSG3nO1 to TSG3nO6 are changed at a rising edge of an external input (TSG3nOPCI1 and TSG3nOPCI0 signals).

For pattern output order, see **Section 20.4.7.6(5), Operation in 120-DC Mode**.

The initial output pattern can be controlled with TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0.

When starting the TSG3n operation (TSG3nTRG0.TSG3nTS = 1) after setting TSG3nSPC2 to TSG3nSPC0, the initial pattern is output. For details, see **Section 20.4.7.6(6) List of Output Patterns in 120-DC Mode**.

Operation of trigger switch method

With the trigger input switch method, the rising edges of the TSG3nOPCI0 and TSG3nOPCI1 signals are detected and the output switch timing is generated. The initial timer output pattern is set with TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0. The subsequent output patterns are switched after rising of the TSG3nOPCI0 and TSG3nOPCI1 signals is detected. Furthermore, the output patterns can be switched by setting TSG3nOPT1.TSG3nSPC2 to TSG3nOPT1.TSG3nSPC0.

The 18-bit counter counts based on the carrier period set in TSG3nCMP0E. The 18-bit counter is cleared by match of the 18-bit counter and TSG3nCMP0E, by a write access to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0, or by detecting a rising of TSG3nOPCI0 and TSG3nOPCI1 signals. (When TSG3nS120DCO = 0)

For examples of operation in 120-DC mode when trigger input switch method is used, see **Figure 20.88** to **Figure 20.91**.

CAUTION

The initial pattern should be set according to the read input level of the port to which TSG3nPTSI2 to TSG3nPTSI0 pins are connected.

(4) Timer Output in 120-DC Mode

In 120-DC mode, the PWM output is controlled with TSG3nPAT0W, TSG3nPAT1W, and TSG3nCMP1E to TSG3nCMP12E. TSG3nPAT0W, TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E are set to control the output of TSG3nO1, TSG3nO3, and TSG3nO5 pins. TSG3nPAT1W, TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E are set with the output of SG2nO2, TSG3nO4, and TSG3nO6 pins.

With PWM output control, eight types of output patterns can be selected for each of TSG3nO1, TSG3nO3, and TSG3nO5 pins and TSG3nO2, TSG3nO4, and TSG3nO6 pins.

Table 20.84 TSG3nPAT0W Set Value and Output Control

PATmT Value	Output Control
000	Fixed to low
001	PWM output set with TSG3nCMP1E
010	PWM output set with TSG3nCMP2E
011	PWM output set with TSG3nCMP5E
100	PWM output set with TSG3nCMP6E
101	PWM output set with TSG3nCMP9E
110	PWM output set with TSG3nCMP10E
111	Fixed to high

Note: m = 0, 1, 2, 3, 4, 5

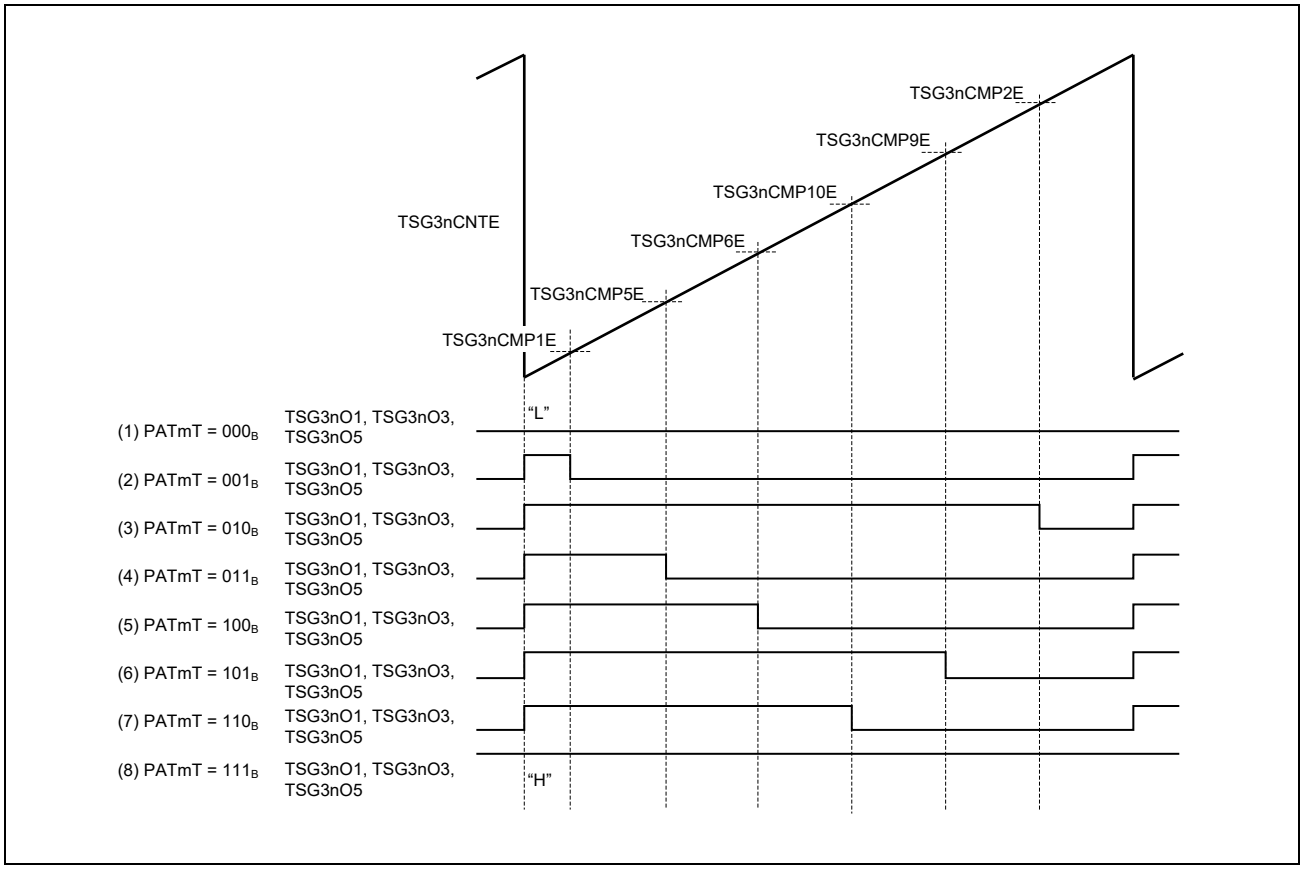


Figure 20.86 TSG3nO1, TSG3nO3, TSG3nO5 Pin Output of Each Output Pattern

Table 20.85 TSG3nPAT1W Set Value and Output Control

PATmB Value	Output Control
000	Fixed to low
001	PWM output set with TSG3nCMP3E
010	PWM output set with TSG3nCMP4E
011	PWM output set with TSG3nCMP7E
100	PWM output set with TSG3nCMP8E
101	PWM output set with TSG3nCMP11E
110	PWM output set with TSG3nCMP12E
111	Fixed to high

Note: m = 0, 1, 2, 3, 4, 5

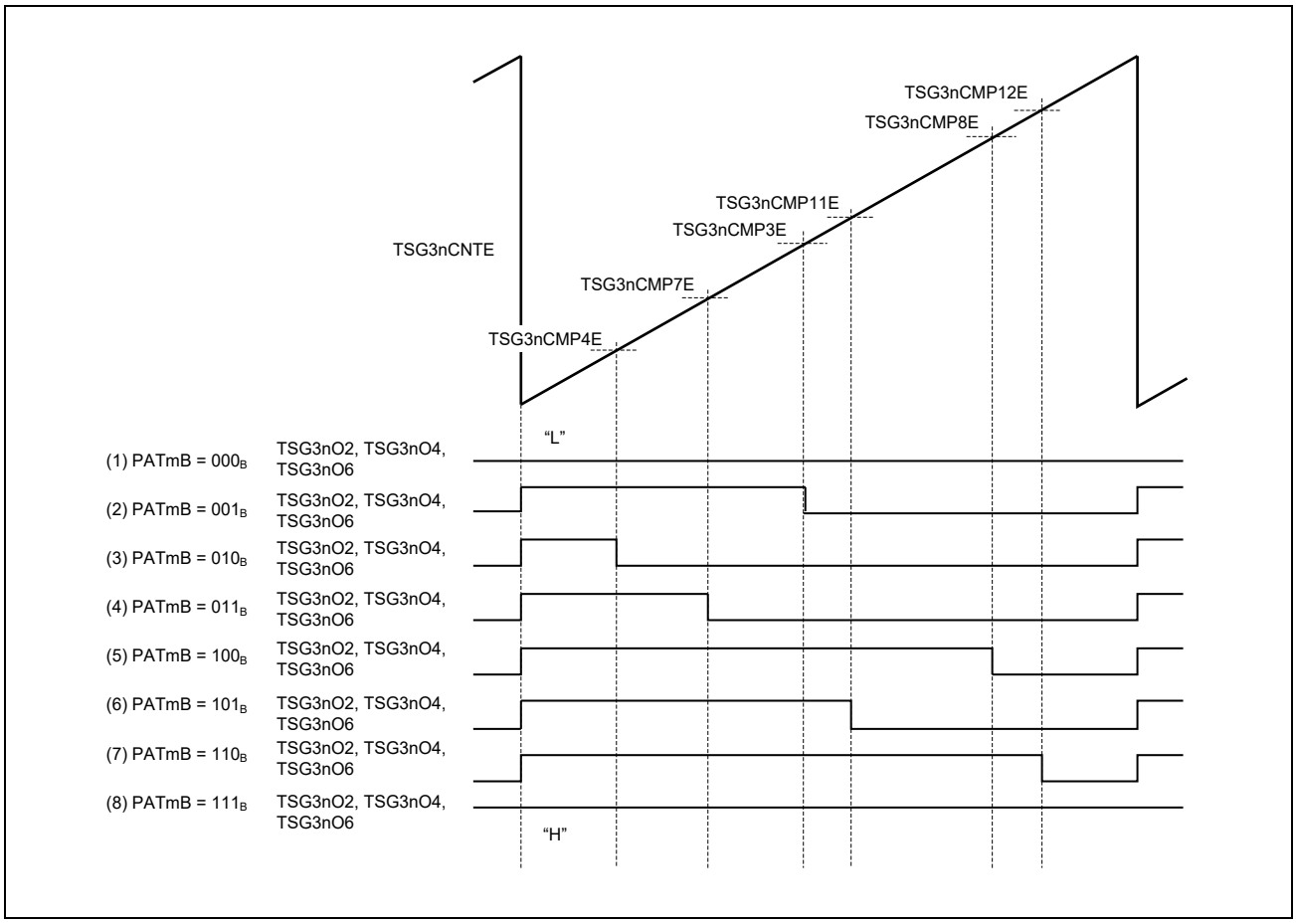


Figure 20.87 TSG3nO2, TSG3nO4, TSG3nO6 Pin Output of Each Output Pattern

(5) Operation in 120-DC Mode

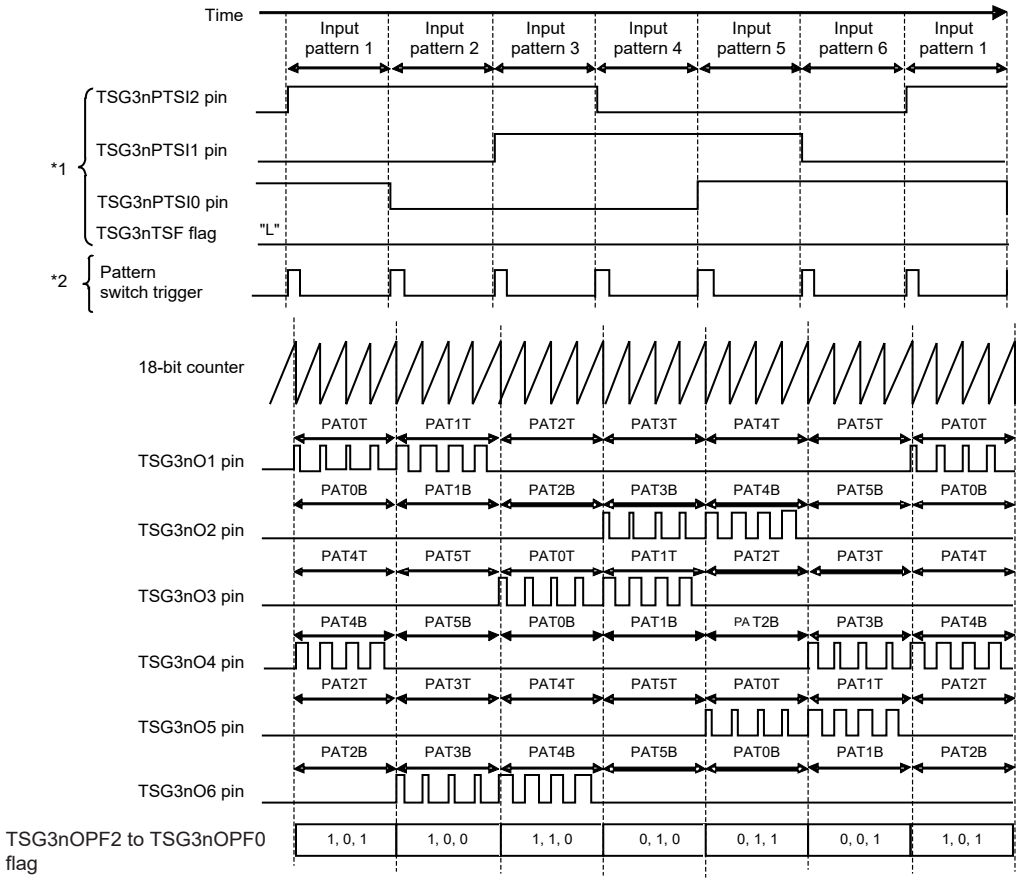
Figure 20.88 to **Figure 20.91** show examples of operation in 120-DC mode.

The TSG3nO1 to TSG3nO6 pins detect the input level change of the TSG3nPTSI2 to TSG3nPTSI0 pins, and then change the output pattern. The 18-bit counter produces sawtooth waveform and TSG3nCMP0E to TSG3nCMP12E output PWM signal. When TSG3nS120DCO = 0, the 18-bit counter is cleared to 00000_H each time the counter value matches with TSG3nCMP0E or a change in the TSG3nPTSI2 to TSG3nPTSI0 pins is detected. The timer output pattern is switched each time a change in the TSG3nPTSI2 to TSG3nPTSI0 pins is detected.

When TSG3nS120DCO = 1, the 18-bit counter is cleared to 00000_H when the counter value matches with TSG3nCMP0E but not cleared with a change in the TSG3nPTSI2 to TSG3nPTSI0 pins. The timer output pattern is switched to the one corresponding to the patterns of TSG3nPTSI2 to TSG3nPTSI0 pins at the next match timing of TSG3nCNTE and TSG3nCMP0E.

NOTE

PAT0T to PAT5T and PAT0B to PAT5B show PWM operation set by TSG3nCMP1E to TSG3nCMP12E, respectively.

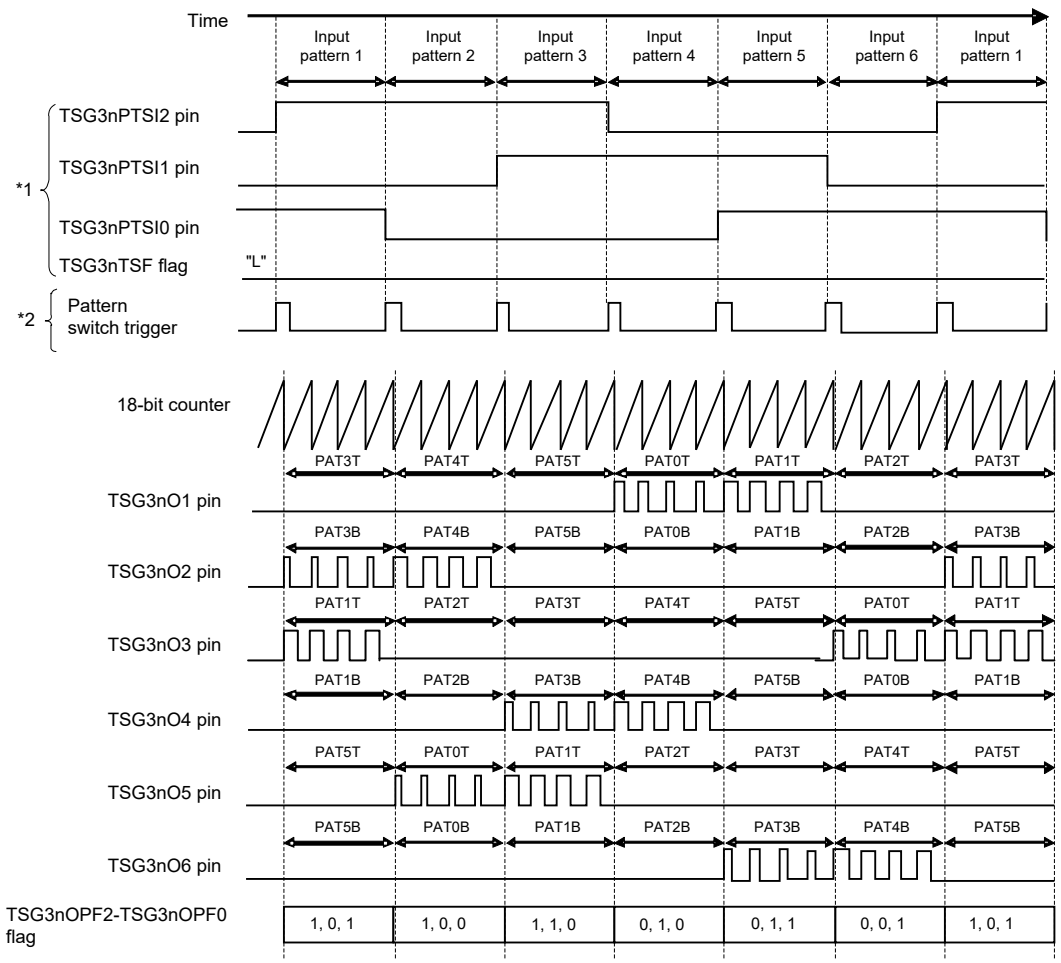


Note 1. For pattern switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0)

Note 2. For trigger switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1)

Note: TSG3nOPT0.TSG3nSOC = 0

Figure 20.88 Example of Operation in 120-DC Mode (Normal Rotation: TSG3nSTR1.TSG3nTSF = 0 and TSG3nOPT0.TSG3nIDC = 0)

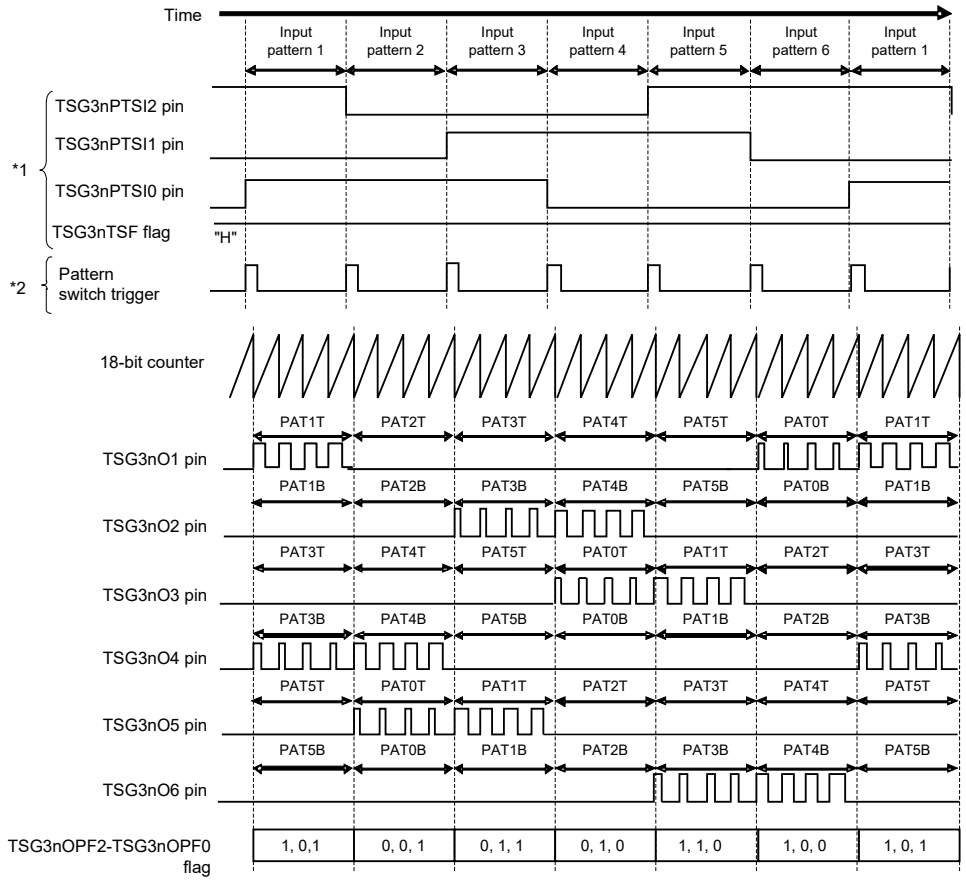


Note 1. For pattern switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0)

Note 2. For trigger switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1)

Note: TSG3nOPT0.TSG3nSOC = 0

Figure 20.89 Example of Operation in 120-DC Mode (Normal Rotation: TSG3nSTR1.TSG3nTSF = 0 and TSG3nOPT0.TSG3nIDC = 1)

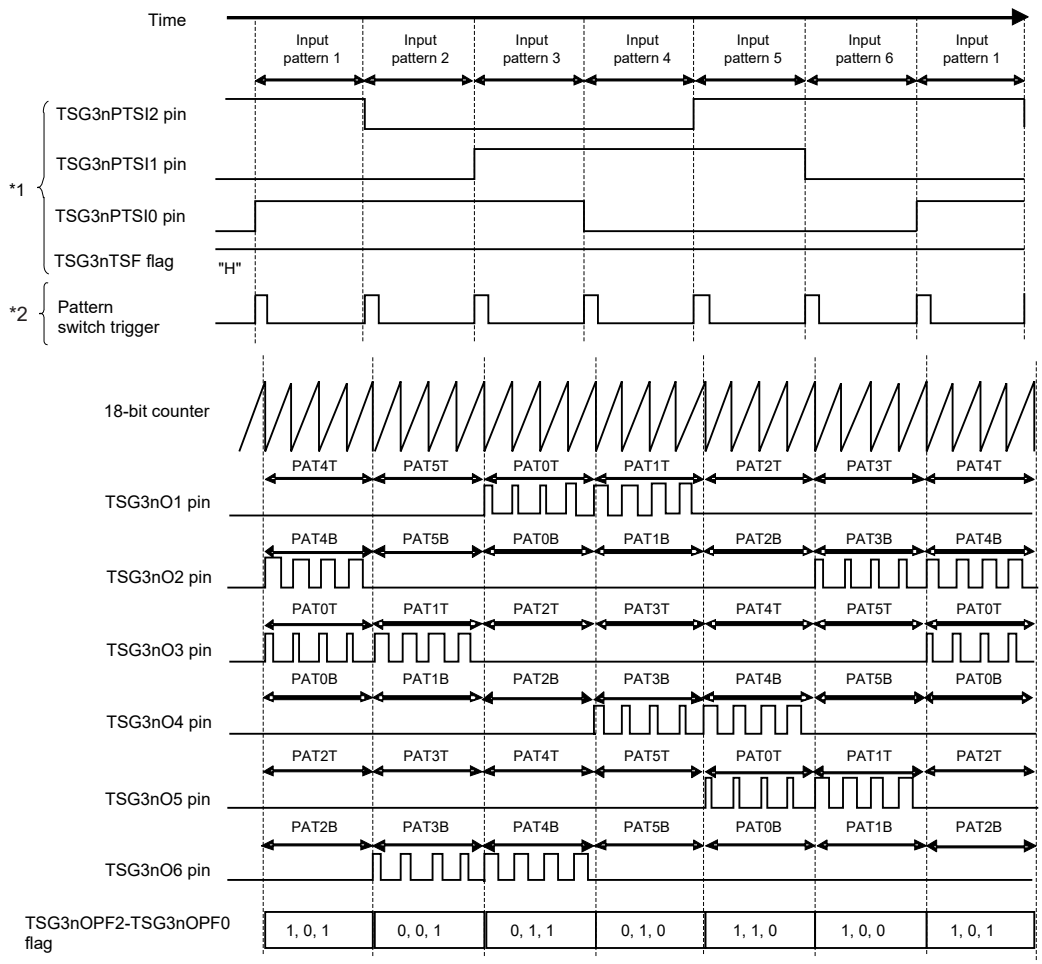


Note 1. For pattern switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0)

Note 2. For trigger switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1)

Note: TSG3nOPT0.TSG3nSOC = 0

Figure 20.90 Example of Operation in 120-DC Mode (Reverse Rotation: TSG3nSTR1.TSG3nTSF = 1 and TSG3nOPT0.TSG3nIDC = 0)



Note 1. For pattern switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0)

Note 2. For trigger switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1)

Note: TSG3nOPT0.TSG3nSOC = 0

Figure 20.91 Example of Operation in 120-DC Mode (Reverse Rotation: TSG3nSTR1.TSG3nTSF = 1 and TSG3nOPT0.TSG3nIDC = 1)

(6) List of Output Patterns in 120-DC Mode

In 120-DC mode, the output pattern is determined according to the electric current direction (TSG3nOPT0.TSG3nIDC) and the pattern order direction.

Table 20.86 Selection of Pattern Order Direction in 120-DC Mode

TSG3nOPT0			
TSG3nSTE	TSG3nPOT	TSG3nPSS	Pattern Order Direction
0	—	1	TSG3nPSC
1	0	—	TSG3nTSF (TSGnPSC only for initial setting at operation start)
1	1	1	TSG3nPSC

Table 20.87 List of Output Patterns in 120-DC Mode (1/4)

Electric current direction: normal (TSG3nIDC = 0)
Pattern order direction: normal (TSG3nTSF = 0 or TSG3nPSC = 0)

	TSG3nOPT1.TSG3nSPC2-TSG3nSPC0*1/TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
Output Pins	101	100	110	010	011	001	000	111
TSG3nO1	PAT0T	PAT1T	PAT2T	PAT3T	PAT4T	PAT5T	Low	Low
TSG3nO2	PAT0B	PAT1B	PAT2B	PAT3B	PAT4B	PAT5B	Low	Low
TSG3nO3	PAT4T	PAT5T	PAT0T	PAT1T	PAT2T	PAT3T	Low	Low
TSG3nO4	PAT4B	PAT5B	PAT0B	PAT1B	PAT2B	PAT3B	Low	Low
TSG3nO5	PAT2T	PAT3T	PAT4T	PAT5T	PAT0T	PAT1T	Low	Low
TSG3nO6	PAT2B	PAT3B	PAT4B	PAT5B	PAT0B	PAT1B	Low	Low

Table 20.87 List of Output Patterns in 120-DC Mode (2/4)

Electric current direction: reverse (TSG3nIDC = 1)
Pattern order direction: normal (TSG3nTSF = 0 or TSG3nPSC = 0)

	TSG3nOPT1.TSG3nSPC2-TSG3nSPC0*1/TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
Output Pins	101	100	110	010	011	001	000	111
TSG3nO1	PAT3T	PAT4T	PAT5T	PAT0T	PAT1T	PAT2T	Low	Low
TSG3nO2	PAT3B	PAT4B	PAT5B	PAT0B	PAT1B	PAT2B	Low	Low
TSG3nO3	PAT1T	PAT2T	PAT3T	PAT4T	PAT5T	PAT0T	Low	Low
TSG3nO4	PAT1B	PAT2B	PAT3B	PAT4B	PAT5B	PAT0B	Low	Low
TSG3nO5	PAT5T	PAT0T	PAT1T	PAT2T	PAT3T	PAT4T	Low	Low
TSG3nO6	PAT5B	PAT0B	PAT1B	PAT2B	PAT3B	PAT4B	Low	Low

Note 1. When TSG3nSPC2 to TSG3nSPC0 are written to while the values of TSG3nSPC2 to TSG3nSPC0, TSG3nSTE, and TSG3nPOT are 1, the output pattern changes. Thereafter, when a pattern switch trigger is generated on rising of the TSG3nOPCI0 and TSG3nOPCI1 signals, the output is switched according to the order of pattern switching. When TSG3nS120DCO = 0, the output is switched immediately. When TSG3nS120DCO = 1, the output is switched when the main counter (TSG3nCNTE) is matched with TSG3nCMP0E (from the next timer cycle). TSG3nSPC2 to TSG3nSPC0 remain unchanged even if the output pattern is switched.

NOTES

1. PAT0T to PAT5T: PWM output set by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, TSG3nCMP10E
2. PAT0B to PAT5B: PWM output set by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, TSG3nCMP12E

Table 20.87 List of Output Patterns in 120-DC Mode (3/4)

Electric current direction: normal (TSG3nIDC = 0)
Pattern order direction: reverse (TSG3nTSF = 1 or TSG3nPSC = 1)

	TSG3nOPT1.TSG3nSPC2-TSG3nSPC0*1/TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
Output Pins	101	100	110	010	011	001	000	111
TSG3nO1	PAT1T	PAT0T	PAT5T	PAT4T	PAT3T	PAT2T	Low	Low
TSG3nO2	PAT1B	PAT0B	PAT5B	PAT4B	PAT3B	PAT2B	Low	Low
TSG3nO3	PAT3T	PAT2T	PAT1T	PAT0T	PAT5T	PAT4T	Low	Low
TSG3nO4	PAT3B	PAT2B	PAT1B	PAT0B	PAT5B	PAT4B	Low	Low
TSG3nO5	PAT5T	PAT4T	PAT3T	PAT2T	PAT1T	PAT0T	Low	Low
TSG3nO6	PAT5B	PAT4B	PAT3B	PAT2B	PAT1B	PAT0B	Low	Low

Table 20.87 List of Output Patterns in 120-DC Mode (4/4)

Electric current direction: reverse (TSG3nIDC = 1)
Pattern order direction: reverse (TSG3nTSF = 1 or TSG3nPSC = 1)

	TSG3nOPT1.TSG3nSPC2-TSG3nSPC0*1/TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
Output Pins	101	100	110	010	011	001	000	111
TSG3nO1	PAT4T	PAT3T	PAT2T	PAT1T	PAT0T	PAT5T	Low	Low
TSG3nO2	PAT4B	PAT3B	PAT2B	PAT1B	PAT0B	PAT5B	Low	Low
TSG3nO3	PAT0T	PAT5T	PAT4T	PAT3T	PAT2T	PAT1T	Low	Low
TSG3nO4	PAT0B	PAT5B	PAT4B	PAT3B	PAT2B	PAT1B	Low	Low
TSG3nO5	PAT2T	PAT1T	PAT0T	PAT5T	PAT4T	PAT3T	Low	Low
TSG3nO6	PAT2B	PAT1B	PAT0B	PAT5B	PAT4B	PAT3B	Low	Low

Note 1. When TSG3nSPC2 to TSG3nSPC0 are written to while the values of TSG3nSPC2 to TSG3nSPC0, TSG3nSTE, and TSG3nPOT are 1, the output pattern changes. Thereafter, when a pattern switch trigger is generated on rising of the TSG3nOPCI0 and TSG3nOPCI1 signals, the output is switched according to the order of pattern switching. When TSG3nS120DCO = 0, the output is switched immediately.
When TSG3nS120DCO = 1, the output is switched when the main counter (TSG3nCNTE) is matched with TSG3nCMP0E (from the next timer cycle). TSG3nSPC2 to TSG3nSPC0 remain unchanged even if the output pattern is switched.

NOTES

1. PAT0T to PAT5T: PWM output set by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, TSG3nCMP10E
2. PAT0B to PAT5B: PWM output set by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, TSG3nCMP12E

(7) Operation Start Timing in 120-DC Mode

In 120-DC mode, when trigger switch control (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1) is used, pattern set with TSG3nOPT1.TSG3nSPC2 to TSG3nOPT1.TSG3nSPC0, TSG3nOPT0.TSG3nPSC, and TSG3nOPT0.TSG3nIDC can be output. However, when pattern switch control (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0) is used, the pattern of the TSG3nPTSI2 to TSG3nPTSI0 pins can be detected but the pattern order direction (TSG3nSTR1.TSG3nTSF) cannot be determined. Therefore, set the pattern order direction in TSG3nPSC when TSG3nTE is 0. The TSG3nPSC set value is loaded to TSG3nTSF, and the value can be used for the initial pattern setting.

- TSG3nOPT0.TSG3nSOC = 0, TSG3nPSC = 0, TSG3nPOT = 0, TSG3nIDC = 0, TSG3nSTE = 1

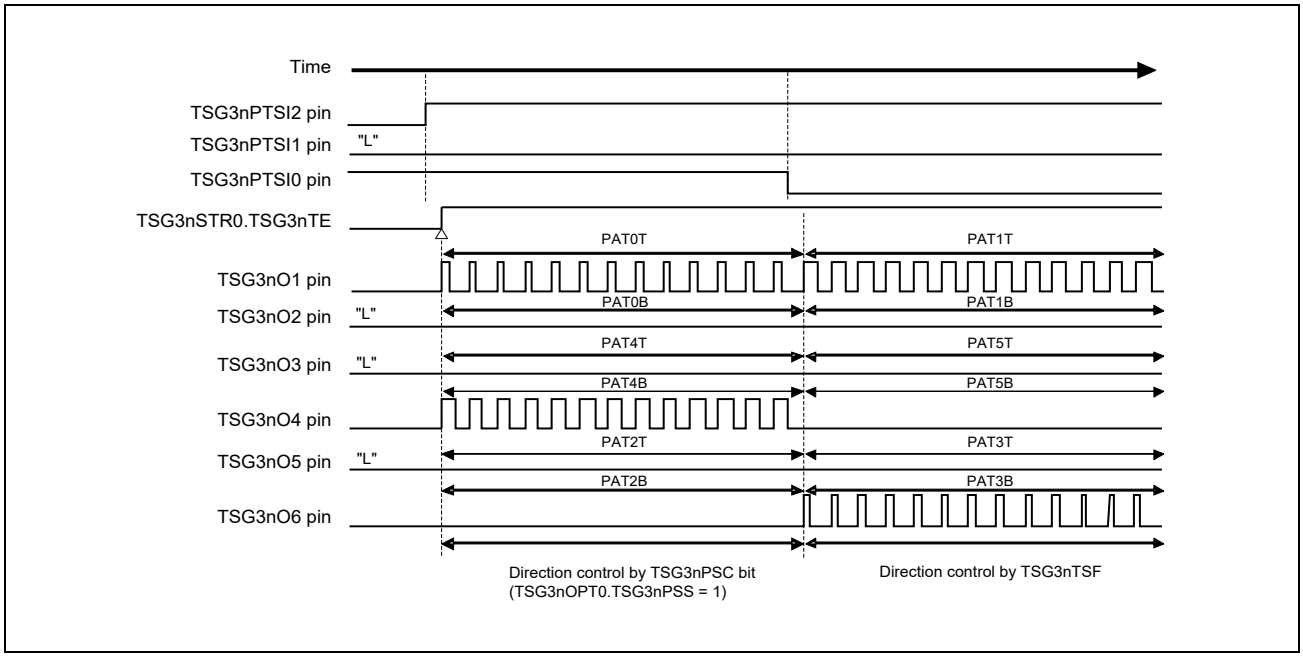


Figure 20.92 Control when Timer Output Starts in Normal Rotation (when Normal Pattern is Input)

- TSG3nOPT0.TSG3nSOC = 0, TSG3nPSC = 1, TSG3nPOT = 0, TSG3nIDC = 1, TSG3nSTE = 1

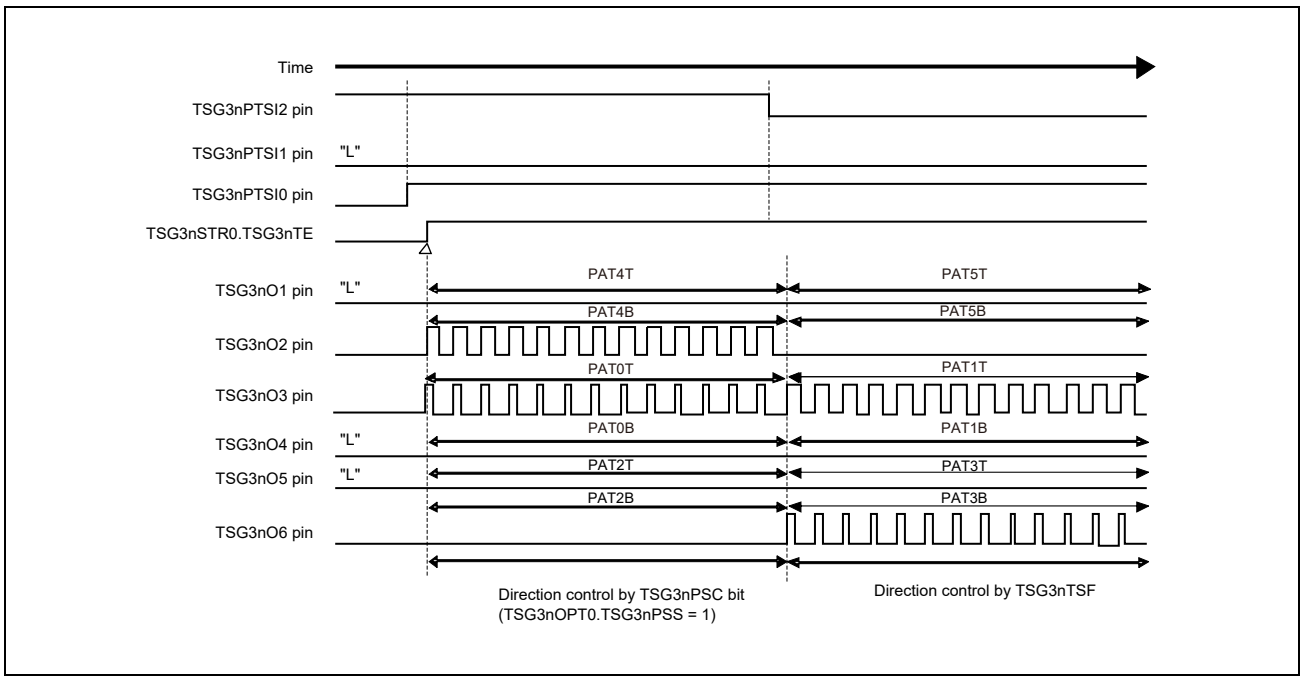


Figure 20.93 Control when Timer Output Starts in Reverse Rotation (when Normal Pattern is Input)

- TSG3nOPT0.TSG3nSOC = 0, TSG3nPSC = 0, TSG3nPOT = 0, TSG3nIDC = 0, TSG3nSTE = 1

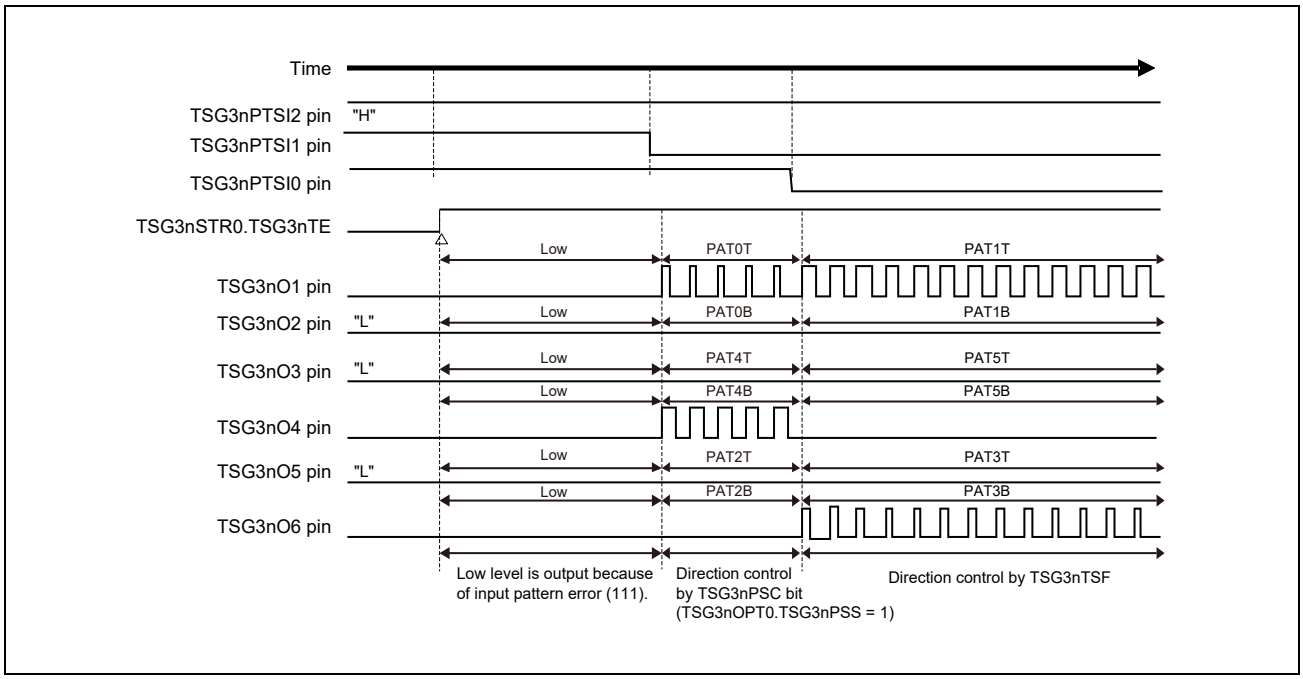


Figure 20.94 Control when Timer Output Starts in Normal Rotation (when Error Pattern is Input)

- TSG3nOPT0.TSG3nSOC = 0, TSG3nPSC = 1, TSG3nPOT = 0, TSG3nIDC = 1, TSG3nSTE = 1

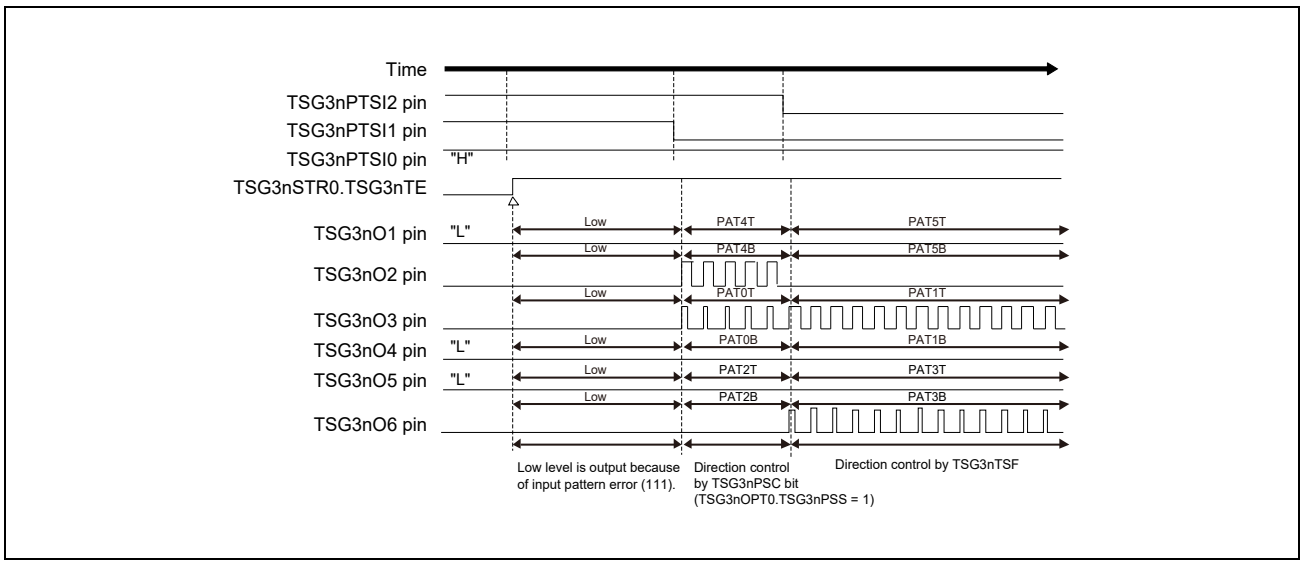


Figure 20.95 Control when Timer Output Starts in Reverse Rotation (when Error Pattern is Input)

(8) Output Switch Timing in 120-DC Mode (TSG3nS120DCO = 0)

As shown in **Figure 20.96** to **Figure 20.99**, in 120-DC mode, the external switch timing for output pattern (TSG3nOPCI0 and TSG3nOPCI1 signals, and TSG3nPTSI2 to TSG3nPTSI0 pins) is input irrespective of the 18-bit counter operation. When TSG3nS120DCO is 0, the output is switched to the new pattern by clearing the 18-bit counter using the pattern switch timing signal applied from outside.

In the pattern switch method, if a change in TSG3nPTSI2 to TSG3nPTSI0 pins occurs several times within one period, the output pattern is switched by clearing the 18-bit counter at each change. In the trigger switch method, if TSG3nOPCI0 and TSG3nOPCI1 signal trigger is input for several times within one period, the output pattern is switched by clearing the 18-bit counter each time the trigger is accepted.

If TSG3nSPC2 to TSG3nSPC0 are rewritten several times within one period, the output pattern is switched by clearing the 18-bit counter each time TSG3nSPC2 to TSG3nSPC0 are rewritten.

In case of a conflict between a rewrite to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 and TSG3nOPCI0 and TSG3nOPCI1 trigger, rewriting of TSG3nSPC2 to TSG3nSPC0 takes precedence.

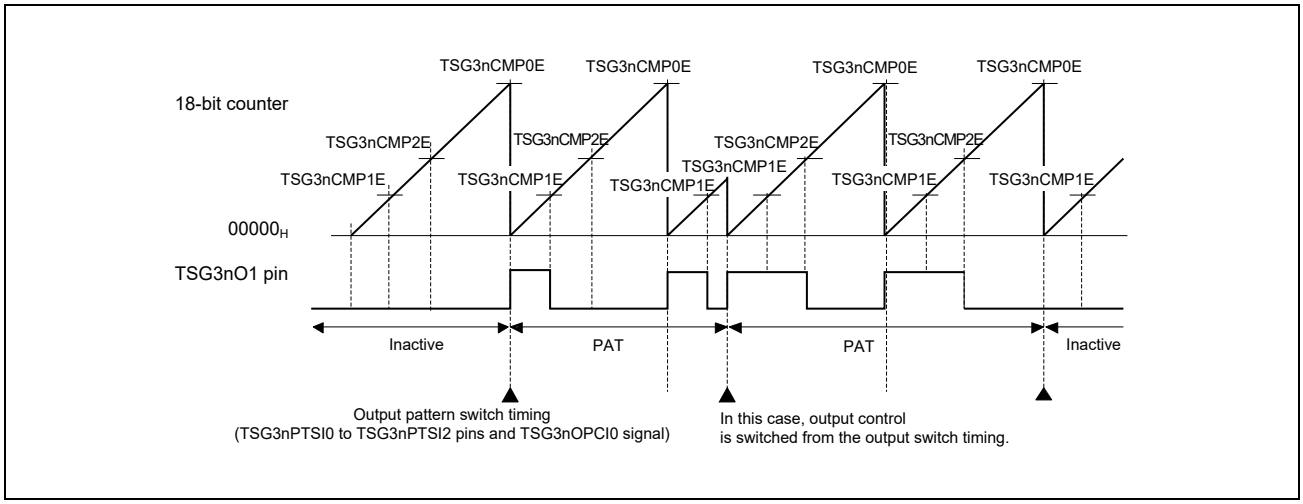
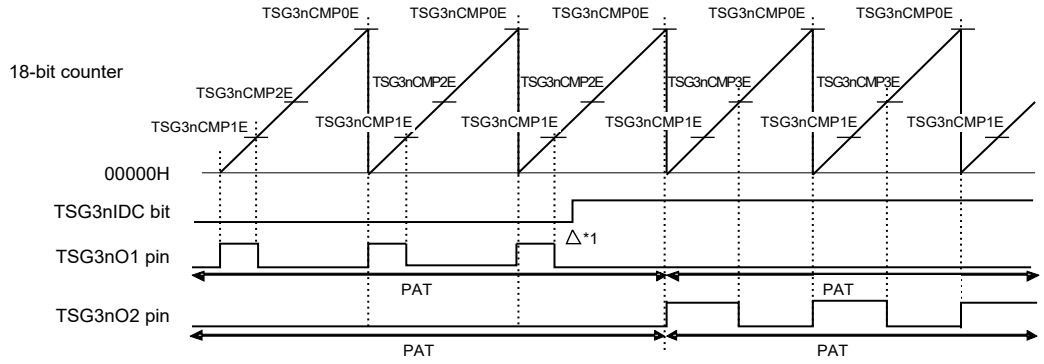


Figure 20.96 Output Switch Example (TSG3nPTSI2 to TSG3nPTSI0 Pins and TSG3nOPCI0 and TSG3nOPCI1 Signal Trigger Input)

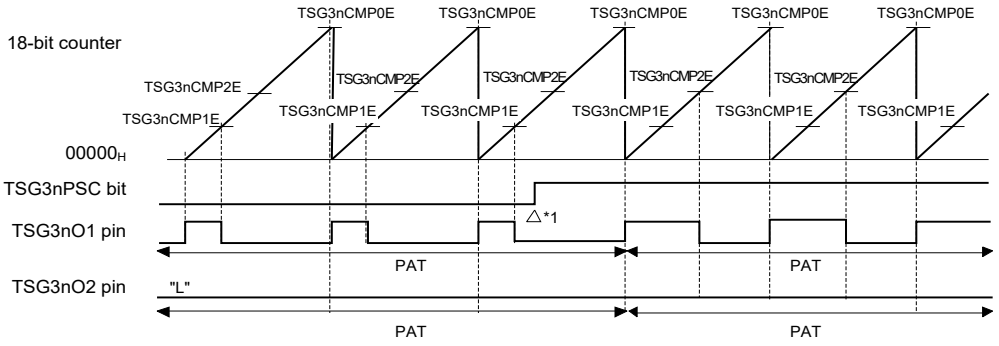


Note 1. When the pattern is switched here, output control is switched in the next period.

Note: If a change in the TSG3nPTSI2-0 pins occurs by the time next period when output control is switched by the TSG3nIDC bit, the 18-bit counter is cleared and output control is switched.

Figure 20.97 Output Switch Example (Switched by TSG3nOPT0.TSG3nIDC)

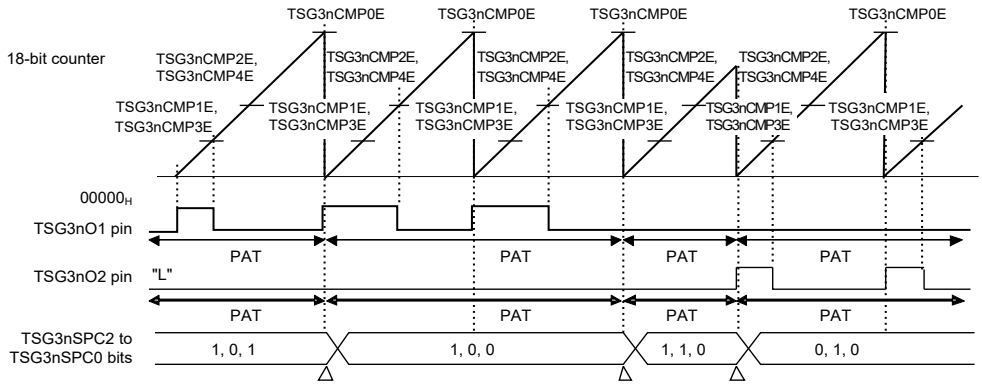
- TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1



Note 1. When the pattern is switched here, output control is switched from the next period.

Figure 20.98 Output Switch Example (Switched by TSG3nOPT0.TSG3nPSC)

- TSG3nOPT0.TSG3nSOC = 0, TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1



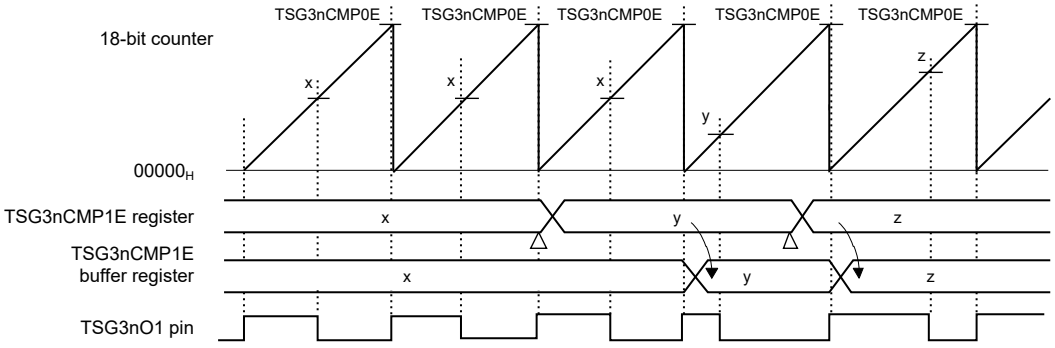
Note: Δ: Write access

Figure 20.99 Output Switch Example (Switched by TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0)

(9) Compare Register Rewrite Timing in 120-DC Mode

Example of operation when TSG3nCMP1E is reloaded (rewritten simultaneously) is shown below.

Figure 20.100 shows an output example when TSG3nCMP1E is rewritten. After TSG3nCMP1E is changed, data is not transferred to the TSG3nCMP1E buffer register (changed data is not valid) until the next reload timing; therefore, the specified output waveform can be obtained. However, do not write to TSG3nCMP1E again while the reload is suspended (period from when TSG3nCMP1E is changed to when simultaneous rewrite is executed). Be sure to read the reload request flag (TSG3nRSF) to confirm that the flag is 0, and write data to TSG3nCMP1E.



Note: Δ: Write access

Figure 20.100 Output Example when TSG3nCMP1E is Rewritten

(10) Dead Time Control in 120-DC Mode

In 120-DC mode, the dead time is controlled on falling of each phase, and the dead time is added.

The dead time set in TSG3nDTC1W is inserted on falling of the positive phase, and the dead time set in TSG3nDTC0W is inserted on falling of the inverse phase.

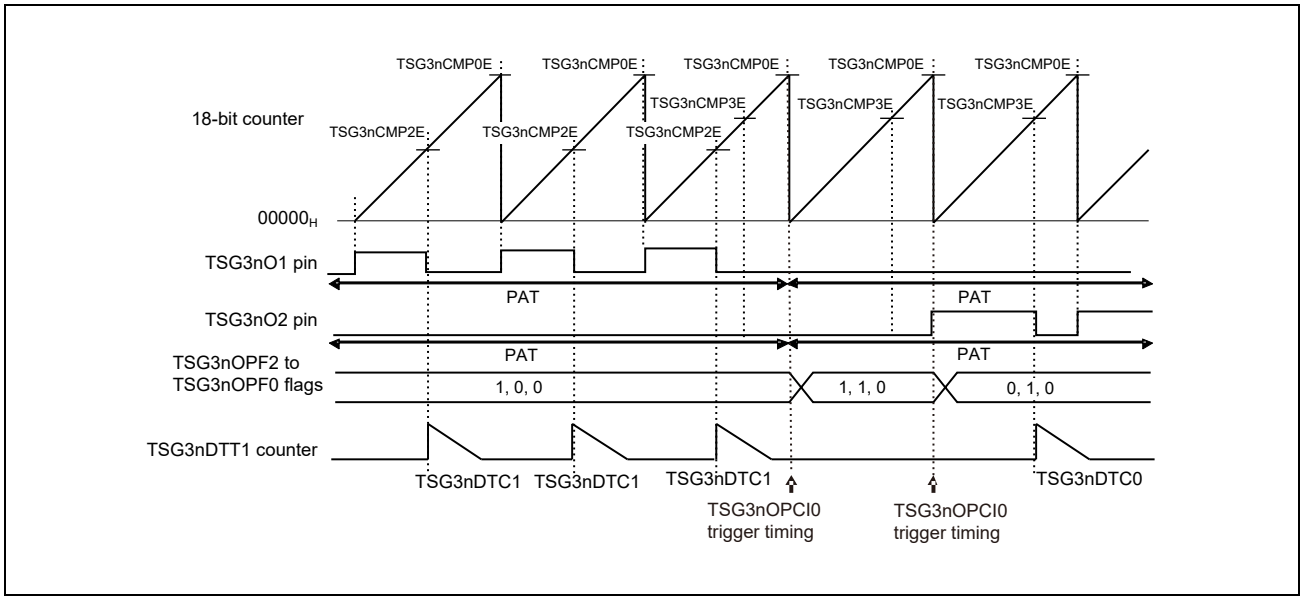


Figure 20.101 Output Switch Example

CAUTION

The dead time control method may affect the timer output. The timer output may not have the specified active level width due to the dead time control under the following conditions:

- When noise is generated on the input pattern in the pattern switch method
- When a change in the input pattern occurs earlier than the PWM period in the pattern switch method
- When TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 are changed and the output pattern is forcibly changed in the trigger switch method
- When switch method is changed
- When the current direction control bit (TSG3nOPT0.TSG3nIDC) is changed
- When the software output control function is used

(11) Output Switch in 120-DC Mode

In 120-DC mode, the output pattern can be controlled by writing values to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 when the trigger switch method (TSG3nOPT0.TSG3nSTE = 1, and TSG3nPOT = 1) is used. The dead time is secured by hardware at the switch timing.

CAUTION

When 111_B or 000_B is written to TSG3nSPC2 to TSG3nSPC0, the TSG3nO1 to TSG3nO6 pins are driven low.

(12) Operation when Noise is Generated in TSG3nPTSI2 to TSG3nPTSI0 Pins in 120-DC Mode

Input to the TSG3nPTSI2 to TSG3nPTSI0 pins is assumed to be the hall sensor signals of the brushless DC motor. Depending on the system, a noise may be generated on the TSG3nPTSI2 to TSG3nPTSI0 pins. Operation when a noise is generated is described below.

For system product design, be sure to insert a noise filter circuit between the hall sensor and the TSG3nPTSI2 to TSG3nPTSI0 pins.

Figure 20.103 shows a case when a noise is generated on the TSG3nPTSI2 to TSG3nPTSI0 pins during operation with the pattern switch method used.

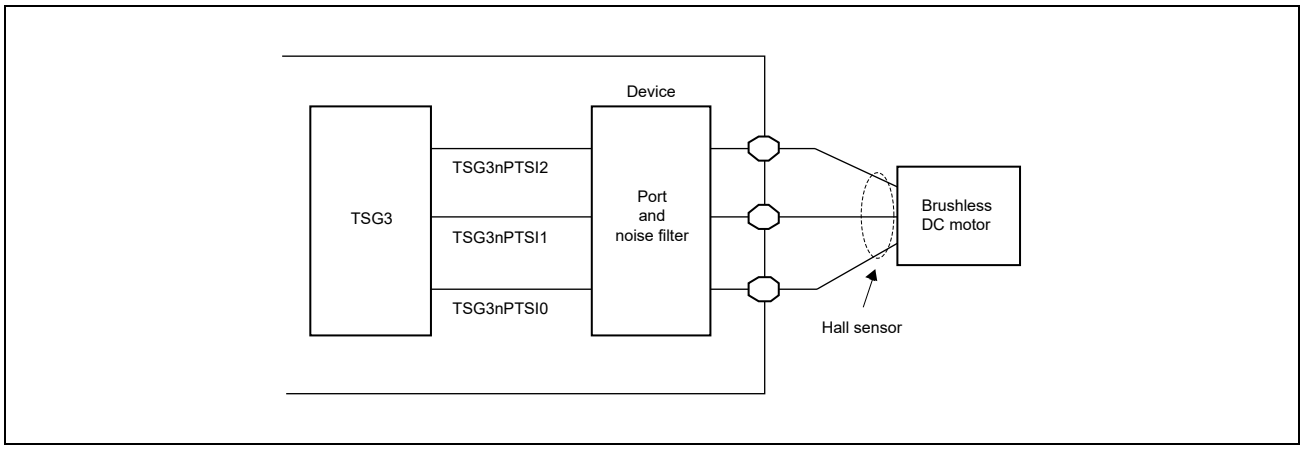


Figure 20.102 Example of Noise Filter Circuit Connection

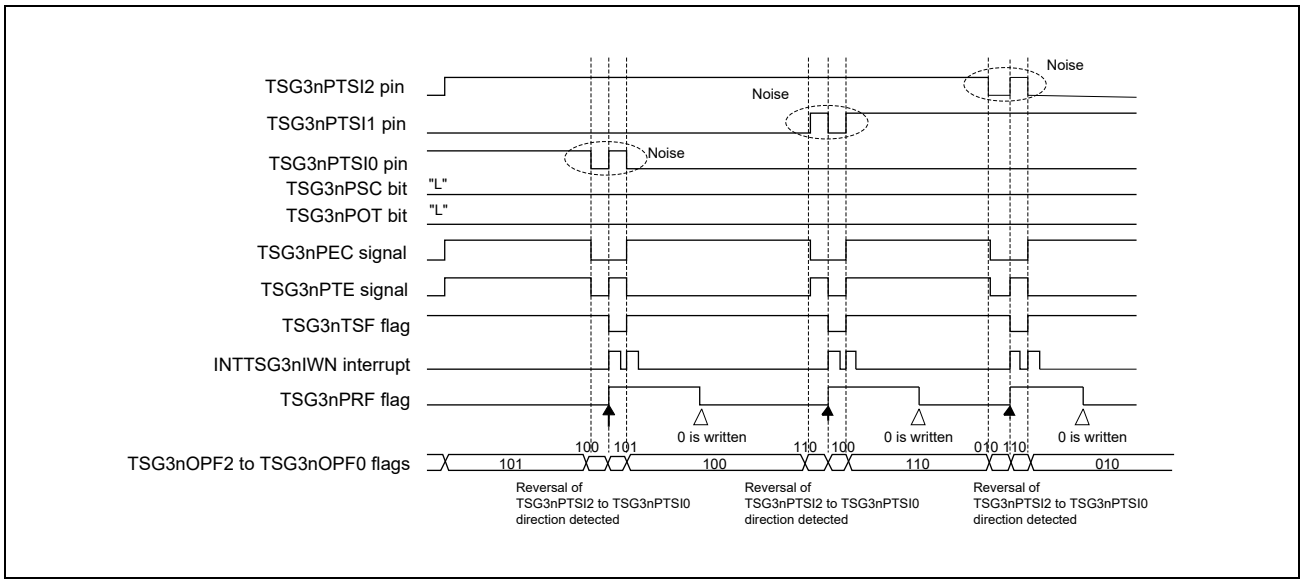


Figure 20.103 Example of Noise Generation at Level Change in TSG3nPTSI2 to TSG3nPTSI0 Pins (Pattern Switch Method)

(a) Change Timing of Input Pattern Change Detection Signal (TSG3nPTE)

The TSG3nPTE signal toggles when the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) changes.

CAUTION

Be sure to specify the pattern order direction by the TSG3nPSC bit (when the TSG3nPSS bit in the TSG3nOPT0 register is 1) in the TSG3nOPT0 register.

- When TSG3nPSC = 0:

Table 20.88 TSG3nPTE Toggle Operation when TSG3nPSC is 0

		TSG3nPTSI2-TSG3nPTSI0 Pins After Change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	—
	100	—	—	—	—	Toggle	—	—	—
	110	—	—	—	—	—	Toggle	—	—
	010	—	—	—	—	—	—	Toggle	—
	011	—	—	—	—	—	—	—	Toggle
	001	—	—	Toggle	—	—	—	—	—

- When TSG3nPSC = 1:

Table 20.89 TSG3nPTE Toggle Operation when TSG3nPSC is 1

		TSG3nPTSI2-TSG3nPTSI0Pins After Change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	—	—	—	—	Toggle
	100	—	—	Toggle	—	—	—	—	—
	110	—	—	—	Toggle	—	—	—	—
	010	—	—	—	—	Toggle	—	—	—
	011	—	—	—	—	—	Toggle	—	—
	001	—	—	—	—	—	—	Toggle	—

(b) Change Timing of Three-Phase Encode Signal (TSG3nPEC)

The TSG3nPEC signal toggles when input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) changes.

Table 20.90 TSG3nPEC Toggle Operation

		TSG3nPTSI2-TSG3nPTSI0 After Change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 Pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	Toggle
	100	—	—	Toggle	—	Toggle	—	—	—
	110	—	—	—	Toggle	—	Toggle	—	—
	010	—	—	—	—	Toggle	—	Toggle	—
	011	—	—	—	—	—	Toggle	—	Toggle
	001	—	—	Toggle	—	—	—	Toggle	—

(c) Change Timing of TSG3nO1 to TSG3nO6 Pins

- When the pattern switch method is used, the output pattern changes when the input signal of the TSG3nPTSI2 to TSG3nPTSI0 pins*[1] changes. The output is also switched when two or more pins change simultaneously.
- When the trigger switch method is used, the output pattern changes at the rising edge of the TSG3nOPCI0 and TSG3nOPCI1 signals. The output also changes when data is written to TSG3nSPC2 to TSG3nSPC0*[1] in TSG3nOPT0.

Note 1. When the input pattern changes to 000 or 111, the TSG3nO1-TSG3nO6 pins are driven low. The output pattern of TSG3nO1-TSG3nO6 changes immediately only when TSG3nS120DCO = 0. When TSG3nS120DCO = 1, the output pattern is switched when the main counter (TSG3nCNTE) is matched with TSG3nCMP0E (from the next timer cycle).

(d) Change Timing of TSG3nTSF Flag

The TSG3nTSF flag toggles when the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) changes

Table 20.91 Setting and Clearing of TSG3nTSF

		TSG3nPTSI2-TSG3nPTSI0 Pins After Change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 Pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	0	—	—	—	1
	100	—	—	1	—	0	—	—	—
	110	—	—	—	1	—	0	—	—
	010	—	—	—	—	1	—	0	—
	011	—	—	—	—	—	1	—	0
	001	—	—	0	—	—	—	1	—

(e) Set Timing of TSG3nNDF Flag

This flag is set when two or more pins of the TSG3nPTSI2 to TSG3nPTSI0 pins change simultaneously, and cleared when 1 is written to the TSG3nNDR bit. The TSG3nNDF flag is valid when 1 is set to the TSG3nNDC bit.

(f) Set Timing of TSG3nPRF Flag

This flag is set when the TSG3nTSF flag changes, and cleared when 1 is written to the TSG3nPRR bit. The TSG3nPRF flag is valid when 1 is set to the TSG3nPRC bit.

(g) Set Timing of TSG3nPEF Flag

This flag is set when 000 or 111 is input to the TSG3nPTSI2 to TSG3nPTSI0 pins, and cleared when 1 is written to the TSG3nPER bit. The TSG3nPEF flag is valid when 1 is set to the TSG3nPEC bit.

(13) Basic Control Flow in 120-DC Mode

In 120-DC mode, there are eight control states as listed in **Table 20.92**.

When TSG3nOPT0.TSG3nSTE = 1 and TSG3nPOT = 0, the pattern switch method is used for 120-DC control.

This is defined as fixed phase control. The fixed phase control should be performed considering the factors such as delay from the hall sensor and delay from sensor level detection to timer output. However, acceleration or deceleration can be performed simply by changing the PWM duty.

When TSG3nOPT0.TSG3nSTE = 1 and TSG3nPOT = 1, the trigger switch method is selected for 120-DC control.

This is defined as variable phase control. With the variable phase control, the timer output pattern is set prior to the hall sensor; therefore, acceleration or deceleration control according to the phase difference can be performed. However, control is more complex than the fixed phase control because offset width with respect to the hall sensor and the predicted value with respect to the hall sensor should be considered. For details, **Section 24.2.3.11, Three-Phase Pulse Input Control Function**.

When TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1, and TSG3nPSS = 1, the pattern order direction of the motor can be set with the TSG3nPSC bit in the TSG3nOPT0 register. Set TSG3nPSC to 0 to set normal rotation, and 1 to set reverse rotation.

The TSG3nIDC bit in the TSG3nOPT0 register sets the electric current direction. If the same value as the rotation direction of the motor (TSG3nPSC set value) is set, acceleration control is set. If the different value from the rotation direction of the motor is set, deceleration control is set.

Table 20.92 Timer Control Status

Status	TSG3nPSC in TSG3nOPT0	TSG3nTSF in TSG3nSTR1	TSG3nIDC in TSG3nOPT0	TSG3nPOT in TSG3nOPT0	Control
A	—	0	0	0	Normal rotation, acceleration, and fixed phase
B	0	—	0	1	Normal rotation, acceleration, and variable phase
C	0	—	1	1	Normal rotation, deceleration, and variable phase
D	—	0	1	0	Normal rotation, deceleration, and fixed phase
E	—	1	1	0	Reverse rotation, acceleration, and fixed phase
F	1	—	1	1	Reverse rotation, acceleration, and variable phase
G	1	—	0	1	Reverse rotation, deceleration, and variable phase
H	—	1	0	0	Reverse rotation, deceleration, and fixed phase

Generally, the state, when the motor rotation stops, is assumed to be a state of the start and the control begins. First the fixed phase control is used to rotate the motor from the stopped state. Afterwards, to accelerate the motor speed to the fast rotation, the variable phase control is switched on. In combination with encoder timer (ENCA) and the variable phase control, the timer output is changed according to timing that is earlier than the change point of the hall sensors (leading).

To decelerate the motor speed from fast rotation, the direction of control is switched to deceleration control by rewriting only TSG3nIDC in TSG3nOPT0. When the rotation count can be reduced to low-speed rotation, the rotation can be placed in the stopped state by decreasing the PWM duty.

State transition is shown in **Figure 20.104** and **Figure 20.105**.

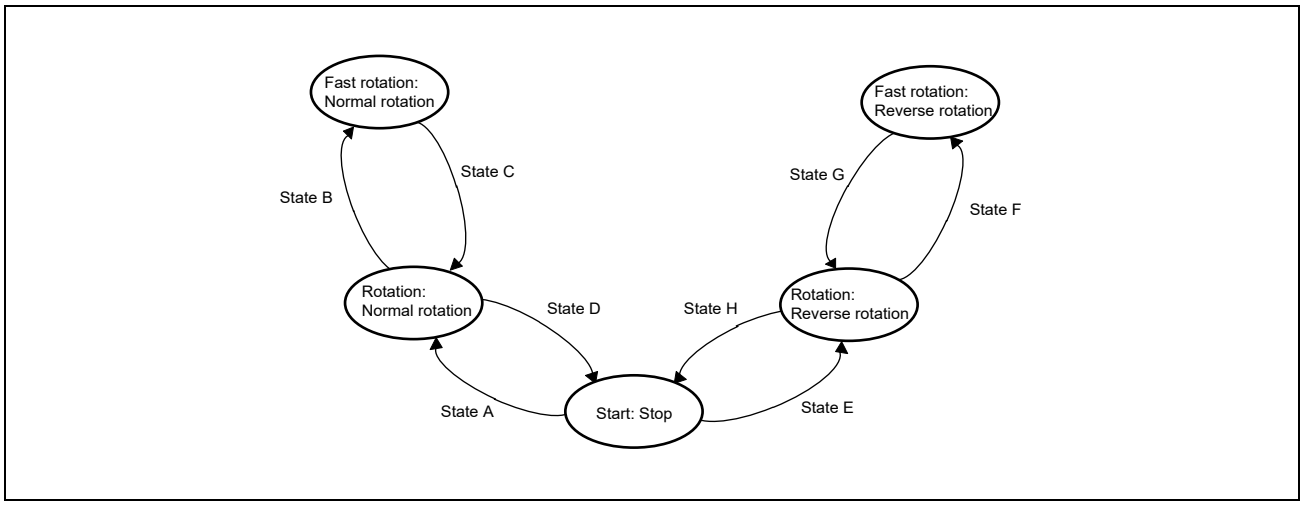
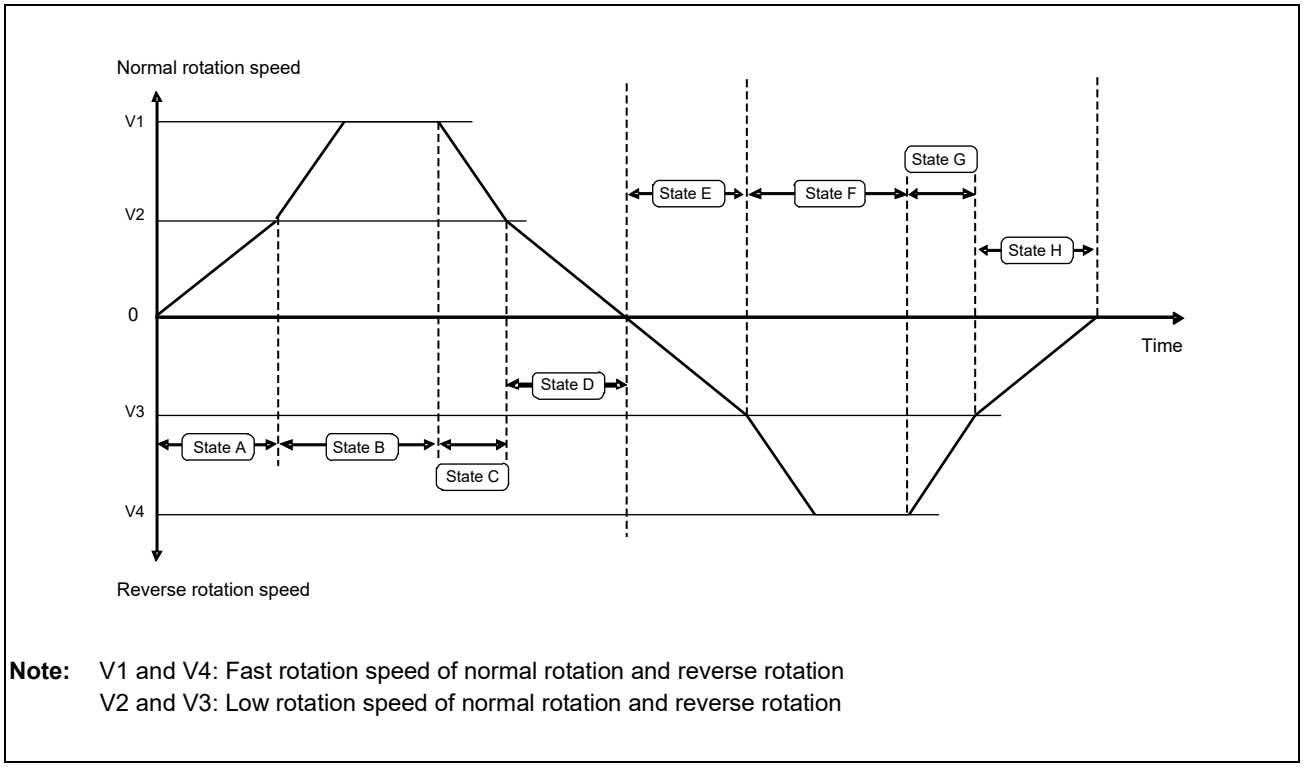


Figure 20.104 State Transition Diagram



Note: V1 and V4: Fast rotation speed of normal rotation and reverse rotation
V2 and V3: Low rotation speed of normal rotation and reverse rotation

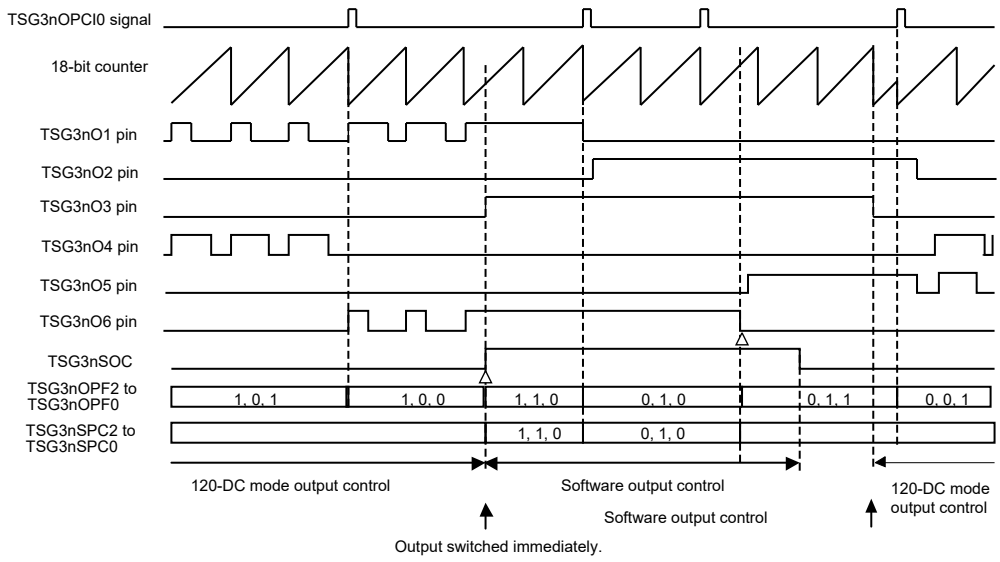
Figure 20.105 V1 and V4: Fast rotation speed of normal rotation and reverse rotation V2 and V3: Low rotation speed of normal rotation and reverse rotation

(14) Software Output Control Function in 120-DC Mode

TSG3nOPT0.TSG3nSOC and TSG3nIDC, and TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 are used in 120-DC mode for timer output control by software.

As shown in **Figure 20.106**, the output control is switched immediately when TSG3nSOC is set to 1. If the dead time is set, the period of the dead time period is guaranteed. After that, to switch the output control from software output control to 120-DC control, set TSG3nSOC to 0. At this timing, output control is retained. When the reload timing is generated, output control is switched to 120-DC mode.

For details on software output control function, see **20.4.7.10, Software** Output Control Function.



Note: Δ: Write access

Figure 20.106 Example of Switching from 120-DC Mode to Software Output Control Function

(a) Procedure for Software Output Control

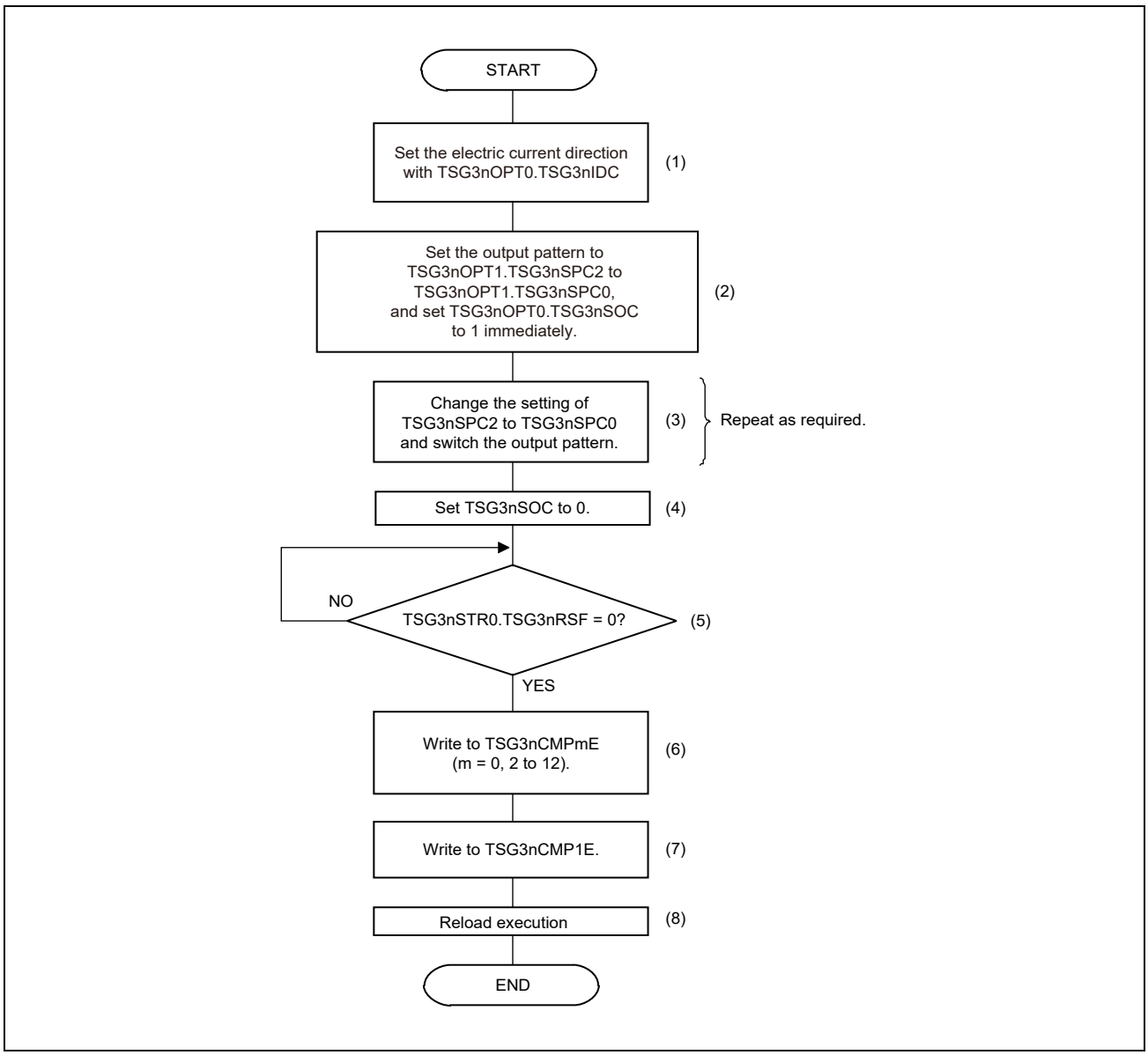


Figure 20.107 Process Flow of Software Output Control

The procedure for software output control is described below.

(1) Set the TSG3nIDC bit. The phase of the timer output with TSG3nIDC = 0 is different by 180 degrees from that with TSG3nIDC = 1. When this bit is rewritten with the software output control function, the output pattern will change according to the new setting at the next timer cycle. However, if the period match occurs before step (2), the output pattern of 120-DC control changes. So, schedule so as to prevent the period match from occurring before step (2).

(2) Set the output pattern to TSG3nSPC2 to TSG3nSPC0. To enable software output control, set TSG3nSOC to 1 simultaneously.

(3) Change the output pattern setting of TSG3nSPC2 to TSG3nSPC0 to change the timer output.

(4) Confirm that the reload request flag (TSG3nRSF) = 0. If TSG3nRFS = 1, do not proceed to the following step until TSG3nRSF = 0.

(5) Setting TSG3nSOC = 0 starts releasing of the software control (it is not released here yet).

(6) Make necessary settings of the compare registers that will be used after the software output control is released. Proceed to the following step when the register settings are not required. Here, change the registers with the reload function.

(7) Write to TSG3nCMP1E to start reloading.

(8) Reload is executed and software output is released.

CAUTION

Execute reload after executing steps (4) to (7). When reload cannot be executed, the software output cannot be released.

20.4.7.7 HSP-PWM Mode (High-Accuracy Shifted-Pulse - Pulse Width Modulation Mode)

Overview

In this mode, the 18-bit counter and the 18-bit compare register are used to generate a high accuracy sawtooth waveform PWM signal.

Prerequisites

Set the PWM period to TSG3nCMP0E.

Set PWM output width with the TSG3nHSPCMUE, TSG3nHSPCMVE, and TSG3nHSPCMWE registers. Set PWM shift width with the TSG3nHSPSHUE, TSG3nHSPSHVE, and TSG3nHSPSHWE registers. Set dead time with the TSG3nDTC0W and TSG3nDTC1W registers. The values set in these registers are immediately reflected in the corresponding TSG3nCMPmE (m = 1 to 12) based on the calculation described later.

Functional description

Set the PWM period, the duty cycle, and the PWM sift width. Then set the PWM output width. Counting up begins when TSG3nTRG0.TSG3nTS is set to 1.

The 18-bit counter starts counting from 00000_H and is cleared by the match with TSG3nCMP0E.

During counting, a compare match interrupt (INTTSG3nI0 to INTTSG3nI12) is generated by the match of the buffer register TSG3nCMP0E to TSG3nCMP12E with the 18-bit counter.

NOTE

The HSP-PWM mode is set when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 100_B.

(1) Basic Timing Chart

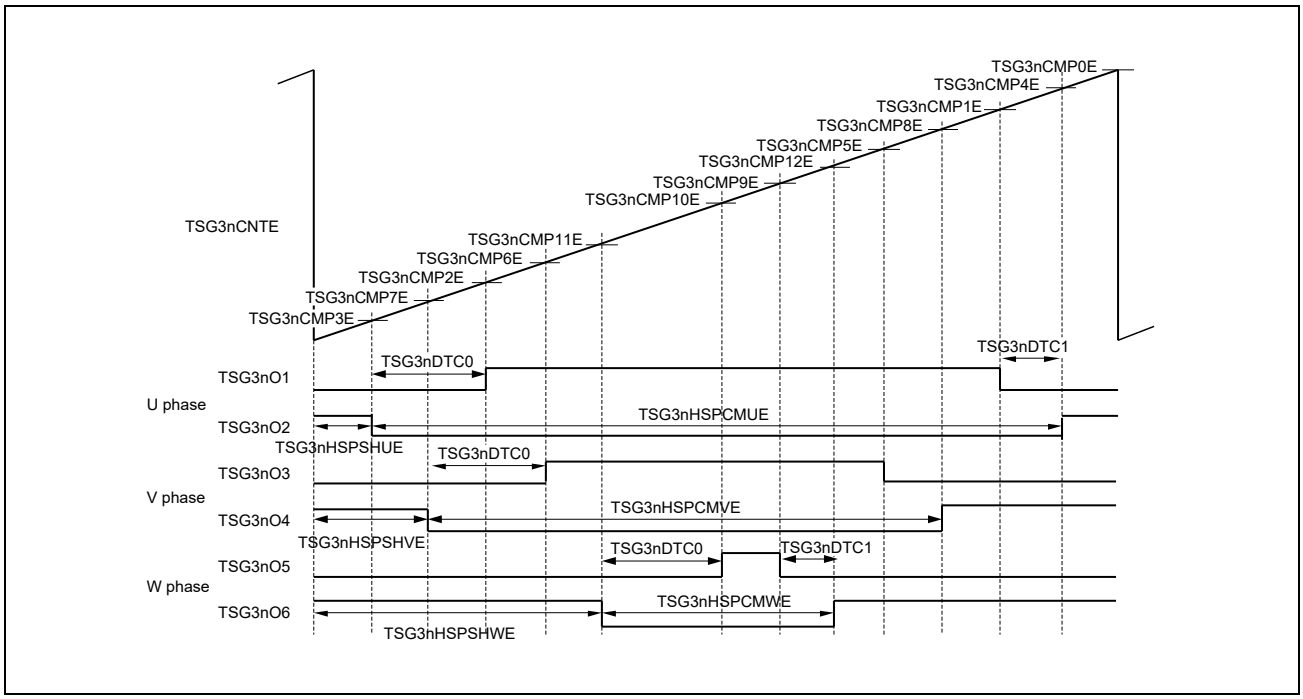
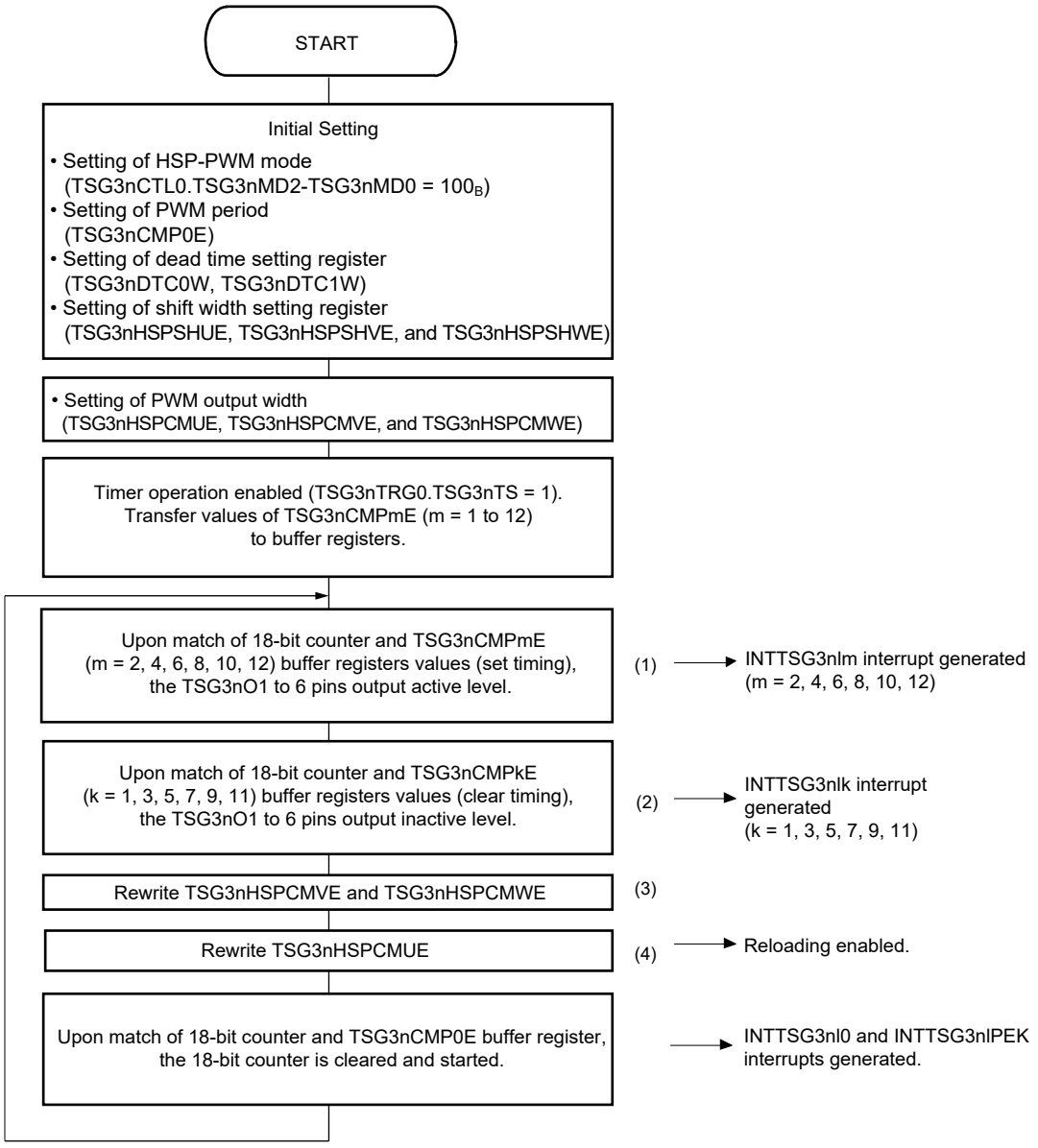


Figure 20.108 Basic Timing in HSP-PWM Mode



Note: The timing may be different from the orders described above but be sure to execute (4) after (3).

Figure 20.109 Basic Operation Flow in HSP-PWM Mode

CAUTIONS

1. When changing the settings for PWM output width in the TSG3nHSPCMUE, VE, and WE registers during operation, set the TSG3nHSPCMUE register last. When only the settings for the V phase and W phase are changed but the PWM output width in the U phase is not, the existing value should be written back to the TSG3nHSPCMUE register.
2. After changing the PWM cycle by using the TSG3nCMP0E register, the PWM output width settings in the TSG3nHSPCMUE, VE, and WE registers must be remade. Changing the settings for the PWM shift width in the TSG3nHSPSHUE, VE, and WE registers and for the PWM cycle in the TSG3nCMP0E register at the same time is not allowed.
3. Remake the settings for PWM output width in the TSG3nHSPCMUE, VE, and WE registers after changing the settings for PWM shift width in the TSG3nHSPSHUE, VE, and WE registers.

(2) List of Operation in HSP-PWM

Table 20.93 Counter Function in HSP-PWM

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger
	Clear	Compare match of TSG3nCMP0E buffer register and 18-bit counter
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 20.94 Functions of Compare Registers, Sift Width Setting Register, and Dead Time Setting Register in HSP-PWM

Register	Rewriting Method	Rewrite During Operation	Function
TSG3nCMP0E	Reload	Possible	Setting period
TSG3nHSPCMUE	Reload	Possible	PWM control for U phase
TSG3nHSPSHUE			
TSG3nHSPCMVE	Reload	Possible	PWM control for V phase
TSG3nHSPSHVE			
TSG3nHSPCMWE	Reload	Possible	PWM control for W phase
TSG3nHSPSHWE			
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload	Possible	Diagnostic output or A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Prohibited	Dead time

Table 20.95 Timer Output in HSP-PWM Mode

Pins	Function
TSG3nO1	PWM output by compare match of the TSG3nCMP1E buffer register (clear timing) or the TSG3nCMP2E buffer register (set timing) with the 18-bit counter.
TSG3nO2	PWM output by compare match of the TSG3nCMP3E buffer register (clear timing) or the TSG3nCMP4E buffer register (set timing) with the 18-bit counter.
TSG3nO3	PWM output by compare match of the TSG3nCMP5E buffer register (clear timing) or the TSG3nCMP6E buffer register (set timing) with the 18-bit counter.
TSG3nO4	PWM output by compare match of the TSG3nCMP7E buffer register (clear timing) or the TSG3nCMP8E buffer register (set timing) with the 18-bit counter.
TSG3nO5	PWM output by compare match of the TSG3nCMP9E buffer register (clear timing) or the TSG3nCMP10E buffer register (set timing) with the 18-bit counter.
TSG3nO6	PWM output by compare match of the TSG3nCMP11E buffer register (clear timing) or the TSG3nCMP12E buffer register (set timing) with the 18-bit counter.
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger

Table 20.96 Interrupt Request in HSP-PWM Mode

Interrupt	Function
INTTSG3nIm (m = 0 to 12)	Compare match of the TSG3nCMPmE buffer register with 18-bit counter
INTTSG3nIER	Error
INTTSG3nIVLY	—
INTTSG3nIPEK	Peak interrupt (generated at the same timing as INTTSG3nI0)
INTTSG3nIWN	Error

Table 20.97 Compare Match Timing in HSP-PWM Mode

Compare Match	Function
TSG3nCMP0E	When 18-bit counter changes from TSG3nCMP0E to 00000_H
TSG3nCMPmE (m = 1 to 12)	After match of 18-bit counter and TSG3nCMPmE

(3) Various Settings of HSP-PWM Mode

Mode setting

HSP-PWM mode is entered by setting TSG3nCTL0.TSG3nMD2-TSG3nMD0 to 100_B.

Setting timer output

The output pins TSG3nO1-6 are controlled by setting TSG3nIOC0, TSG3nIOC2, and TSG3nIOC3.

The TSG3nO7 pin provides output pulse as diagnostic output or A/D conversion trigger. The pin should be set as necessary.

Enabling error interrupt generation

Error interrupt (INTTSG3nIER) due to the detection of the simultaneous active state of positive and inverse phases is enabled by setting TSG3nIOC1.TSG3nEOC to 1. For details, see **Section 20.4.6 Error/Warning Interrupt**.

Setting rewriting timing of register with reload function

This function is only available in reload mode. Set TSG3nCTL3.TSG3nRMC to 0.

Setting A/D conversion trigger output

The A/D conversion trigger 0 (TSG3nADTRG0 signal) is set with TSG3nCTL5.TSG3nAT09-TSG3nAT00.

TSG3nAT09-TSG3n00 is used to enable or disable the A/D conversion trigger output on timing match of TSG3nDCMP2E-0E with 18-bit counter (up count).

TSG3nCTL6.TSG3nAT19-TSG3nAT10 is used to set the A/D conversion trigger 1 (TSG3nADTRG1 signal).

To set the match timing of the 18-bit counter and TSG3nDCMP2E-TSG3nDCMP0E, set the compare value to each register.

The skipping function can be used for TSG3nADTRG0, TSG3nADTRG1 signals. TSG3nACC01 and TSG3nACC00 of TSG3nCTL5, and TSG3nACC11 and TSG3nACC10 of TSG3nCTL6 can be used to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTIONS

1. Be sure to set TSG3nCTL5, TSG3nCTL6, and TSG3nDCMP2E-0E correctly when the timing pulse of A/D conversion trigger is output to TSG3nO7.
2. In HSP-PWM mode, no trough interrupt (INTTSG3nIVLY) is generated. Therefore, SG3nCTL5.TSG3nAT00 and TSG3nCTL6.TSG3nAT10 must be set to 0.
3. In HSP-PWM mode, the 18-bit sub-counter does not operate. Therefore, TSG3nCTL5.TSG3nAT09 and TSG3nAT08, and TSG3nCTL6.TSG3nAT19 and TSG3nAT18 must be set to 0.
4. In HSP-PWM mode, down counting by the 18-bit counter is not generated. Therefore, TSG3nCTL5.TSG3nAT07, TSG3nAT05, and TSG3nAT03, and TSG3nCTL6.TSG3nAT17, TSG3nAT15, and TSG3nAT13 should be set to 0.

Setting PWM period

Set the PWM period with TSG3nCMP0E according to the following expression:

PCLK × (TSG3nCMP0E + 1)

Setting PWM output width

The PWM output width is set with TSG3nHSPCMUE, TSG3nHSPCMVE, and TSG3nHSPCMWE (TSG3nCMP1E-12E).

Satisfy the following requirements when setting the TSG3nHSPCMUE, TSG3nHSPCMVE, and TSG3nHSPCMWE registers:

$0 \leq$ TSG3nHSPCMUE, TSG3nHSPCMVE, TSG3nHSPCMWE $\leq$ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1

Set the PWM output width after setting TSG3nCMP0E, TSG3nHSPSHUE, TSG3nHSPSHVE, TSG3nHSPSHWE, TSG3nDTC0, and TSG3nDTC1.

Setting of PWM sift width

PWM shift width is set with TSG3nHSPSHUE, TSG3nHSPSHVE, and TSG3nHSPSHWE.

Satisfy the following requirements when setting TSG3nHSPSHUE, TSG3nHSPSHVE, and TSG3nHSPSHWE.

TSG3nHSPSHUE, TSG3nHSPSHVE, TSG3nHSPSHWE $\leq$ TSG3nCMP0E

Setting dead time

The dead time can be set with TSG3nDTC0 and TSG3nDTC1.

PCLK × TSG3nDTC0

PCLK × TSG3nDTC1

TSG3nDTC0 can set the time between a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the inactive state and a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the active state.

TSG3nDTC1 can set the time between a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the inactive state and a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the active state.

Satisfy the following requirements when setting TSG3nDTC0 and TSG3nDTC 1.

(TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1) < $3FFFF_H$

TSG3nCMP0E > 3 × TSG3nDTC0

TSG3nCMP0E > 3 × TSG3nDTC1

CAUTION

Do not modify the settings of TSG3nDTC0 and TSG3nDTC1 during timer operation in HSP-PWM (TSG3nTE = 1). Set TSG3nDTC0 and TSG3nDTC1 while TSG3nTE = 0. Set dead time in HSP-PWM mode. Do not set 0 to TSG3nDTC0 and TSG3nDTC1.

Settings for operation in HSP-PWM Mode

Use HSP-PWM mode with the following settings of control registers and bits.

Do not modify the settings during operation (TSG3nTE = 1).

Table 20.98 Setting Prohibited in HSP-PWM Mode

Bit Name	Setting Value	Description
TSG3nCTL3.TSG3nRMC	0	Available only in reload mode.
TSG3nIOC3.TSG3nTOL6-1	000000_B	Logical inverse of set/clear of PWM is prohibited (HSP-PWM mode limitation)
TSG3nOPT0.TSG3nSOC	0	Switching to software control function is prohibited.
TSG3nOPT0.TSG3nSTE	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPOT	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPSS	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nIDC	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPSC	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT1.TSG3nSPC2-0	000_B	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nPAT0W	0000000_H	Operation setting of 120-DC mode (value after reset).
TSG3nPAT1W	0000000_H	Operation setting of 120-DC mode (value after reset).

CAUTION

In HSP-PWM mode, any setting should not be made directly to the TSG3nCMPmE register (m = 1 to 12).

The PWM output width and the PWM shift width should be set with TSG3nHSPCMUE, VE, and WE registers and TSG3nHSPSHUE, VE, and WE registers.

20.4.7.8 Compare Register Set Value in HSP-PWM Mode

In HSP-PWM mode, the PWM output width is set with TSG3nHSPCMUE, VE and WE.

With a write access to TSG3nHSPCMUE, VE, and WE, TSG3 calculates and sets the values to TSG3nCMP1E-12E based on the values written to the followings:

- TSG3nCMP0E (PWM period setting)
- TSG3nDTC0 (dead time setting 0)
- TSG3nDTC1 (dead time setting 1)
- TSG3nHSPSHUE/VE/WE (PWM shift width setting)
- TSG3nHSPCMUE/VE/WE (PWM output width setting)

The high accuracy PWM is realized by these values.

The algorithm to set compare register values are listed in the following table.

Table 20.99 Algorithm for Compare Setting in HSP-PWM Mode

Value Set in HSPCMUE					CMP4E	CMP3E	CMP2E	CMP1E
HSPCMUE = 0					if (HSPSHUE = 0) 0 else HSPCMUE – 1 + HSPSHUE	CMP0E + 1	0	0
0	<	HSPCMUE	≤	DTC0 + DTC1	HSPCMUE – 1 + HSPSHUE	if (HSPSHUE = 0) CMP0E else HSPSHUE – 1	0	0
DTC0 + DTC1	<	HSPCMUE	≤	CMP0E	HSPCMUE – 1 + HSPSHUE	if (HSPSHUE = 0) CMP0E else HSPSHUE – 1	DTC0 – 1 + HSPSHUE	HSPCMUE – DTC1 – 1 + HSPSHUE
CMP0E	<	HSPCMUE	≤	CMP0E + DTC1 + 1	0	0	DTC0 – 1 + HSPSHUE	HSPCMUE – DTC1 – 1 + HSPSHUE
CMP0E + DTC1 + 1	<	HSPCMUE	<	CMP0E + DTC0 + DTC1 + 1	0	0	DTC0 – 1 + HSPSHUE	HSPCMUE – CMP0E – DTC1 – 2 + HSPSHUE
HSPCMUE = CMP0E + DTC0 + DTC1 + 1					0	0	DTC0 – 1 + HSPSHUE	CMP0E + 1

: For the colored sells, subtract CMP0E + 1 when the calculated result is greater than CMP0E.

NOTE

"TSG3n" is omitted from the register names used in the calculation.

20.4.7.9 Timer Output Operation in HSP-PWM Mode

TSG3nO1-6 output is set by compare match of TSG3nCNTE with each TSG3nCMP2E, 4E, 6E, 8E, 10E, and 12E, and cleared by compare match of TSG3nCNTE with each TSG3nCMP1E, 3E, 5E, 7E, 9E, and 11E.

When PWM output width is set in TSG3nHSPCMUE, VE, and WE, the value is set in the TSG3nCMP1E-12E registers based on the calculation described in **Section 20.4.7.8, Compare Register Set Value in HSP-PWM Mode** that enables a high accuracy PWM output from 0% to 100%.

PWM output timing can be shifted by setting desired width to TSG3nHSPSHUE, VE, WE.

(1) When TSG3nHSPCMUE, VE, and WE (PWM output width setting) set value is 0

When 0 is set to TSG3nHSPCMUE (U phase PWM output width setting), 0 is set to TSG3nCMP1E, 2E, and 4E, and "TSG3nCMP0E+1" is set to TSG3nCMP3E.

Setting of TSG3nO1 by the match of TSG3nCNTE with TSG3nCMP2E and clearing by the match with TSG3nCMP1E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO1 is fixed to inactive.

Setting of TSG3nO2 by the match of TSG3nCNTE with TSG3nCMP4E occurs when TSG3nCNTE = 0.

"TSG3nCMP0E + 1" is set to TSG3nCMP3E. Here, with no match of TSG3nCNTE with TSG3nCMP3E occurs, TSG3nO2 is fixed to active.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when 0 is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

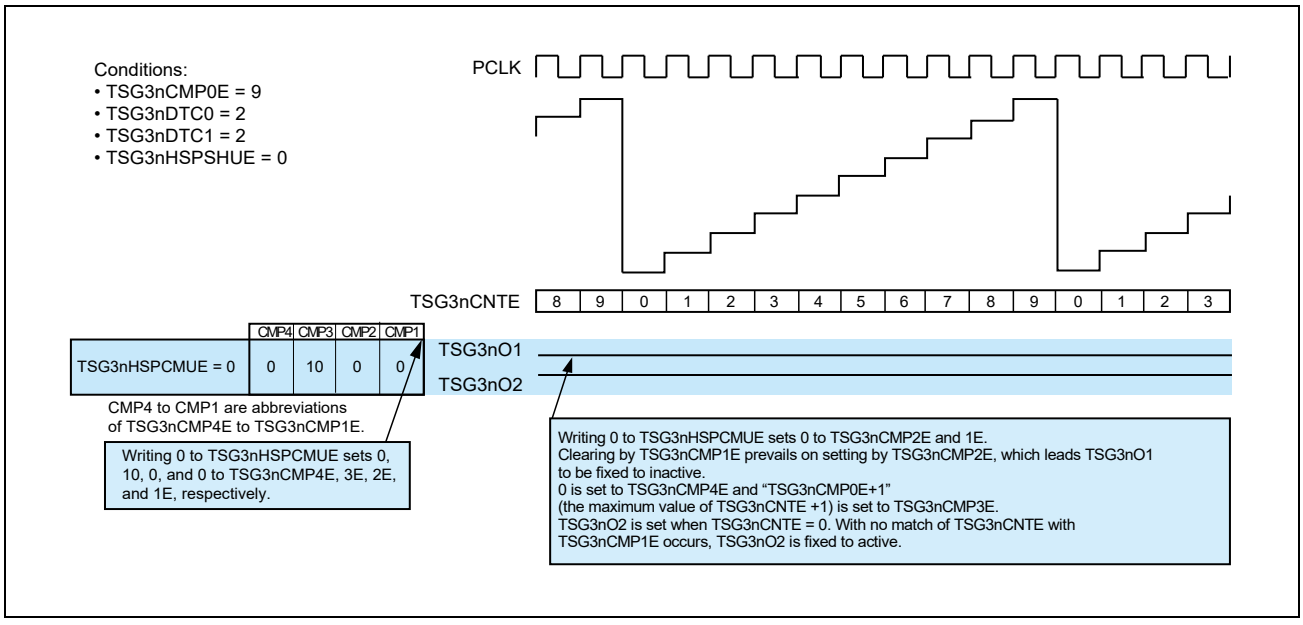


Figure 20.110 Waveform in HSP-PWM Mode (0 is set to TSG3nHSPCMUE)

(2) When TSG3nHSPCMUE/VE/WE (PWM output width setting) set value is 0 < TSG3nHSPCMUE ≤ TSG3nDTC0 + TSG3nDTC1

When the value within this range is set to TSG3nHSPCMUE (U phase PWM output width setting), 0 is set to TSG3nCMP1E and 2E, “the TSG3nHSPCMUE set value - 1” is set to TSG3nCMP4E, and the value same as TSG3nCMP0E is set to TSG3nCMP3E.

Setting of TSG3nO1 by the match of TSG3nCNTE with TSG3nCMP2E and clearing by the match with TSG3nCMP1E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO1 is fixed to inactive.

TSG3nO2 is cleared by the match of TSG3nCNTE with TSG3nCMP3E and set by the match with TSG3nCMP4E. Here, the output is shifted according to the set value in TSG3nHSPCMUE, that is, inactive for one cycle during PWM period when 1 is set, two cycles when 2 is set, and three cycles when 3 is set.

When the value other than 0 is set to TSG3nHSPSHUE (PWM shift width), set/clear timing of TSG3nO2 is shifted to the right for the number of cycles set in TSG3nHSPSHUE.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

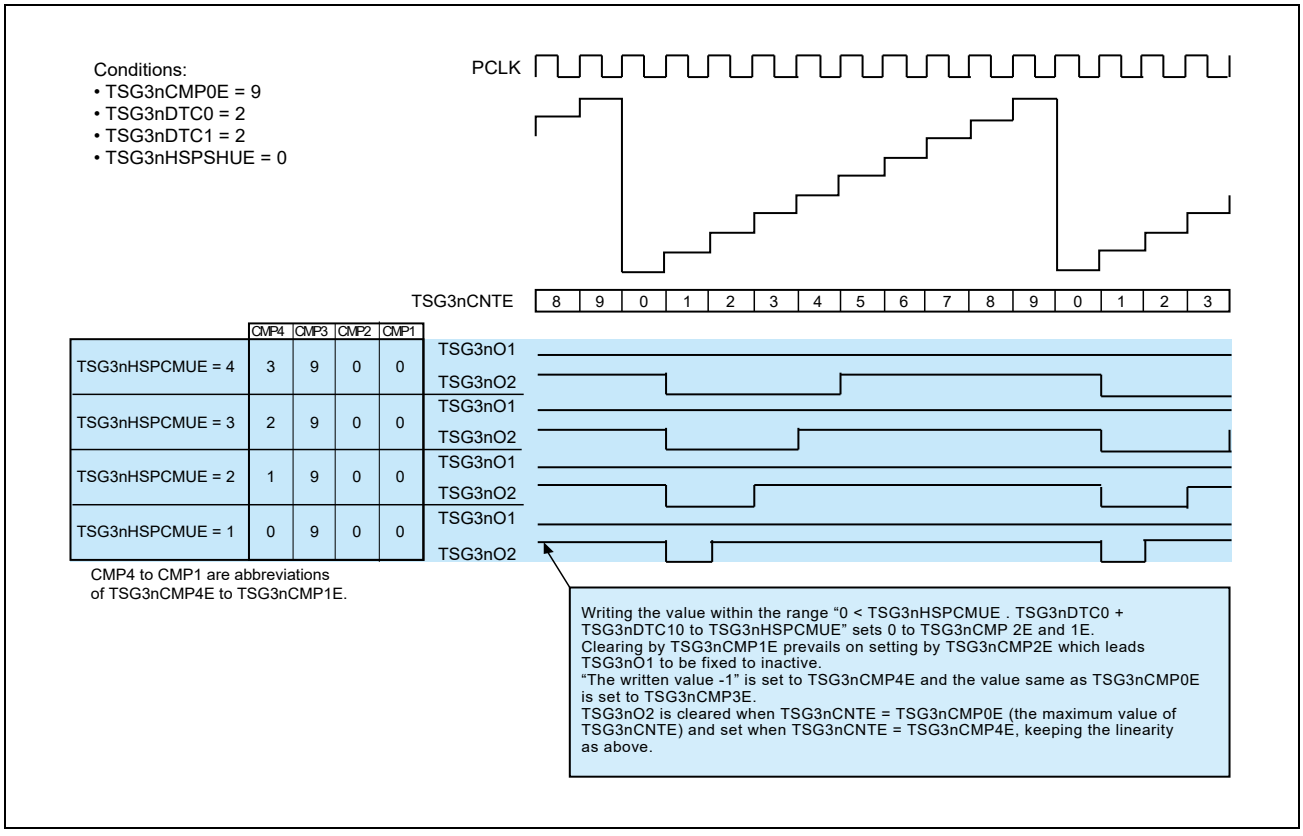


Figure 20.111 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to 0 < TSG3nHSPCMUE ≤ TSG3nDTC0 + TSG3nDTC1)

(3) When TSG3nHSPCMUE/VE/WE (PWM Output Width Setting) set value is TSG3nDTC0 + TSG3nDTC1 <TSG3nHSPCMUE/VE/WE ≤ TSG3nCMP0E

When the value within this range is set to TSG3nHSPCMUE (U phase PWM output width setting), “TSG3nDTCD0 -1” is set to TSG3nCMP2E, “TSG3nHSPCMUE - TSG3nDTC1 - 1” is set to TSG3nCMP1E, “TSG3nHSPCMUE - 1” is set to TSG3nCMP4E, and the value same as TSG3nCMP0E is set to TSG3nCMP3E.

TTSG3nO1 is set by the match of TSG3nCNTE with TSG3nCMP2E and cleared by the match with TSG3nCMP1E while TSG3nO2 is set by the match of TSG3nCNTE with TSG3nCMP4E and cleared by the match with TSG3nCMP3E

Here, the outputs are shifted according to the set value in TSG3nHSPCMUE, that is, when

“TSG3nDTC0 + TSG3nDTC1 + 1” is set, TSG3nO1 is active for one cycle during PWM period and TSG3nO2 is inactive for the cycles of “TSG3nDTC0 + TSG3nDTC1+ 1” during PWM period. On the other hand, when “TSG3nDTC0 + TSG3nDTC1 + 2” is set, TSG3nO1 is active for two cycles during PWM period and TSG3nO2 is inactive for the cycles of “TSG3nDTC0 + TSG3nDTC1+ 2” during PWM period.

When the value other than 0 is set to TSG3nHSPSHUE (PWM shift width), set/clear timing of TSG3nO1 and TSG3nO2 are shifted for the number of cycles set in TSG3nHSPSHUE to the right.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

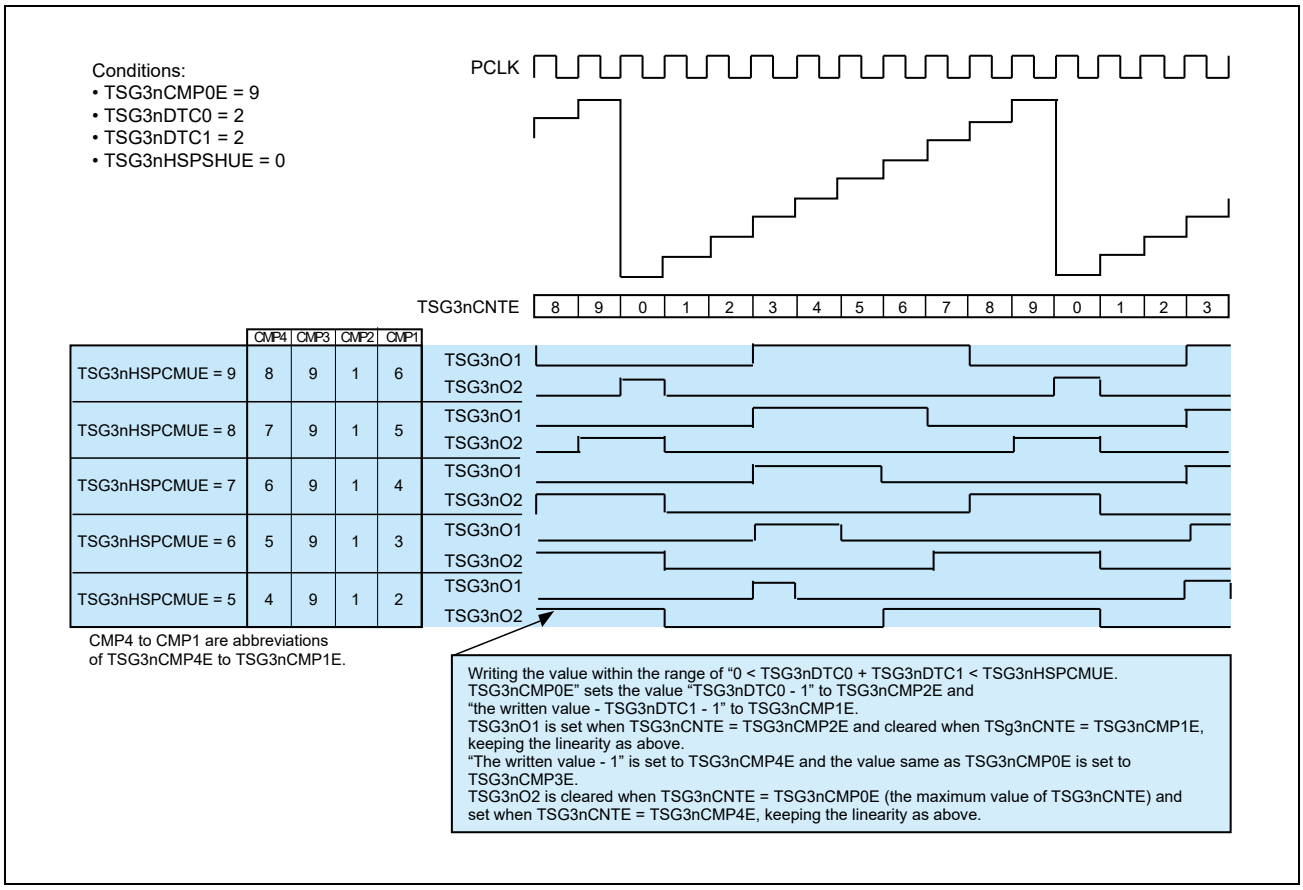


Figure 20.112 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to TSG3nDTC0 + TSG3nDTC1 < TSG3nHSPCMUE ≤ TSG3nCMP0E)

(4) When TSG3nHSPCMUE/VE/WE (PWM Output Width Setting) set value is TSG3nCMP0E < TSG3nHSPCMUE/VE/ WE < TSG3nCMP0E + TSG3nDTC1 + 1

When the value within this range is set to TSG3nHSPCMUE (U phase PWM output width setting), “TSG3nDTC0 - 1” is set to TSG3nCMP2E, “TSG3nHSPCMU - TSG3nDTC1 - 1” is set to TSG3nCMP1E, and 0 is set to TSG3nCMP3E and 4E.

TSG3nO1 is cleared by the match of TSG3nCNTE with TSG3nCMP1E and set by the match with TSG3nCMP2E.

Here, the output is shifted according to the set value in TSG3nHSPCMUE, that is, inactive for the cycles of “TSG3nDTC0 + TSG3nDTC1” during PWM period when “TSG3nCMP0E + 1” is set, inactive for the cycles of “TSG3nDTC0 + TSG3nDTC1 - 1” during PWM period when “TSG3nCMP0E + 2” is set.

Setting of TSG3nO2 by the match of TSG3nCNTE with TSG3nCMP4E and clearing by the match with TSG3nCMP3E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO2 is fixed to inactive.

When the value other than 0 is set to TSG3nHSPSHUE (PWM shift width), set/clear timing of TSG3nO1 is shifted to the right for the number of cycles set in TSG3nHSPSHUE.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

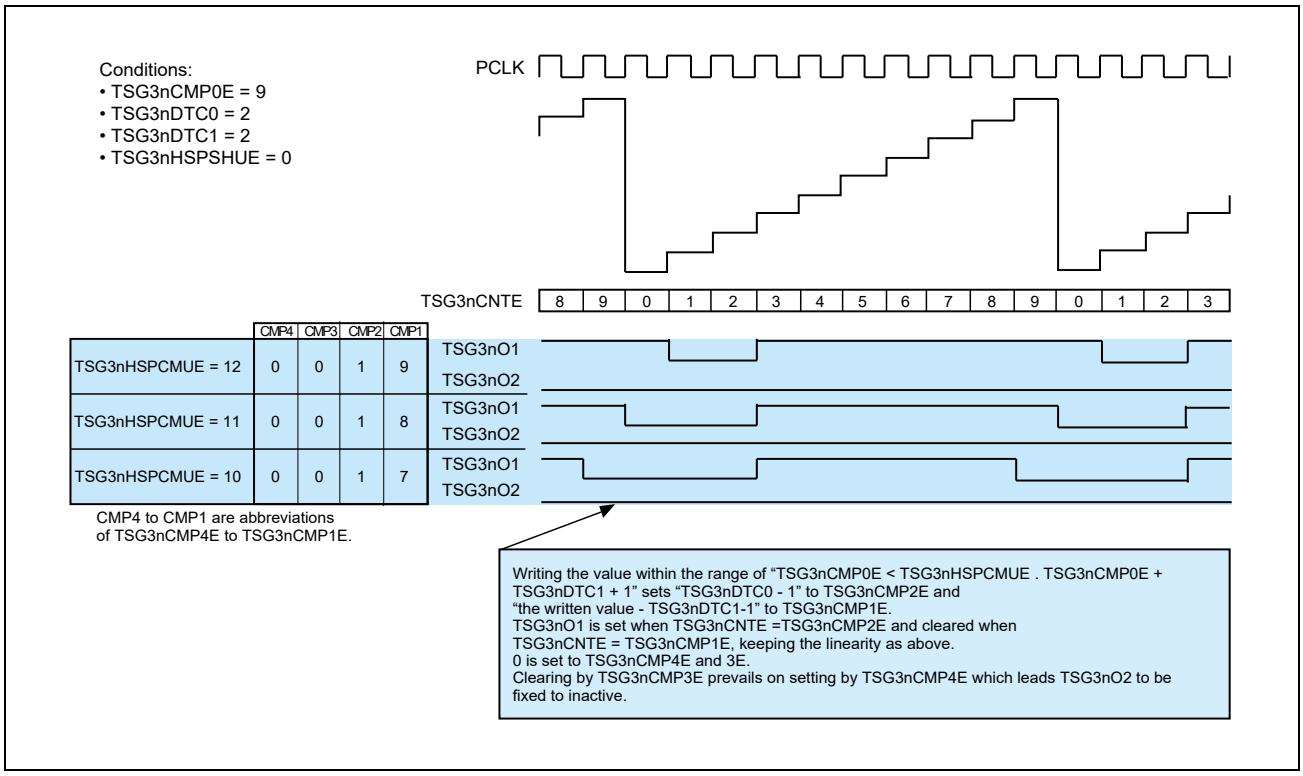


Figure 20.113 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to TSG3nCMP0E < TSG3nHSPCMUE < TSG3nCMP0E + TSG3nDTC1 + 1)

(5) When TSG3nHSPCMUE/VE/WE (PWM Output Width Setting) set value is TSG3nCMP0E + TSG3nDTC1 + 1 < TSG3nHSPCMUE/VE/WE < TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1

When the value within this range is set to TSG3nHSPCMUE (U phase PWM output width setting), “TSG3nDTC0 - 1” is set to TSG3nCMP2E, “TSG3nHSPCMUE - TSG3nCMP0E - TSG3nDTC1 - 2” is set to TSG3nCMP1E, and 0 is set to TSG3nCMP3E and 4E.

TSG3nO1 is cleared by the match of TSG3nCNTE with TSG3nCMP1E and set by the match with TSG3nCMP2E.

Here, the output is shifted according to the set value in TSG3nHSPCMUE, that is, inactive for one cycle during PWM period when “TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1” (the maximum value of PWM output width - 1) is set, inactive for two cycles during PWM period when “TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 - 1” (the maximum value of PWM output width - 2) is set.

Setting of TSG3nO2 by the match of TSG3nCNTE with TSG3nCMP4E and clearing by the match with TSG3nCMP3E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO2 is fixed to inactive.

When the value other than 0 is set to TSG3nHSPSHUE (PWM shift width), set/clear timing of TSG3nO1 is shifted to the right for the number of cycles set in TSG3nHSPSHUE.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

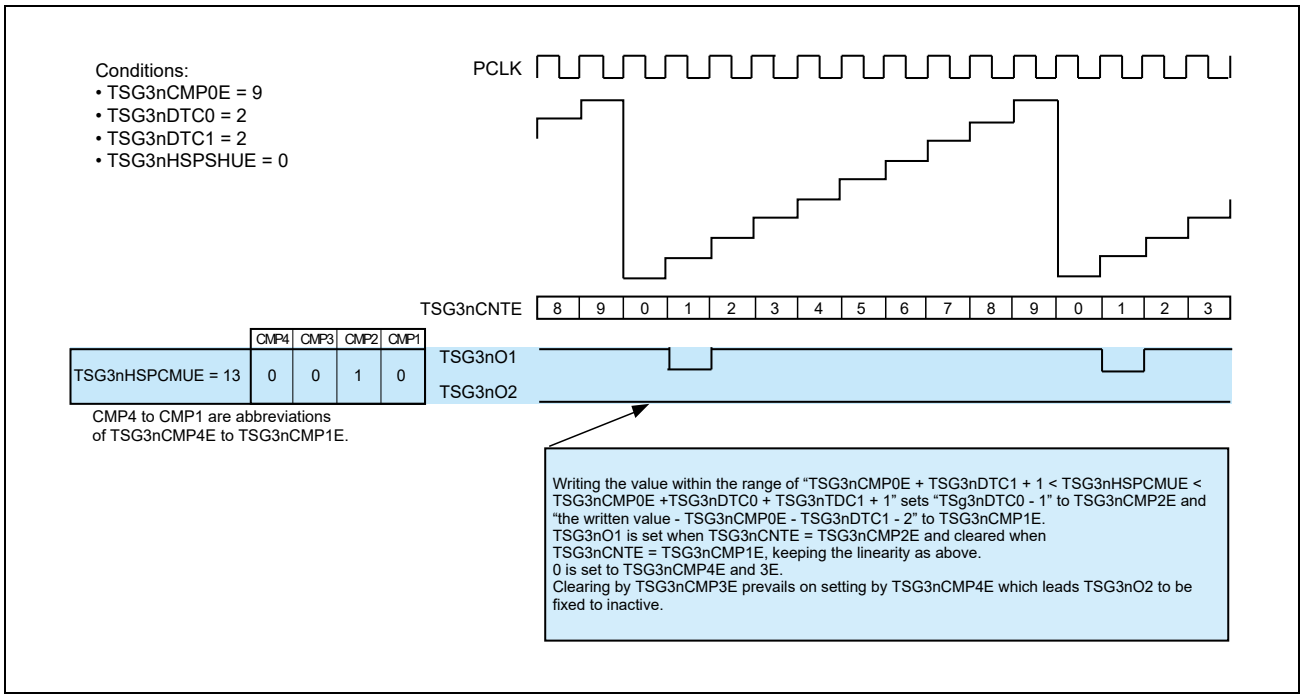


Figure 20.114 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to TSG3nCMP0E + TSG3nDTC1 + 1 < TSG3nHSPCMUE < TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1)

(6) When TSG3nHSPCMUE/VE/WE (PWM Output Width Setting) set value is TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1

When “TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1” (the maximum value of PWM output width) is set to TSG3nHSPCMUE, “TSG3nDTC0 - 1” is set to TSG3nCMP2E, “TSG3nCMP0E + 1” is set to TSG3nCMP1E, and 0 is set to TSG3nCMP3E and 4E.

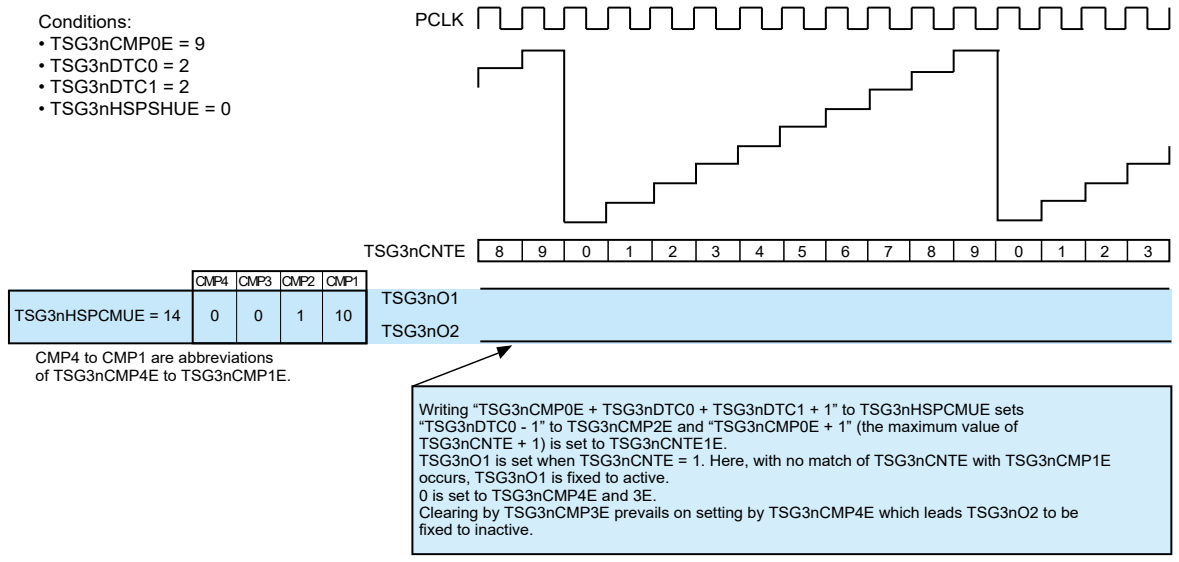
Setting of TSG3nO1 by the match of TSG3nCNTE with TSG3nCMP2E occurs when TSG3nCNTE = TSG3nDTC0 - 1.

“TSG3nCMP0E + 1” is set to TSG3nCMP3E. Here, with no match of TSG3nCNTE with TSG3nCMP3E occurs, TSG3nO1 is fixed to active.

Setting of TSG3nO2 by the match of TSG3nCNTE with TSG3nCMP4E and clearing by the match with TSG3nCMP3E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO2 is fixed to inactive.

When the value other than 0 is set to TSG3nHSPSHUE, set/clear timing of TSG3nO1 is shifted to the right for the number of cycles set in TSG3nHSPSHUE. Note that the set timing is shifted only at operation start (TSG3nTE = 0) because TSG3nO1 output is fixed to active. For the operation when operation start, see **Section (8) TO Operation at Operation Start in HT-PWM Mode**.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).



Note 1. In HSP-PWM mode, TSG3nO1 and TSG3nO2 can be set at the same timing by setting the same value to TSG3nCMP4E and 2E. In this case, both TSG3nO1 and TSG3nO2 go inactive.

Note 2. In HSP-PWM mode, dead time counter keeps operating even when TSG is stopped (TSG3nTE = 0) and the dead time set in TSG3nO1 and TSG3nO2 are always inserted.

Figure 20.115 Waveform in HSP-PWM (TSG3nHSPCMUE is set to TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1)

(7) PWM Adjustment at Reload Timing in HSP-PWM Mode

In HSP-PWM mode, PWM output width is adjusted when TSG3nHSPCMUE/VE/WE (PWM output width setting) is modified during operation. TSG3nO1-6 are set/cleared at reload timing and immediately switched to the output reflecting the new PWM output width.

TSG3nO1-6 are forcibly set/cleared according to the following formula. Even for adjustment, the dead time set to TSG3nDTC0 and 1 is always inserted to TSG3nO1-6. If values are directly written to TSG3nCMP1E-12E, output adjustment is not performed at reload timing.

Formula for reload adjustment operation

Table 20.100 Formula for Reload Adjustment Operation When TSG3nHSPSHUE/VE/WE is 0 (PWM sift width is set to 0)

Pin	Set	Clear
TSG3nO1/3/5	CMP0E + DTC1 + 1 < HSPCMUE/VE/WE	HSPCMUE/VE/WE ≤ CMP0E + DTC1 + 1
TSG3nO2/4/6	HSPCMUE/VE/WE = 0	0 < HSPCMUE/VE/WE

Table 20.101 Formula for Reload Adjustment Operation When TSG3nHSPSHUE/VE/WE is not 0 (PWM sift width is not set to 0)

Pin	Set	Clear
TSG3nO1/3/5	(i) CMP0E + DTC1 + 1 – HSPSHUE/VE/WE < HSPCMUE/VE/WE	HSPCMUE/VE/WE ≤ CMP0E + DTC1 + 1 – HSPSHUE/VE/WE
	(ii) (CMP0E + 1) × 2 + DTC1 – HSPSHUE/VE/WE < HSPCMUE/VE/WE	HSPCMUE/VE/WE ≤ (CMP0E + 1) × 2 + DTC1 – HSPSHUE/VE/WE
TSG3nO2/4/6	HSPCMUE/VE/WE ≤ CMP0E + 1 – HSPSHUE/VE/WE	CMP0E + 1 – HSPSHUE/VE/WE < HSPCMUE/VE/WE

If the value other than 0 is set as PWM shift width, set/clear condition for positive phase is determined whether the set shift width is greater than “CMP0E + DTC0 – 1” or not.

(i) HSPSHUE/VE/WE ≤ CMP0 – DTC0 + 1

(ii) HSPSHUE/VE/WE > CMP0 – DTC0 + 1

“TSG3n” is omitted from the register names used in the calculation.

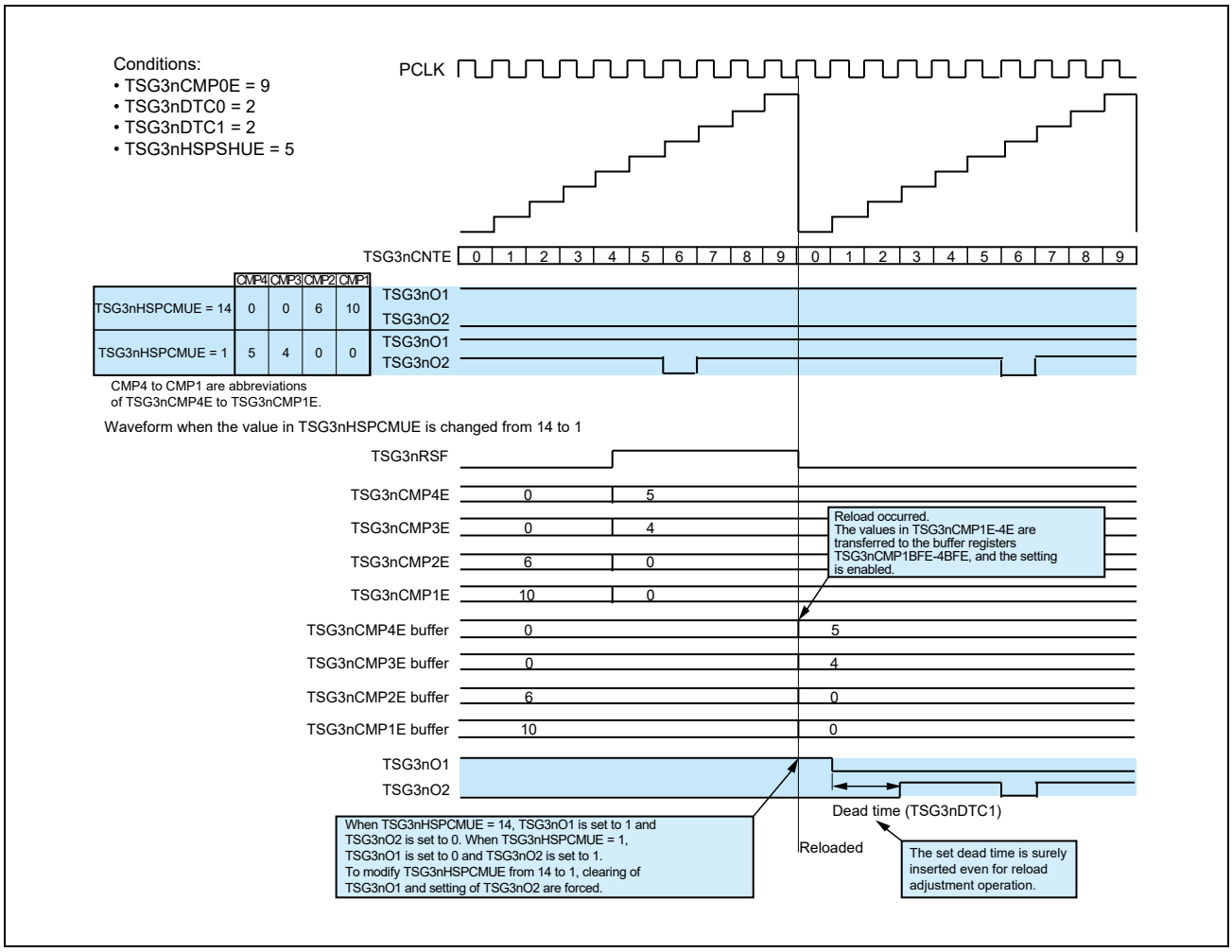


Figure 20.116 Detailed Timing Diagram of Reload Adjustment (Modification of TSG3nHSPCMUE from 14 to 1)

(8) TO Operation at Operation Start in HT-PWM Mode

In HSP-PWM mode, TSG3nO1-6 are cleared when operation starts.

Then, TSG3nO1-6 are set/cleared as TSG3nCNTE counts up depending on the values set in TSG3nHSPCMUE/VE/WE (TSG3nCMP1E to 12E).

Even if TSG3nO1-6 are set before operation start and cleared at operation start, and then set again, the set dead time is always inserted.

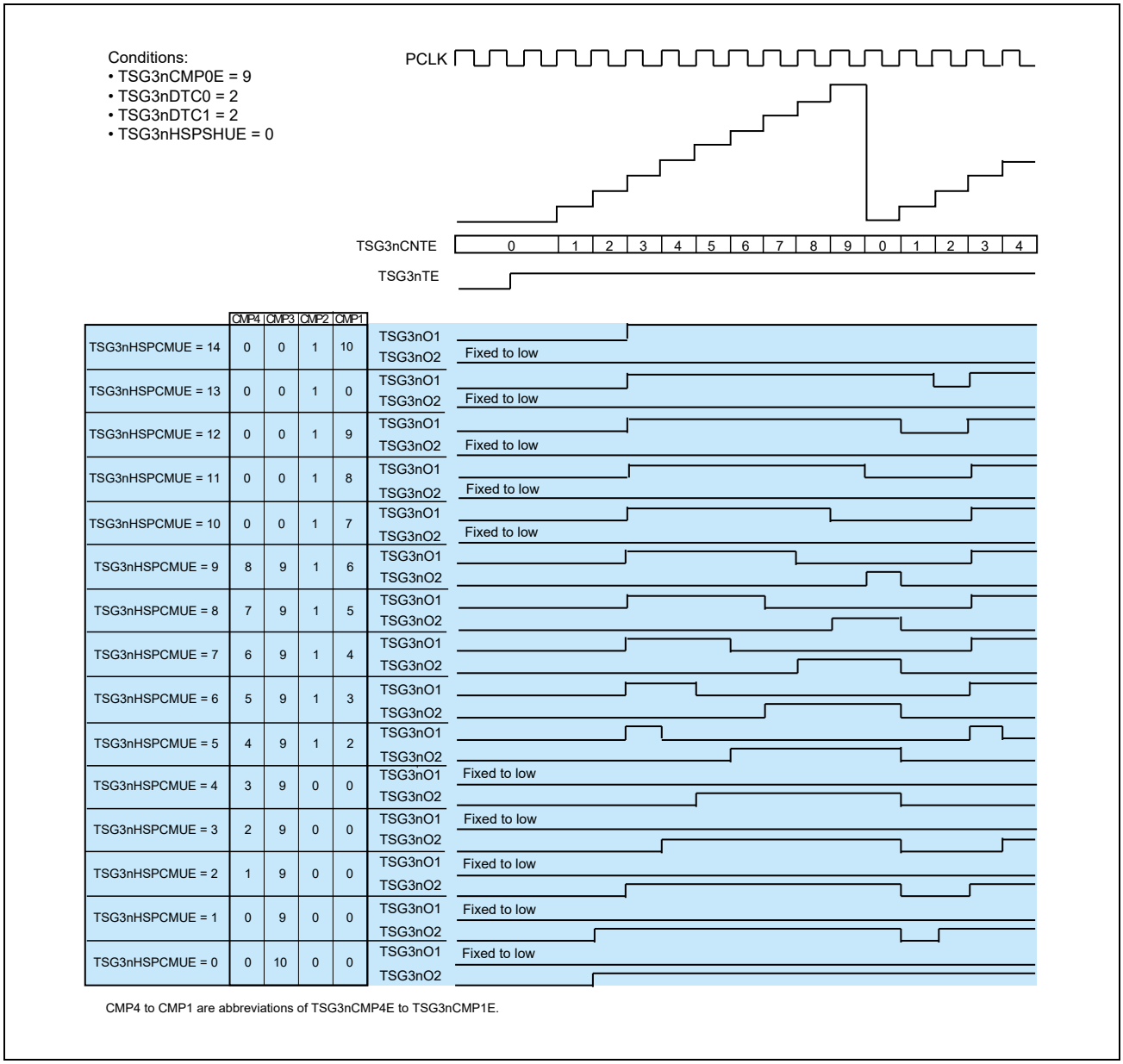


Figure 20.117 Timing Diagram of Operation Start in HSP-PWM Mode (TSG3nHSPSHUE = 0 (shift width is set to 0))

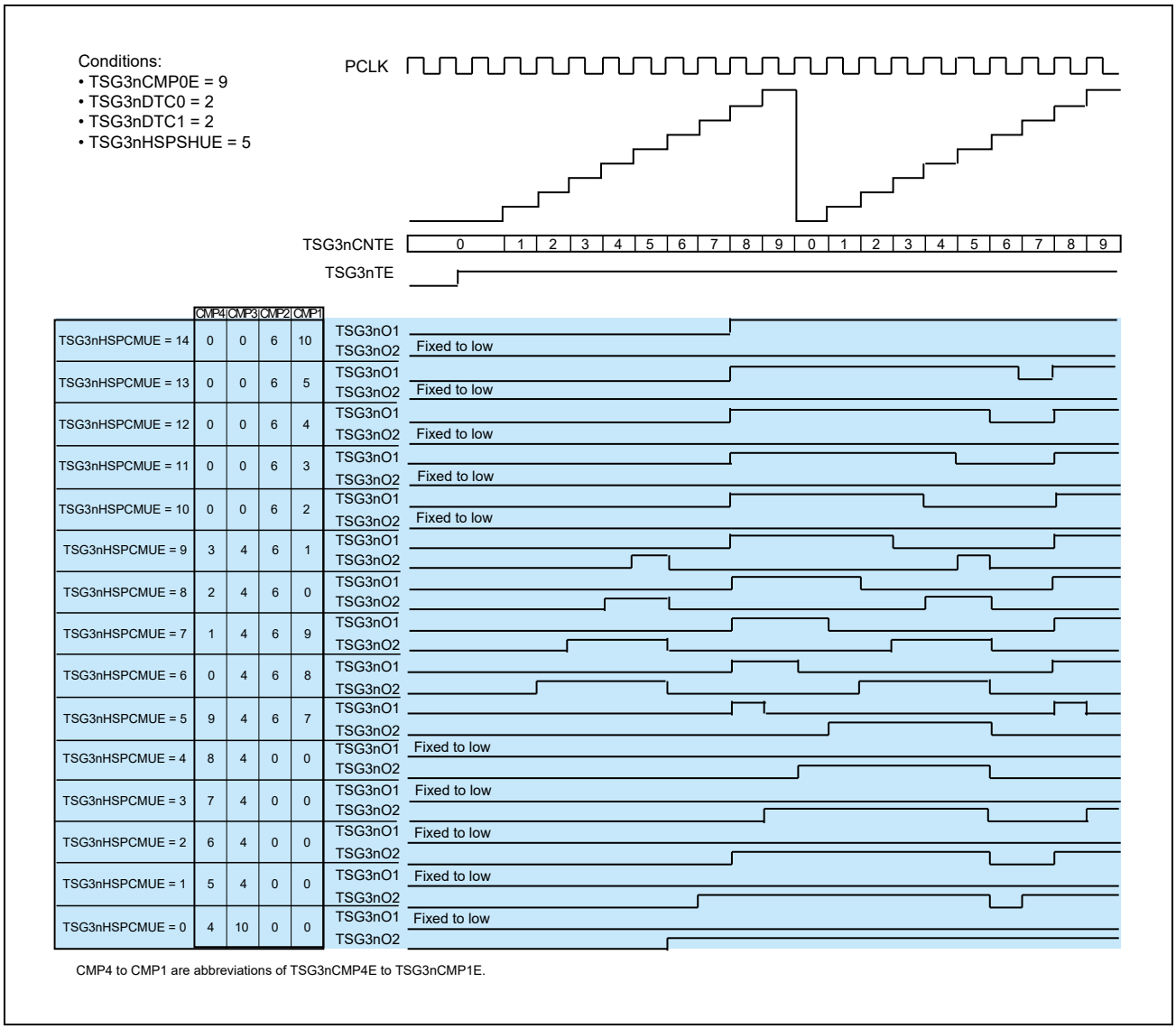


Figure 20.118 Timing Diagram of Operation Start in HSP-PWM Mode (TSG3nHSPSHUE = 5 (shift width is set to five clock cycles))

20.4.7.10 Software Output Control Function

Software output control function is available in any mode except HSP-PWM mode. This function can switch six output patterns for the TSG3nO1 to TSG3nO6 pins using TSG3nOPT0.TSG3nSOC, TSG3nIDC, and TSG3nOPT0.TSG3nSPC2-0.

When TSG3nSOC is switched from 0 to 1, the output control method of the TSG3nO1 to TSG3nO6 pins is switched to the software output control immediately. On the contrast, when TSG3nSOC is switched from 1 to 0, the software output control is released at the reload timing.

Table 20.102 Registers Associated with Software Output Control Function

Register	Operation
TSG3nOPT0.TSG3nSOC	TSG3nSOC = 1
TSG3nOPT0.TSG3nSTE	TSG3nSTE = 0
TSG3nOPT1.TSG3nSPC2-TSG3nSPC0	Sets output patterns listed in the following **Table 20.103** and **Table 20.104**.
TSG3nOPT0.TSG3nIDC	Sets output pattern (electric current direction).

Table 20.103 Output Patterns by Software Output Control (TSG3nOPT0.TSG3nIDC = 0) TSG3nOPT0.TSG3nSOC = 1, TSG3nSTE = 0, TSG3nIDC = 0

	TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
Output Pins	101	100	110	010	011	001	000	111
TSG3nO1	ACT	ACT	ACT	INACT	INACT	INACT	INACT	ACT
TSG3nO2	INACT	INACT	INACT	ACT	ACT	ACT	ACT	INACT
TSG3nO3	INACT	INACT	ACT	ACT	ACT	INACT	INACT	ACT
TSG3nO4	ACT	ACT	INACT	INACT	INACT	ACT	ACT	INACT
TSG3nO5	ACT	INACT	INACT	INACT	ACT	ACT	INACT	ACT
TSG3nO6	INACT	ACT	ACT	ACT	INACT	INACT	ACT	INACT

Note: ACT: Indicates active level is output.
INACT: Indicates inactive level is output.

Table 20.104 Output Patterns by Software Output Control (TSG3nOPT0.TSG3nIDC = 1) TSG3nOPT0.TSG3nSOC = 1, TSG3nSTE = 0, TSG3nIDC = 1

	TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
Output Pins	101	100	110	010	011	001	000	111
TSG3nO1	INACT	INACT	INACT	ACT	ACT	ACT	ACT	INACT
TSG3nO2	ACT	ACT	ACT	INACT	INACT	INACT	INACT	ACT
TSG3nO3	ACT	ACT	INACT	INACT	INACT	ACT	ACT	INACT
TSG3nO4	INACT	INACT	ACT	ACT	ACT	INACT	INACT	ACT
TSG3nO5	INACT	ACT	ACT	ACT	INACT	INACT	ACT	INACT
TSG3nO6	ACT	INACT	INACT	INACT	ACT	ACT	INACT	ACT

Note: ACT: Indicates active level is output.
INACT: Indicates inactive level is output.

Section 21 Timer Option Module (TAPA)

This section contains a generic description of the timer option module (TAPA).

The first part of this section describes RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of this section describes TAPA functions and registers.

21.1 Features of RH850/C1M-A TAPA

21.1.1 Units

This LSI has the following number of TAPA units.

Table 21.1 Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	6	4
Name	TAPAn (n = 0 to 5)	TAPAn (n = 0, 1, 3, 4)

Table 21.2 Index

Index	Meaning
n	Throughout this section, the individual TAPA units are identified by the index “n” (n = 0 to 5); for example, TAPAnFLG is the TAPAn flag register.

21.1.2 Register Base Address

TAPA base addresses are listed in the table below.

TAPA register addresses are given as offsets from the base addresses in general.

Table 21.3 Register Base Address

Base Address Name	Base Address
<TAPA0_base>	FFE9 0000_H
<TAPA1_base>	FF89 1000_H
<TAPA2_base>	FFE9 2000_H
<TAPA3_base>	FFE9 3000_H
<TAPA4_base>	FF89 4000_H
<TAPA5_base>	FFE9 5000_H

21.1.3 Clock Supply

Clock supply by and to TAPA is listed in the following table.

Table 21.4 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
TAPAn	PCLK	CLKC_HSB (unmodulated high-speed peripheral clock)

21.1.4 Interrupt Requests

TAPA interrupt requests are listed in the following table.

Table 21.5 Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
TAPA0						
TAPA0TIPEK0	TAPA0 peak interrupt	176	—	89	—	89
TAPA0TIVLY0	TAPA0 trough interrupt	177	—	90	—	90
TAPA1						
TAPA1TIPEK0	TAPA1 peak interrupt	178	—	93	—	93
TAPA1TIVLY0	TAPA1 trough interrupt	179	—	94	—	94
TAPA2*2						
TAPA2TIPEK0	TAPA2 peak interrupt	180	—	97	—	97
TAPA2TIVLY0	TAPA2 trough interrupt	181	—	98	—	98

—: Not assigned

Note 1. 1st: Primary channel, 2nd: Secondary channel

Note 2. RH850/C1M-A1 does not have TAPA2.

21.1.5 Reset Source

TAPA reset sources are listed in the following table. TAPA is initialized by the reset sources below.

Table 21.6 Reset Source

Unit Name	Reset Source
TAPAn	Any reset source

21.1.6 Peripheral Configuration

The following figure shows the peripheral configuration of TAPA.

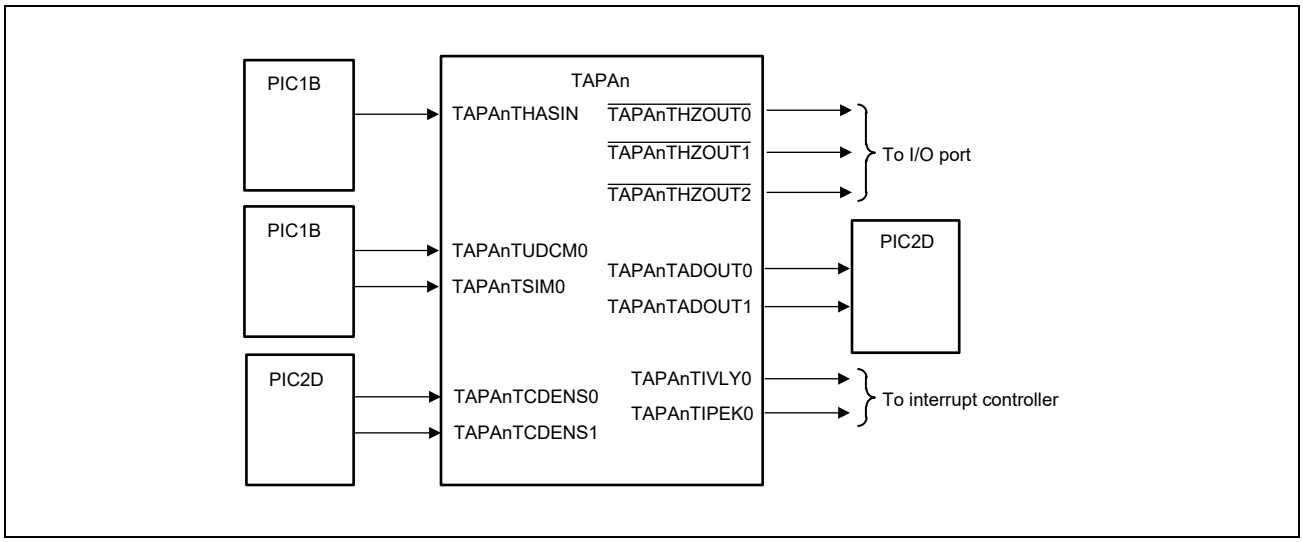


Figure 21.1 Peripheral Configuration of TAPA

The following describes the peripheral configuration of TAPA.

- TAPAnTHASIN: Hi-Z control asynchronous input signal
  This signal controls Hi-Z by the source selected in PIC1B.
  For the sources that can be selected in PIC1B, see **24.2.3.16, Hi-Z Control Function.**
- TAPAnTUDCM0: TAUD master channel up/down input
- TAPAnTSIM0: TAUD master channel INT input
  Peak and trough interrupts can be generated on TAUDn channels selected in PIC1B.
  For TAUDn channels that can be selected in PIC1B, see **24.2.2.34, PIC1BREG200 — Timer Input/Output Control Register 200**, **24.2.2.35, PIC1BREG210 — Timer Input/Output Control Register 210**.
  Only TAPA0, TAPA1, and TAPA2 have this connection.
- TAPAnTCDENS0, TAPAnTCDENS1: TAUD slave channel match detection input
  The TAUDn channel interrupts selected in PIC2D can be handled as AD conversion trigger outputs 0 and 1 (TAPAnTADOUT1 and TAPAnTADOUT0). For the TAUDn channel interrupts that can be selected in PIC, see **24.3.3.2, TAUD Trigger Output Function**.
  Only TAPA0, TAPA1, and TAPA2 have this connection.
- TAPAnTADOUT1-0: A/D conversion trigger outputs 1 and 0
  The A/D conversion trigger signals generated on TAPAn are output to PIC2D. The setting of the PIC2D can also be used to set these signals as triggers for A/D conversion. For the register specifications in the PIC2D, see **24.3.3.1, ADCC Trigger Select Function**.
  Only TAPA0, TAPA1, and TAPA2 have this connection.

21.2 Overview

21.2.1 Functional Overview

The timer option module (TAPA) is for use with the timer array unit D (TAUD) and TSG3 modules.

- Asynchronous Hi-Z control to each TAUD and TSG3 output is enabled by TAPA input signals.
- Output of the INTn signal from the TAUD as a peak or trough interrupt is selectable.
- The INTn signal output by the TAUD provides the basis for the output of two conversion-trigger signals for the A/D converter.

21.2.2 Terms

In this section, the following terms are used.

“Peak” and “Trough”, and “Peak Interrupt” and “Trough Interrupt”

In this document, the period from a TAUD down status (counting-down status) to generation of INT from the master channel is defined as a “trough” period, and this INT is defined as a “trough interrupt” (INT-VLY).

In contrast, the period from a TAUD up status (counting-up status) to generation of INT from the master channel is defined as a “peak” period, and this INT is defined as a “peak interrupt” (INT-PEK).

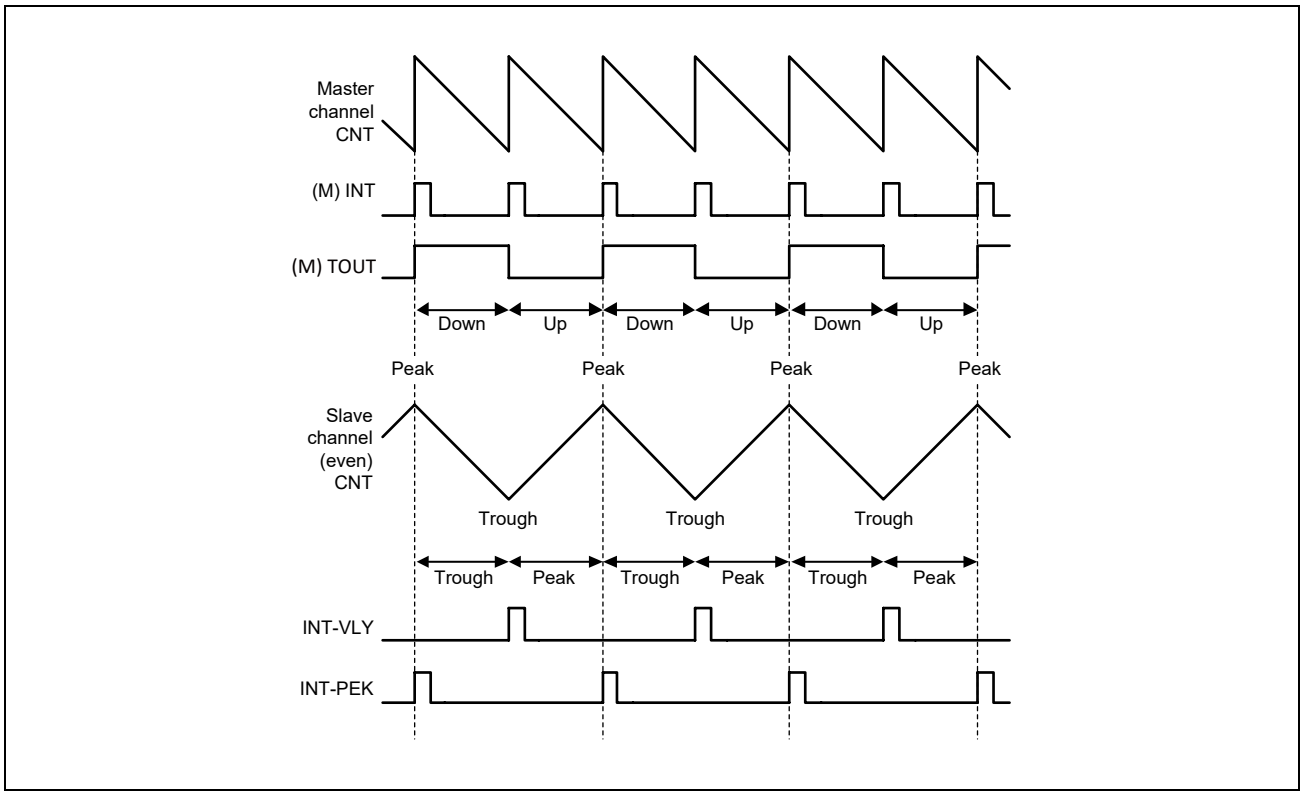


Figure 21.2 Peak and Trough of Timer Counter, and Peak and Trough Interrupts

21.2.3 Block Diagram

The major components of TAPA are shown in the block diagram below.

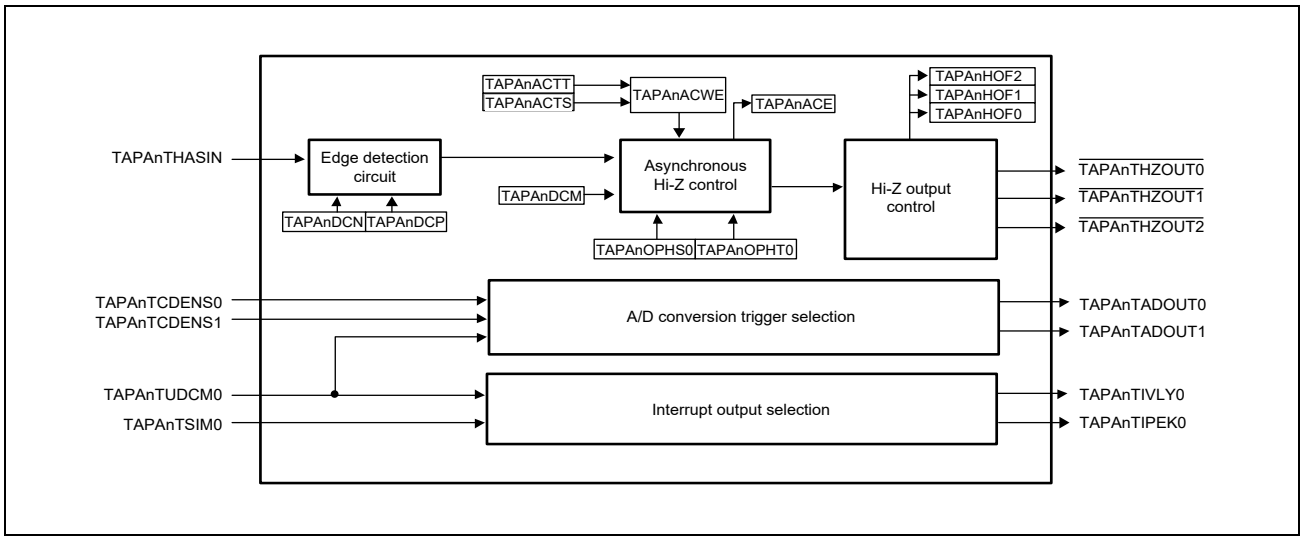


Figure 21.3 Block Diagram of TAPA

21.3 Registers

21.3.1 List of Registers

The following table lists the registers of TAPAn.

For information on <TAPAn_base>, see **Section 21.1.2, Register Base Address**.

Table 21.7 List of Registers

Module Name	Register Name	Symbol	Address
TAPAn	TAPAn control register 0	TAPAnCTL0	<TAPAn_base> + 20_H
TAPAn	TAPAn control register1*1	TAPAnCTL1	<TAPAn_base> + 24_H
TAPAn	TAPAn flag register	TAPAnFLG	<TAPAn_base> + 00_H
TAPAn	TAPAn asynchronous write enable register	TAPAnACWE	<TAPAn_base> + 04_H
TAPAn	TAPAn asynchronous control start trigger register	TAPAnACTS	<TAPAn_base> + 08_H
TAPAn	TAPAn asynchronous control stop trigger register	TAPAnACTT	<TAPAn_base> + $0C_H$
TAPAn	TAPAn Hi-Z start trigger register	TAPAnOPHS	<TAPAn_base> + 14_H
TAPAn	TAPAn Hi-Z stop trigger register	TAPAnOPHT	<TAPAn_base> + 18_H

Note 1. TAPAnCTL1 is valid only when TAPAn (n = 0, 1, 2).

21.3.2 TAPAnCTL0 ─ TAPAn Control Register0

Control register 0 is used to control Hi-Z.

A value in this register can only be rewritten in the following conditions.

- TAPAnFLG.TAPAnACE = 0 while TAPAn (n = 0, 1, 2) and TAUDnTEm = 0 (m = 10 to 15) at the corresponding TAUDn master channel.
- TAPAnFLG.TAPAnACE = 0 while TAPAn (n = 3, 4, 5).

Access: This register can be read/written in 16-bit units.

Address: <TAPAn_base> + 20_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TAPAn DCM	TAPAn DCN	TAPAn DCP	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Table 21.8 TAPAnCTL0 Register Contents

Bit Position	Bit Name	Function
15 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	TAPAnDCM	Clearing Condition Specification This control bit specifies the condition for clearing of the Hi-Z control outputs. 0: Manipulation of TAPAnOPHT0 is enabled regardless of the TAPAnTHASIN signal input level. 1: Manipulation of TAPAnOPHT0 is disabled when the TAPAnTHASIN input signal is at the active level. Manipulation of TAPAnOPHT0 is enabled when the TAPAnTHASIN signal input is inactive.
3, 2	TAPAnDCN, TAPAnDCP	Hi-Z Input Edge Selection These control bits specify the effective edge of TAPAnTHASIN.

TAPAnDCN	TAPAnDCP	Description
0	0	Does not detect effective edges.
0	1	Detects a rising edge as the effective edge. (active level = high)
1	0	Detects a falling edge as the effective edge. (active level = low)
1	1	Setting is prohibited.

1, 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

21.3.3 TAPAnCTL1 ─ TAPAn Control Register 1

TAPAn control register 1 is only valid when n = 0, 1 or 2.

Access: This register can be read/written in 8-bit units.

Address: <TAPAn_base> + 24_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAPAnATS3	TAPAnATS2	TAPAnATS1	TAPAnATS0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.9 TAPAnCTL1 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3, 2	TAPAnATS3, TAPAnATS2	A/D Converter Trigger 1 Select These control bits specify the signal for output as A/D converter conversion trigger output 1 (TAPAnTADOUT1).

TAPA0ATS3	TAPA0ATS2	Description
0	0	Outputs INT while the master channel is in the down state.
0	1	Outputs INT while the master channel is in the up state.
1	0	Outputs INT while the master channel is in the up/down state.
1	1	Outputs INT and a trough interrupt of the master channel (TAPAnTIVLY0) while the master channel is in the up/down state.

Bit Position	Bit Name	Function
1, 0	TAPAnATS1, TAPAnATS0	AD Converter Trigger 0 Select These control bits specify the signal for output as A/D converter conversion trigger output 0 (TAPAnTADOUT0).

TAPA0ATS1	TAPA0ATS0	Description
0	0	Outputs INT while the master channel is in the down state.
0	1	Outputs INT while the master channel is in the up state.
1	0	Outputs INT while the master channel is in the up/down state.
1	1	Outputs INT and a trough interrupt of the master channel (TAPAnTIVLY0) while the master channel is in the up/down state.

21.3.4 TAPAnFLG ─ TAPAn Flag Register

Control register is used to control Hi-Z.

Access: This register can be read/written in 16-bit units.

Address: <TAPAn_base> + 00_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TAPAn HOF2	TAPAn HOF1	TAPAn HOF0	—	—	—	—	—	—	—	TAPAn ACE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.10 TAPAnFLG Register Contents

Bit Position	Bit Name	Function
15 to 11	Reserved	When read, the value after reset is read.
10 to 8	TAPAnHOFm	$\overline{\text{TAPAnTHZOUTm}}$ Output Monitor (m = 0, 1, 2) These bits monitor the $\overline{\text{TAPAnTHZOUTm}}$ output. 0: The $\overline{\text{TAPAnTHZOUTm}}$ output is at the high level. 1: The $\overline{\text{TAPAnTHZOUTm}}$ output is at the low level.*1
7 to 1	Reserved	When read, the value after reset is read.
0	TAPAnACE	Asynchronous Hi-Z Control Enable This bit indicates the state of asynchronous Hi-Z control. 0: Asynchronous Hi-Z control is stopped. 1: Asynchronous Hi-Z control is enabled. The conditions for setting and clearing this bit are as follows. Clearing condition: Writing 1 to TAPAnACTT while TAPAnACWE = 1. Setting condition: Writing 1 to TAPAnACTS while TAPAnACWE = 1.

Note 1. TAPAnHOFm (m = 1, 2) is valid only when TAPAn (n = 0, 1, 2).

21.3.5 TAPAnACWE — TAPAn Asynchronous Control Write Enable Register

This register enables writing for asynchronous Hi-Z control.

Access: This register can be read/written in 8-bit units.

Address: <TAPAn_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnACWE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.11 TAPAnACWE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TAPAnACWE	Asynchronous Control Write Enable This is a write-enable bit for asynchronous Hi-Z control. After 1 has been written to this bit, it is automatically cleared to 0 by writing 1 to TAPAnACTS or TAPAnACTT. 0: Disables writing to TAPAnACTS and TAPAnACTT. 1: Enables writing to TAPAnACTS and TAPAnACTT.

21.3.6 TAPAnACTS — TAPAn Asynchronous Control Start Trigger Register

This register enables the start trigger for asynchronous Hi-Z control.

Access: This register can be written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnACTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 21.12 TAPAnACTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TAPAnACTS	Asynchronous Control Start Trigger This bit enables the start trigger for asynchronous Hi-Z control. The setting of this bit is only valid when TAPAnACWE = 1. 0: Writing 0 to this bit is ignored (no function). 1: Enables asynchronous Hi-Z control if TAPAnACWE = 1.

21.3.7 TAPAnACTT — TAPAn Asynchronous Control Stop Trigger Register

This register enables the stop trigger for asynchronous Hi-Z control.

Access: This register can be written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnACTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 21.13 TAPAnACTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TAPAnACTT	Asynchronous Control Stop Trigger This bit enables the stop trigger for asynchronous Hi-Z control. The setting of this bit is only valid when TAPAnACWE = 1. 0: Writing 0 to this bit is ignored (no function). 1: Stops asynchronous Hi-Z control if TAPAnACWE = 1.

21.3.8 TAPAnOPHS — TAPAn Hi-Z Start Trigger Register

This register sets the start trigger for a Hi-Z control signal ($\overline{\text{TAPAnTHZOUTm}}$ (m = 0 to 2*[1])).

Note 1. TAPAn (n = 3, 4, 5) are not available when m = 1, 2.

Access: This register can be written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnOPHS0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 21.14 TAPAnOPHS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TAPAnOPHS0	Hi-Z Control Signal Start Trigger 0 This bit sets the start trigger for a Hi-Z control signal. 0: The read value is 0 and writing 0 to this bit is ignored (no function). 1: Sets the corresponding Hi-Z control signal ($\overline{\text{TAPAnTHZOUTm}}$ (m = 0 to 2*[1])) to the low level.

21.3.9 TAPAnOPHT ─ TAPAn Hi-Z Stop Trigger Register

This register sets the stop trigger for a Hi-Z control signal ($\overline{\text{TAPAnTHZOUTm}}$ (m = 0 to 2*[1])).

Note 1. TAPAn (n = 3, 4, 5) are not available when m = 1, 2.

Access: This register can be written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 18_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnOPHT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 21.15 TAPAnOPHT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TAPAnOPHT0	Hi-Z Control Signal Stop Trigger 0 This bit sets the stop trigger for a Hi-Z control signal. 0: The read value is 0 and writing 0 to this bit is ignored (no function). 1: Sets a Hi-Z control signal ($\overline{\text{TAPAnTHZOUTm}}$ (m = 0 to 2*[1])) to the high level.

21.4 Function

21.4.1 Asynchronous Hi-Z Control Function

Abnormal operation in timer motor-control under CPU control leads to rotation of the externally connected motor also becoming abnormal. In such a case, this function forcibly sets the motor control output to the Hi-Z state, independently of control by the CPU.

21.4.1.1 Overview

- The following method is available for controlling Hi-Z.
  - Asynchronous Hi-Z control for TAPA input signals (TAPAnTHASIN)
    Controls the Hi-Z control output signals $\overline{\text{TAPAnTHZOUT0}}$ (phase U), $\overline{\text{TAPAnTHZOUT1}}$ (phase V), $\overline{\text{TAPAnTHZOUT2}}$ (phase W) asynchronously at TAPAn (n = 0, 1, 2).
  - Controls the Hi-Z control output signals $\overline{\text{TAPAnTHZOUT0}}$ asynchronously at TAPAn (n = 3, 4, 5).

Table 21.16 Asynchronous Hi-Z Control and Its Operation

Hi-Z Control	Function and Operation
Asynchronous Hi-Z control for TAPA input signals (TAPAnTHASIN)	Asynchronous Hi-Z control This function forcibly places the output from the corresponding timer module (TAUD, TSG3) into Hi-Z. Device port outputs become Hi-Z while TAPAnTHASIN is active and until software sends a stop request (when TAPAnCTL0.TAPAnDCM = 0)

NOTE

The following timer output pins are controlled by this function:

- TAPAn (n = 0, 1, 2)
  - TAPAnTHZOUT0 (U phase): TAPAnUP, TAPAnUN
  - TAPAnTHZOUT1 (V phase): TAPAnVP, TAPAnVN
  - TAPAnTHZOUT2 (W phase): TAPAnWP, TAPAnWN
- TAPAn (n = 3)
  - TAPAnTHZOUT0: TSG30O1 to TSG30O6
- TAPAn (n = 4)
  - TAPAnTHZOUT0: TSG31O1 to TSG31O6
- TAPAn (n = 5)
  - TAPAnTHZOUT0: TSG32O1 to TSG32O6

CAUTION

There is no function that forcibly places the output from TAUD3 into Hi-Z. For the configuration, see **Section 24.2.3.16, Hi-Z Control Function**.

21.4.1.2 An Example of System Configuration

When effective edges of the external error detection signal are detected, an interrupt is generated and, at the same time, the motor-driving signal output is set to the Hi-Z state.

This module assumes that microcontroller operation may hang when an error occurs. To handle such situations, external error detection signals are continuously processed so that the motor-driving signal can be set to the Hi-Z state even if no clock signal is being supplied.

This module only detects an error as an edge of the error-detection signal. A fixed output level is not detected as an error (the signal has no edge).

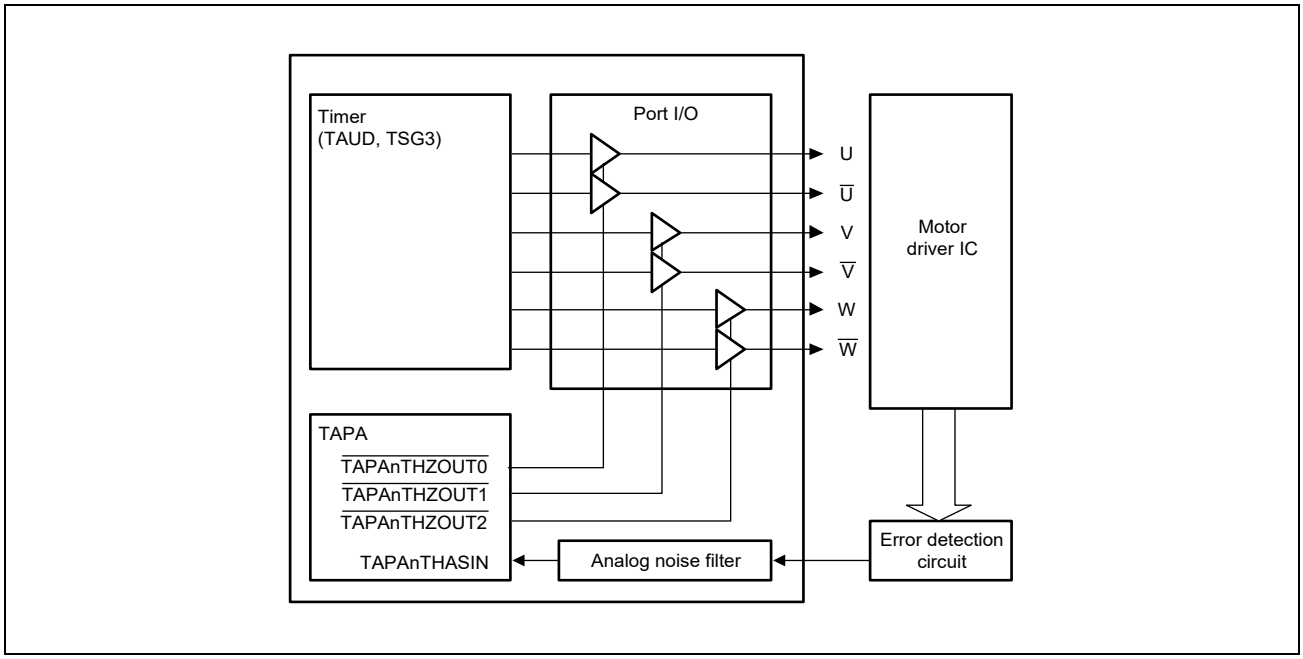


Figure 21.4 An Example of System Configuration of Asynchronous Hi-Z Control for Analog Inputs

21.4.1.3 Basic Operation

Setting examples are described as follows.

Hi-Z Control when TAPAnCTL0.TAPAnDCM = 0, TAPAnDCP = 1, and TAPAnDCN = 0

$\overline{\text{TAPAnTHZOUT0}}$ goes to the low level on detection of an effective edge of the asynchronous input (TAPAnTHASIN). Output is forcibly stopped (by port control for Hi-Z output) as long as the $\overline{\text{TAPAnTHZOUT0}}$ output is at the low level. $\overline{\text{TAPAnTHZOUT0}}$ goes to the high level in response to writing 1 to Hi-Z stop trigger 0 (TAPAnOPHT0), regardless of the level of TAPAnTHASIN.

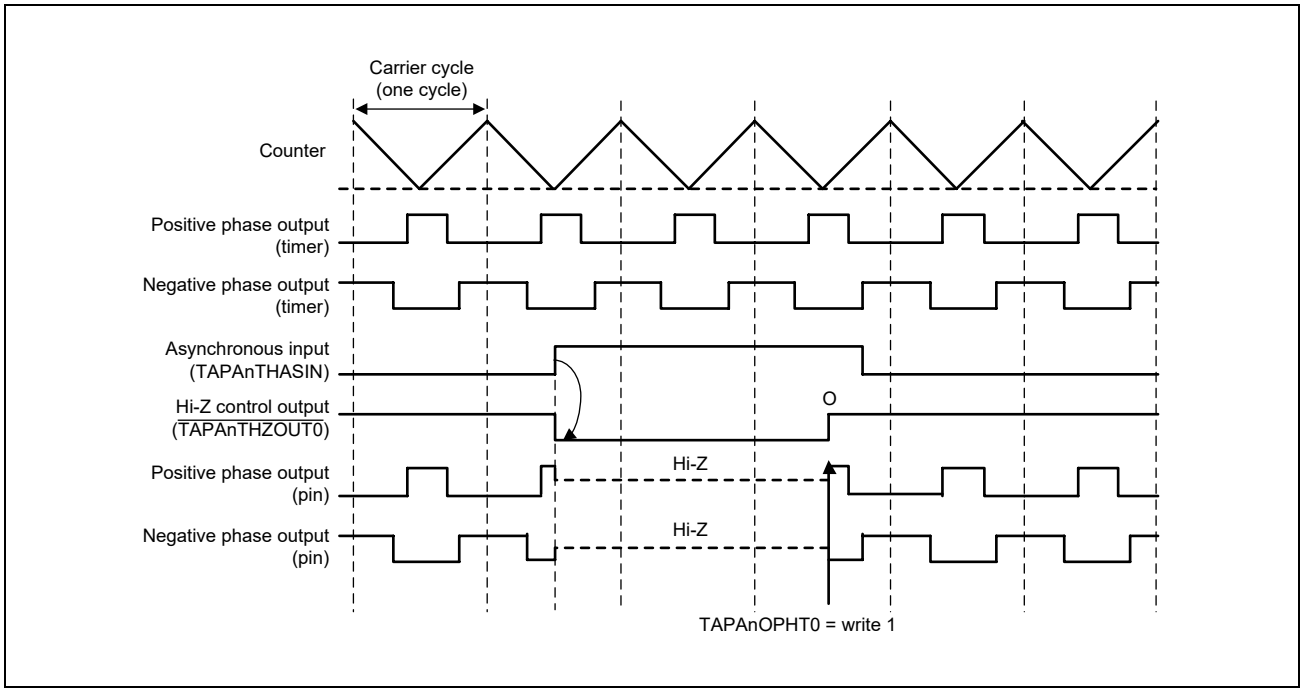


Figure 21.5 $\overline{\text{TAPAnTHZOUT0}}$ Operation when TAPAnDCM = 0, TAPAnDCP = 1, and TAPAnDCN = 0

Hi-Z Control when TAPAnCTL0.TAPAnDCM = 1, TAPAnDCP = 1, and TAPAnDCN = 0

$\overline{\text{TAPAnTHZOUT0}}$ goes to the low level in response to detection of an effective edge of the asynchronous input signal (TAPAnTHASIN). Output is forcibly stopped (by port control for Hi-Z output) as long as the $\overline{\text{TAPAnTHZOUT0}}$ output is at the low level. Writing of 1 to Hi-Z stop trigger 0 (TAPAnOPHT0) is ignored as long as the asynchronous input signal (TAPAnTHASIN) is at the active level (high because TAPAnDCP = 1).

After the asynchronous input signal (TAPAnTHASIN) is switched to the inactive level (low because TAPAnDCP = 1), $\overline{\text{TAPAnTHZOUT0}}$ goes to the high level when 1 is written to Hi-Z stop trigger 0 (TAPAnOPHT0).

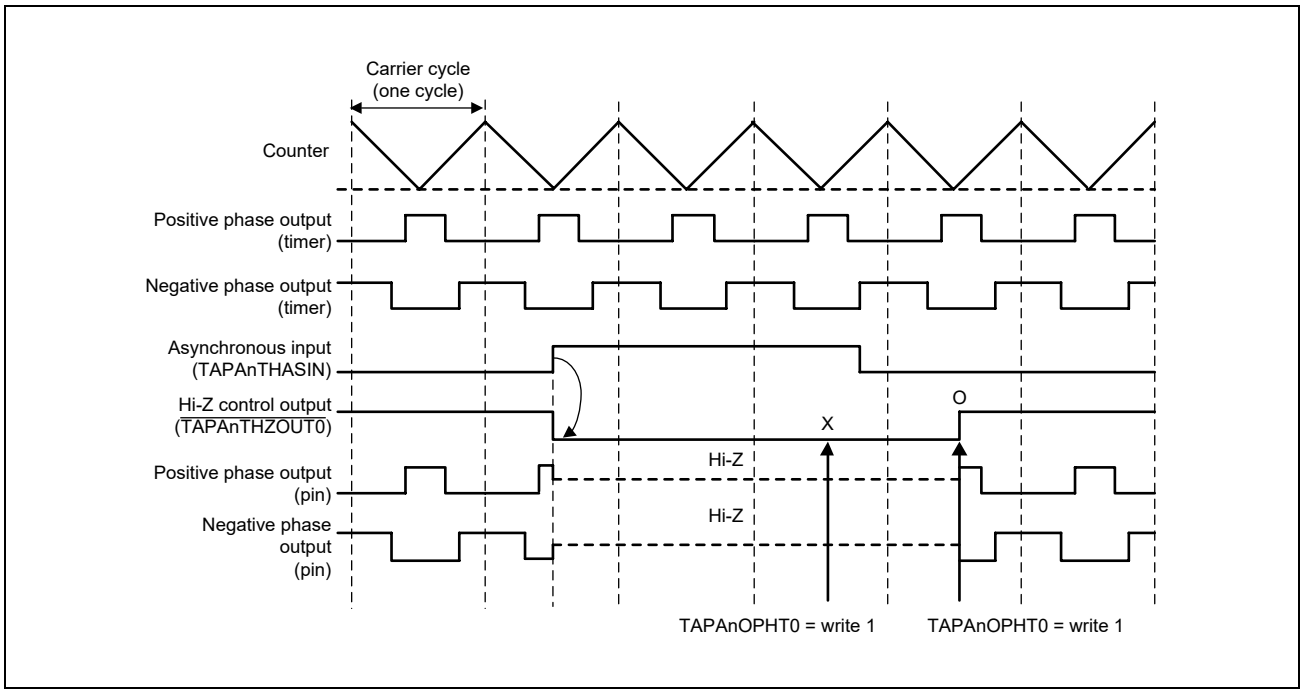


Figure 21.6 $\overline{\text{TAPAnTHZOUT0}}$ Operation when TAPAnDCM = 1, TAPAnDCP = 1, and TAPAnDCN = 0

21.4.1.4 Asynchronous Hi-Z Control Using Software Trigger

This module allows software control of the output of Hi-Z control signals.

Hi-Z start trigger 0 (TAPAnOPHS0) and Hi-Z stop trigger 0 (TAPAnOPHT0) are used to control $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$ *1, and $\overline{\text{TAPAnTHZOUT2}}$ *1.

Note 1. Not available when TAPAn (n = 3, 4, 5).

Table 21.17 Operation of the Hi-Z Start Trigger (TAPAnOPHS)

TAPAnOPHS	Operation
0/1	Writing 1 to the TAPAnOPHS0 bit places the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$, and $\overline{\text{TAPAnTHZOUT2}}$ signals at the low level.

The Hi-Z stop trigger is enabled in the following conditions.

Table 21.18 Operation of the Hi-Z Stop Trigger (TAPAnOPHT) during Hi-Z Control in Response to Asynchronous Input

TAPAnOPHT	Operation
0	Writing 1 to the TAPAnOPHT0 bit places the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$, and $\overline{\text{TAPAnTHZOUT2}}$ signals at the high level.
1	If $\overline{\text{TAPAnTHASIN}}$ is at the inactive level, writing 1 to the TAPAnOPHT0 bit places the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$, and $\overline{\text{TAPAnTHZOUT2}}$ signals at the high level. If TAPAnTHASIN is at the active level, writing of 1 to the TAPAnOPHT0 bit is ignored.

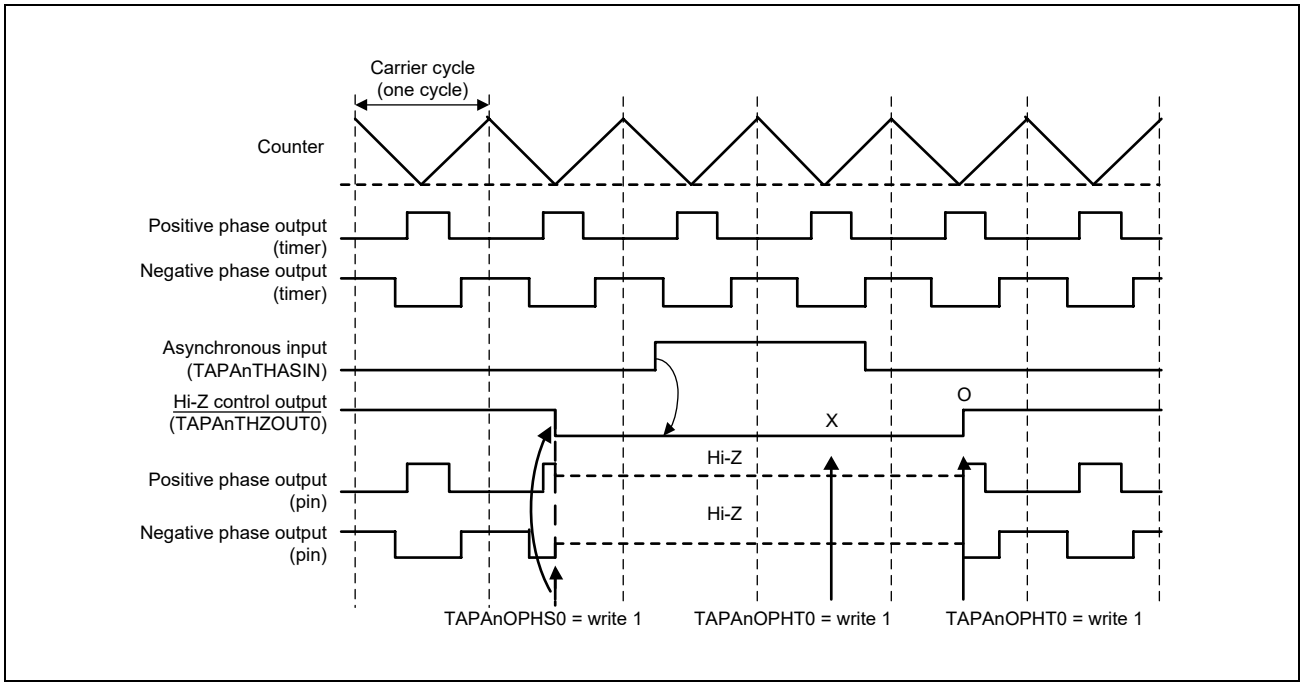


Figure 21.7 $\overline{\text{TAPAnTHZOUT0}}$ Operation when TAPAnDCM= 1, TAPAnDCP = 1, and TAPAnDCN = 0

21.4.1.5 Operating Procedure

An example of the operating procedure for Hi-Z control in response to asynchronous input is as follows (the table only covers settings for the timer option module because this operation does not depend on timer operations).

Table 21.19 Operating Procedure for Hi-Z Control in Response to Asynchronous Input

		Operation	State of TAPA
Restart operation	Initial settings	Setting in the TAPAnCTL0 register. Set TAPAnDCP and TAPAnDCN (input edge selection). Set TAPAnDCM (clearing mode selection).	Hi-Z control in response to asynchronous input is stopped (TAPAnFLG.TAPAnACE = 0).
	Starting operation	Setting in the TAPAnACWE register: Set the TAPAnACWE bit to 1. Setting in the TAPAnACTS register: Set the TAPAnACTS bit to 1.	Writing to the TAPAnACTS bit is enabled. TAPAnFLG.TAPAnACE = 1, enabling Hi-Z control in response to asynchronous input.
	During operation	To start Hi-Z control of an output from the timer: Control is by the TAPAnOPHS0 bit of TAPA Control is by the Hi-Z input signal (TAPAnTHASIN) for TAPA To stop Hi-Z control of output from the timer: Control is by the TAPAnOPHT0 bit of TAPA (if TAPAnDCM = 0) TAPAnOPHT0 is used if the Hi-Z input signal for TAPA (TAPAnTHASIN) is at the inactive level (if TAPAnDCM = 1) The state of TAPA operations can be read from the TAPAnFLG register at all times.	On detection of input of the starting edge of the Hi-Z input signal (TAPAnTHASIN) or setting of the start trigger bit (TAPAnOPHS0 = 1), the Hi-Z controller switches the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$ *1, and $\overline{\text{TAPAnTHZOUT2}}$ *1 pins to low-level output. In accord with the operating mode settings in TAPAnDCM, the Hi-Z controller switches the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$ *1, and $\overline{\text{TAPAnTHZOUT2}}$ *1 pins to high-level outputs in response to setting of the stop trigger bit (TAPAnOPHT0 = 1).
	Stopping operation	Setting in the TAPAnACWE register: Set the TAPAnACWE bit to 1. Setting in the TAPAnACTT register. Set the TAPAnACTT bit to 1.	Writing to the TAPAnACTT bit is enabled. TAPAnFLG.ACE = 0, stopping asynchronous Hi-Z control

Note 1. Not available when TAPAn (n = 3, 4, 5).

21.4.2 Selection of INT Signal Output

21.4.2.1 Configuration

This function outputs of peak or trough interrupts by using the INT and TAPAnTUDCMm signals output from the triangle-wave carrier-cycle generation channel (master) of the TAUD module.

Signals generated at cycle 0 and 1 of TAUD are used as input signals so that two sets of peak or trough interrupts are output.

21.4.2.2 Basic Operation

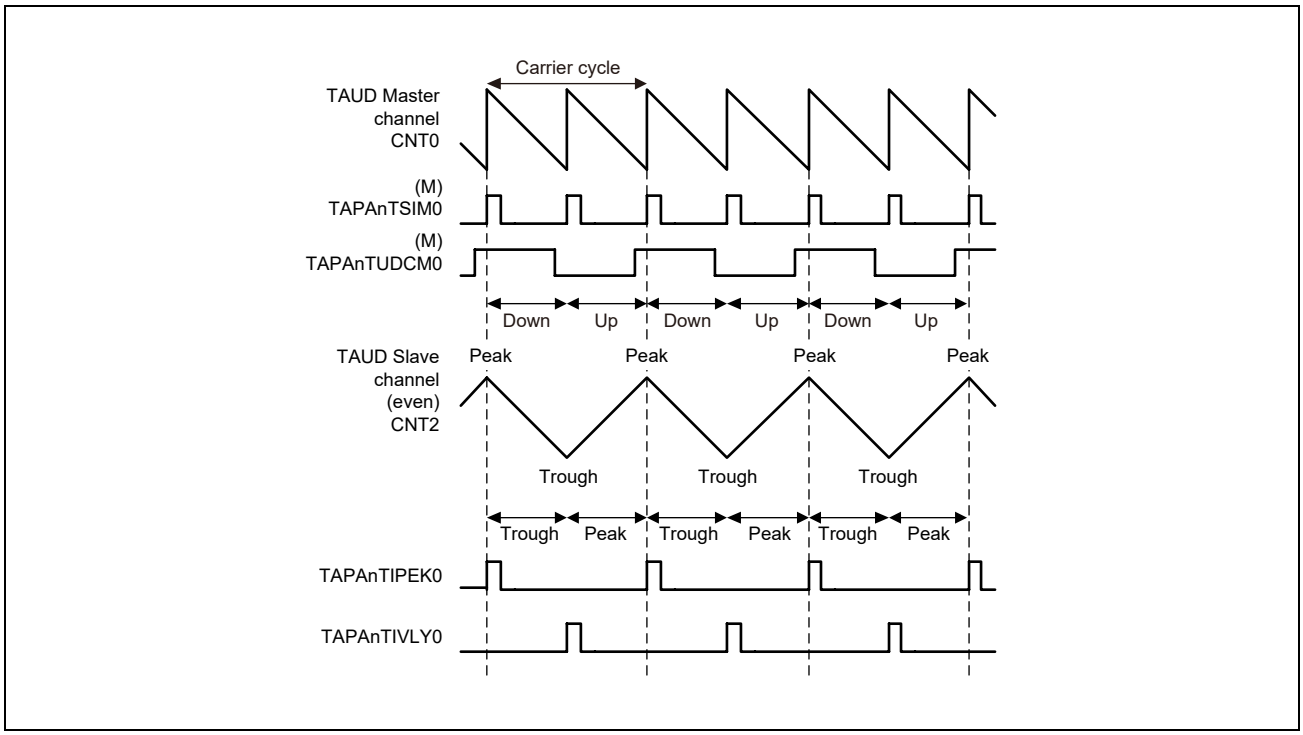


Figure 21.8 Example of Operations in Response to Master Channel 0 of TAUD

TAUD

The master channel of TAUD generates a triangle-wave carrier-cycle.

The TAUD module outputs an INT0 signal and toggles the up/down signal in accord with the setting of TAPAnTUDCM0 per half cycle of this carrier wave.

TAPA

- The TAPA module handles a TAPAnTSIM0 signal arriving while TAPAnTUDCM0 is high as a peak interrupt and outputs the TAPAnTIPEK0 signal in response.
- The TAPA module handles a TAPAnTSIM0 signal arriving while TAPAnTUDCM0 is low as a trough interrupt and outputs the TAPAnTIVLY0 signal in response.

CAUTION

Combined circuits handle output of the TAPAnTIPEK0/TAPAnTIVLY0 signals and operate regardless of the operating mode.

When the TAPAnTIPEK0/TAPAnTIVLY0 signals are not in use, they must be masked by using the registers described in **Section 6.2.3, EI Level Interrupt Mask Registers 0 to 11 (IMR0 to IMR11).**

21.4.2.3 Operating Procedure

The procedure for selecting interrupt output is as follows.

Table 21.20 Operating Procedure for Selecting Interrupt Output

	Operation	States of TAUD and TAPA
Initial settings	The TAPA module does not require initial settings.	TAUD and TAPA are stopped.
Starting operation	Start the TAUD module.	TAUD starts counting.
During operation	TAUD runs in accord with the settings for the various functions.	The interrupt output selector outputs a peak interrupt (TAPAnTIPEK0) or a trough interrupt (TAPAnTIVLY0) for control cycle 0. This is based on interrupt input (TAPAnTSIM0) and up/down input (TAPAnTUDCM0) from TAUD.
Stopping operation	Stop the TAUD module.	TAUD stops counting.

Restart operation

21.4.3 Selecting a Trigger to Start Conversion by the A/D Converter

The TAPA is capable of producing triggers to start conversion by the A/D converter (TAPAnTADOUT0 and TAPAnTADOUT1). A trigger signal is produced from the INT and TOUT signals output from the triangle-wave carrier-cycle generation channel (master) of the TAUD and the INT signal output from the channel selected for operation with the trigger to start conversion by the A/D converter.

21.4.3.1 Configuration

Table 21.21 Signals Used in Generating the TAPAnTADOUT Signals

Output Signal	Up/Down Input	Slave Match Detection Signal	Trough Interrupt Signal
TAPAnTADOUT0	TAPAnTUDCM0	TAPAnTCDENS0	TAPAnTIVLY0
TAPAnTADOUT1	TAPAnTUDCM0	TAPAnTCDENS1	TAPAnTIVLY0

Table 21.22 Operation of TAPAnCTL1.TAPAnATS[1:0] and TAPAnTADOUT0

TAPAnATS1	TAPAnATS0	Description
0	0	The INT signal from slave 0 is output as TAPATADOUT0 while master 0 of the TAUD module is in the down state.
0	1	The INT signal from slave 0 is output as TAPATADOUT0 while master 0 of the TAUD module is in the up state.
1	0	The INT signal from slave 0 of TAUD is output as TAPATADOUT0.
1	1	The INT and TAPATIVLY0 (trough interrupt signal 0) signals from slave 0 of TAUD are output as TAPAnTADOUT0.

Table 21.23 Operation of TAPAnCTL1.TAPAnATS[3:2] and TAPAnTADOUT1

TAPAnATS3	TAPAnATS2	Description
0	0	The INT signal from slave 1 is output as TAPAnTADOUT1 while master 0 of the TAUD module is in the down state.
0	1	The INT signal from slave 1 is output as TAPAnTADOUT1 while master 0 of the TAUD module is in the up state.
1	0	The INT signal from slave 1 of TAUD is output as TAPAnTADOUT1.
1	1	The INT and TAPAnTIVLY0 (trough interrupt signal 0) signals from slave 1 of TAUD are output as TAPAnTADOUT1.

21.4.3.2 Basic Operation

The following figure shows waveforms in control of A/D converter trigger output in triangle-wave PWM mode.

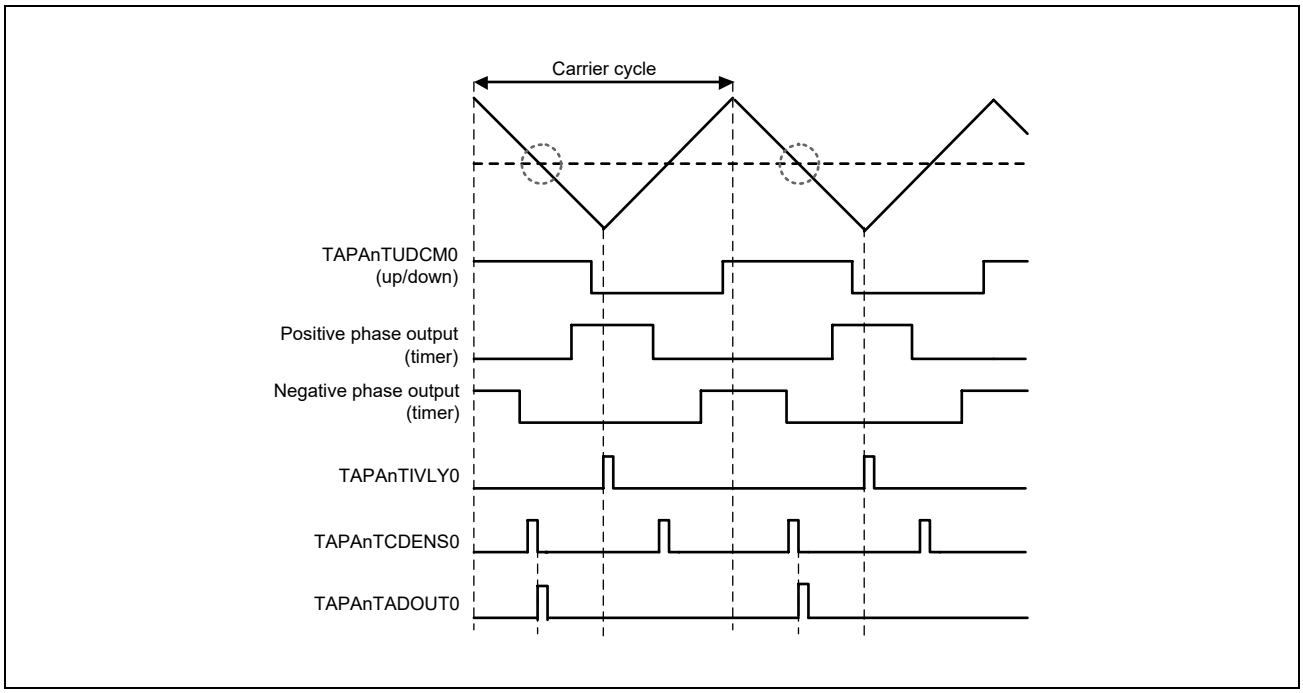


Figure 21.9 TAPAnATS = {0, 0}: INT Output while Master Channel is in the Down State

An INT signal from the slave while the master is in the down state is output as a trigger to start conversion by the A/D converter.

An INT signal from the slave while the master is in the up state is not output.

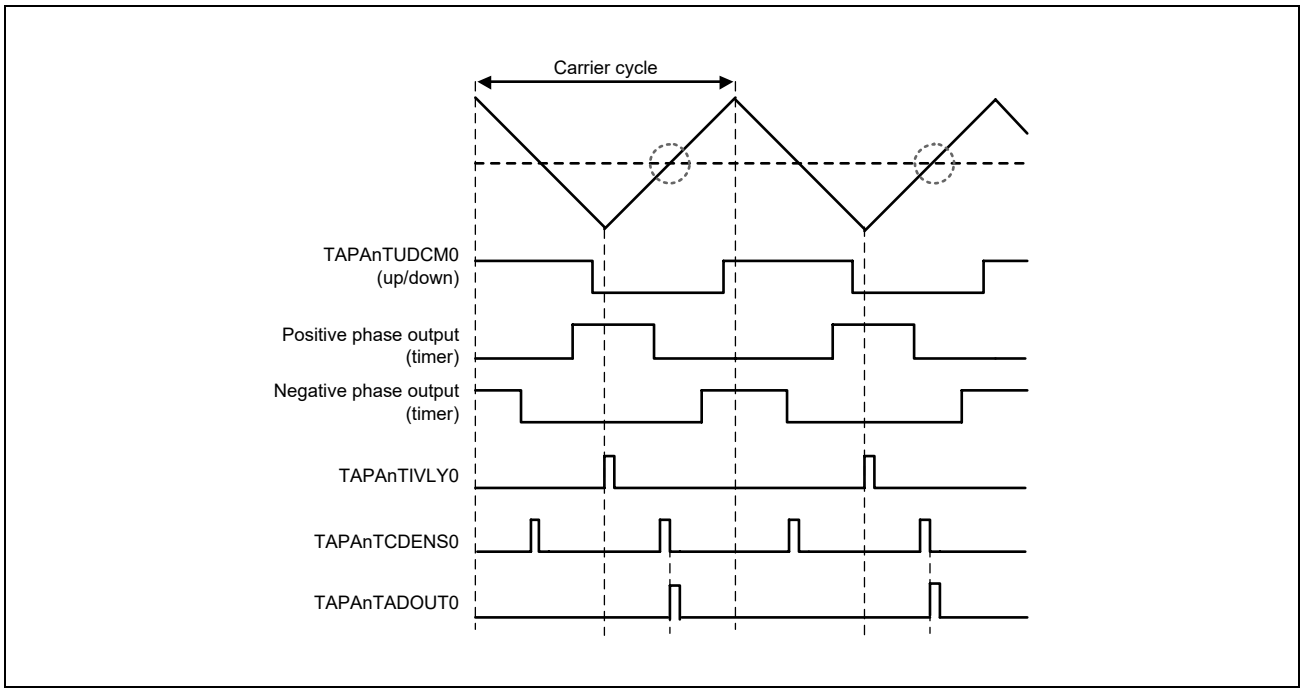


Figure 21.10 TAPAnATS = {0, 1}: INT Output while the Master Channel is in the Up State

An INT signal from the slave while the master is in the up state is output as a trigger to start conversion by the A/D converter.

An INT signal from the slave while the master is in the down state is not output.

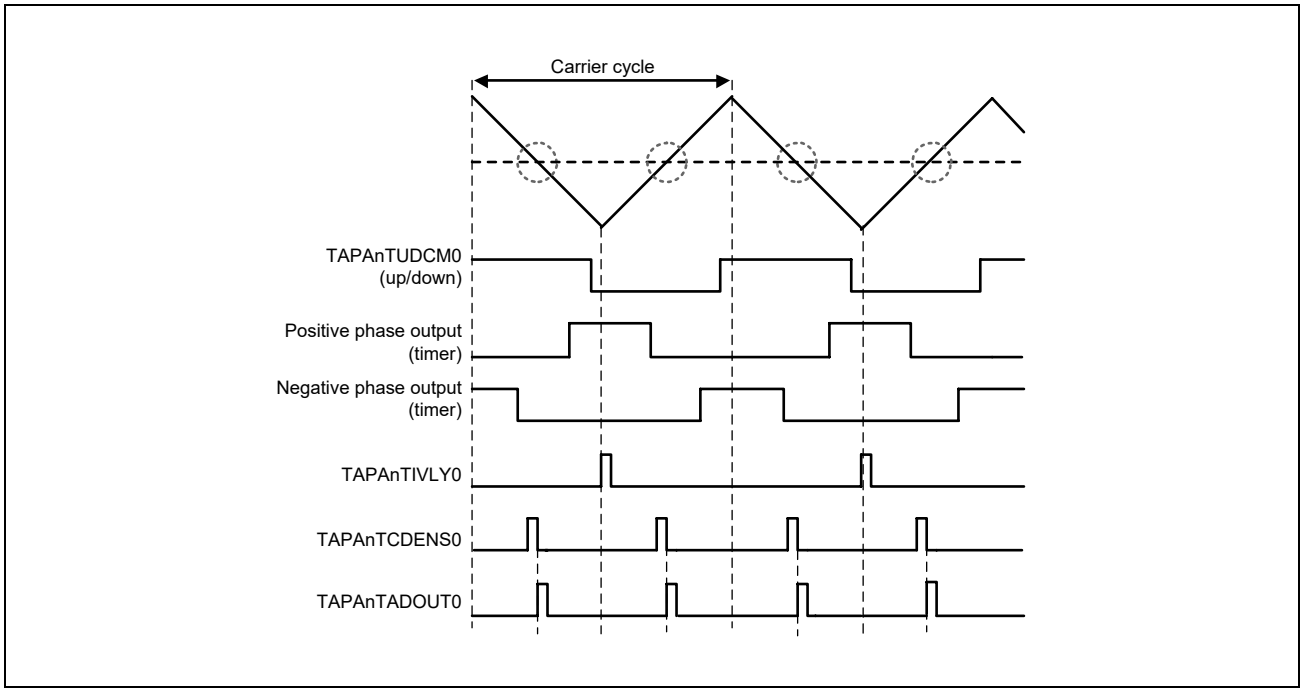


Figure 21.11 TAPAnATS = {1, 0}: INT Output while the Master Channel is in the Down or Up State

An INT signal from the slave at any time is output as a trigger to start conversion by the A/D converter.

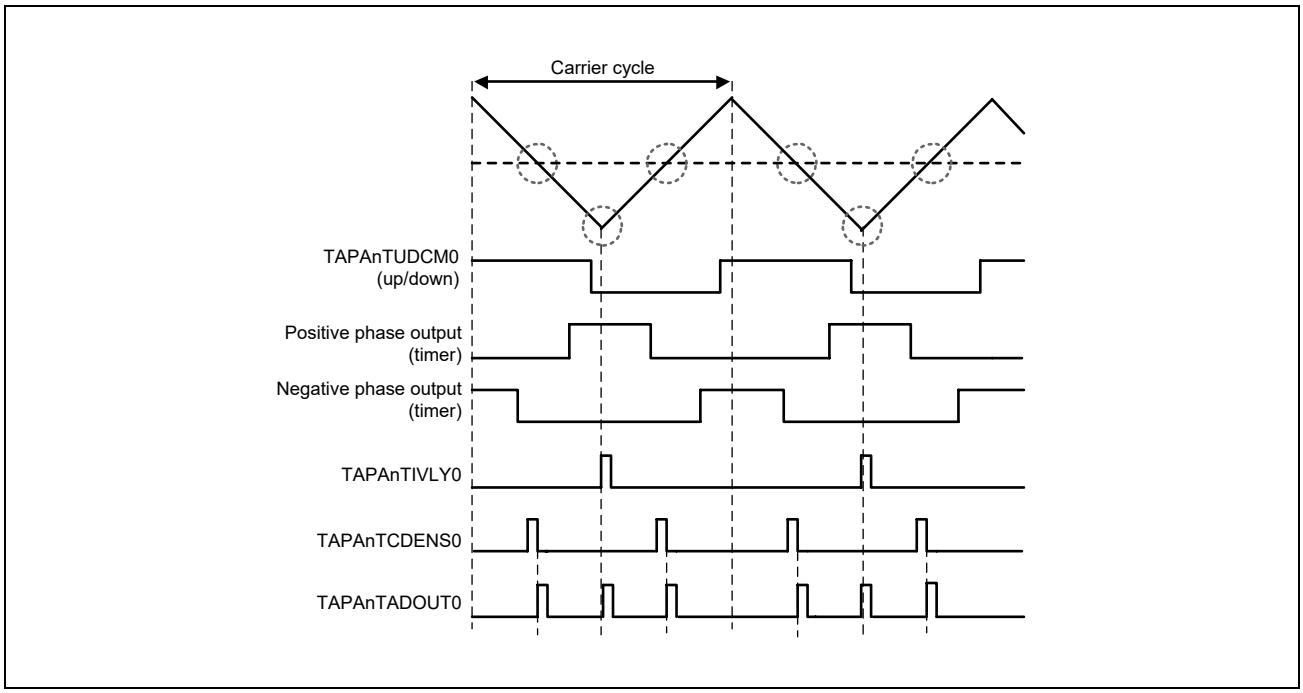


Figure 21.12 TAPAnATS = {1, 1}: INT Output and Trough Interrupt while Master Channel is in the Down or Up State

An INT signal or trough interrupt signal from the slave is output as a trigger to start conversion by the A/D converter.

21.4.3.3 Operating Procedure

The operating procedure for setting a trigger to start conversion by the A/D converter is as follows.

Table 21.24 Operation Procedure for Setting a Trigger to Start Conversion by the A/D Converter

		Operation	States of TAUD and TAPA
	Initial settings	Initialize TAUD. Determine the timer's operating mode.	TAUD and TAPA are stopped.
		Setting in the TAPAnCTL1 register. Set TAPAnATS[1:0] bits (for TAPAnTADOUT0). Set TAPAnATS[3:2] (for TAPAnTADOUT1).	
Restart operation	Starting operation	Start the TAUD module.	TAUD starts counting.
	During operation	TAUD runs in accord with the settings for the various functions.	The A/D converter conversion trigger selector outputs TAPAnTADOUT0 in accord with the settings of TAPAnATS[1:0] or TAPAnTADOUT1 in accord with the settings of TAPAnATS[3:2], based on the interrupt input (TAPAnTCDENS1 or TAPAnTCDENS0) and up/down input (TAPAnTUDCM1 or TAPAnTUDCM0) from the TAUD and the trough interrupt signal (TAPAnTIVLY1 or TAPAnTIVLY0) generated by TAPA.
	Stopping operation	TAUD is stopped.	TAUD stops counting

Section 22 Timer Pattern Buffer (TPBA)

This section contains a generic description of the Timer Pattern Buffer (TPBA).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of TPBA.

22.1 Features of RH850/C1M-A TPBA

22.1.1 Units

This LSI has the following number of TPBA units.

Each TPBA unit has one channel interface.

Table 22.1 Number of Units

Product Name	RH850/C1M-A2	RH850/C1M-A1
Number of units	2	1
Name	TPBAn (n = 0, 1)	TPBAn (n = 0)

Table 22.2 Indices

Index	Meaning
n	Throughout this section, the individual TPBA units are identified by the index “n” (n = 0, 1) (n = 0, 1 for C1M-A2; n = 0 for C1M-A1); for example, TPBAnCTL is the TPBAn control register.
m	The number of buffers are indicated by the index “m” (m = 00 to 63).

22.1.2 Register Base Address

TPBA base addresses are listed in the following table.

TPBA register addresses are given as offsets from the base addresses in general.

Table 22.3 Register Base Address

Base Address Name	Base Address
<TPBA0_base>	FFEA 0000$_H$
<TPBA1_base>	FF8A 1000$_H$*1

Note 1. Not supported for C1M-A1.

22.1.3 Clock Supply

Clock supply by and to TPBA is listed in the following table.

Table 22.4 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
TPBAn	PCLK	CLKC_HSB (non-modulated high-speed peripheral clock)

22.1.4 Interrupt Request

TPBA interrupt requests are listed in the following table.

Table 22.5 Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
TPBA0						
INTTPBA0IPRD	Period matched detection interrupt	184	—	61	—	61
INTTPBA0IDTY	Duty-cycle-matched detection interrupt	185	—	62	—	62
INTTPBA0IPAT	Number-of-patterns matched detection interrupt	186	—	63	—	63
TPBA1*2						
INTTPBA1IPRD	Period matched detection interrupt	187	—	64	—	64
INTTPBA1IDTY	Duty-cycle-matched detection interrupt	188	—	65	—	65
INTTPBA1IPAT	Number-of-patterns matched detection interrupt	189	—	66	—	66

—: No number is assigned.

Note 1. 1st: Primary channel, 2nd: Secondary channel

Note 2. Not supported for C1M-A1.

22.1.5 Reset Sources

TPBA is initialized by these reset sources.

Table 22.6 Reset Sources

Unit Name	Reset Sources
TPBAn	All reset sources

22.1.6 External Input/Output Signals

External input/output signals of TPBA are listed in the following table.

Table 22.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
TPBA0		
TPBA0O	Timer output	TPBA0O
TPBA1*1		
TPBA1O	Timer output	TPBA1O

Note 1. Not supported for C1M-A1.

22.2 Overview

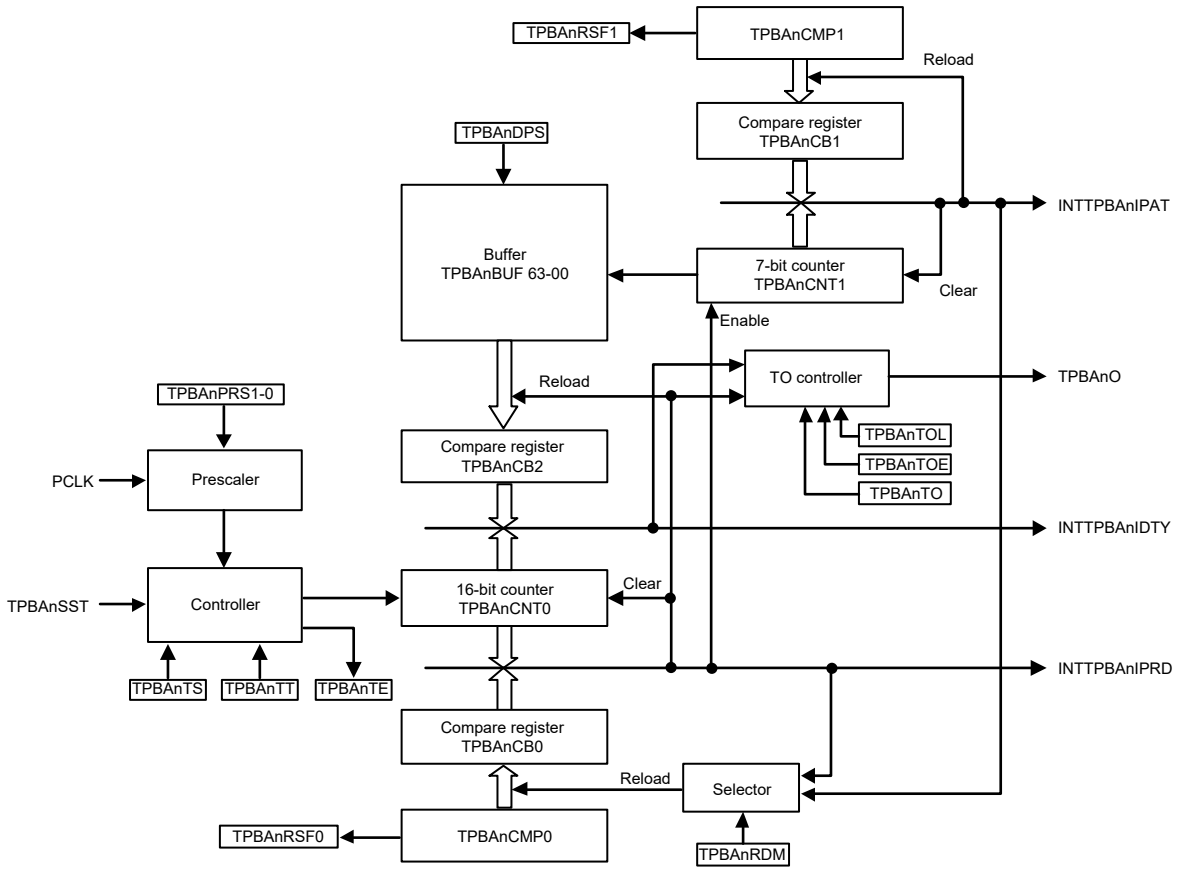
22.2.1 Functional Overview

TPBAn is a 16-bit PWM timer with the duty setting buffer.

- Count clock resolution: Min. 12.5 ns (count clock: 80 MHz)
- 16-bit counter
- 16-bit duty register
- 16-bit period setting register
- 7-bit address counter register
- 7-bit pattern number setting register
- Interrupt request signals
  - Period-matched detection interrupt
  - Duty-cycle-matched detection interrupt
  - Number-of-patterns matched detection interrupt
- Number of duty patterns
  - 64 patterns (16 bits) or 128 patterns (8 bits)
- Automatic duty generation according to the number of patterns
- Output control by software
- The count clock can be selected from PCLK, PCLK/2, PCLK/4, and PCLK/8 according to the prescaler set value.
- Synchronous start with another timer

22.2.2 Block Diagram

The following block diagram shows the main components of TPBA.



Note: TPBAnSST: Simultaneous start trigger (input from PIC1B)

Figure 22.1 Block Diagram of TPBA

22.3 Registers

22.3.1 TPBAn Registers

TPBA registers are listed in the following table.

For information on <TPBAn_base>, see **Section 22.1.2, Register Base Address**.

Table 22.8 List of Registers

Module Name	Register Name	Symbol	Address
TPBAn	TPBAn control register	TPBAnCTL	<TPBAn_base> + 200_H
TPBAn	TPBAn reload data mode register	TPBAnRDM	<TPBAn_base> + 118_H
TPBAn	TPBAn reload status register	TPBAnRSF	<TPBAn_base> + 110_H
TPBAn	TPBAn reload data trigger register	TPBAnRDT	<TPBAn_base> + 114_H
TPBAn	TPBAn timer output enable register	TPBAnTOE	<TPBAn_base> + 120_H
TPBAn	TPBAn timer output register	TPBAnTO	<TPBAn_base> + $11C_H$
TPBAn	TPBAn timer output level register	TPBAnTOL	<TPBAn_base> + 124_H
TPBAn	TPBAn period setting register	TPBAnCMP0	<TPBAn_base> + 100_H
TPBAn	TPBAn duty setting register	TPBAnBUFm	<TPBAn_base> + m × 4_H
TPBAn	TPBAn pattern number setting register	TPBAnCMP1	<TPBAn_base> + 104_H
TPBAn	TPBAn timer counter register	TPBAnCNT0	<TPBAn_base> + 108_H
TPBAn	TPBAn address counter register	TPBAnCNT1	<TPBAn_base> + $10C_H$
TPBAn	TPBAn enable status register	TPBAnTE	<TPBAn_base> + 128_H
TPBAn	TPBAn start trigger register	TPBAnTS	<TPBAn_base> + $12C_H$
TPBAn	TPBAn stop trigger register	TPBAnTT	<TPBAn_base> + 130_H

22.3.2 TPBAnCTL — TPBAn Control Register

This register specifies the operation of the TPBAn.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 200_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	TPBAnPRS[1:0]		—	—	—	TPBAnDPS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W

Table 22.9 TPBAnCTL Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4, 5	TPBAnPRS[1:0]	Selects the count clock.
3 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnDPS	Selects the duty setting pattern mode. 0: 16 bits × 64 patterns mode 1: 8 bits × 128 patterns mode

TPBAnPRS1	TPBAnPRS0	Description
0	0	PCLK is selected.
0	1	PCLK/2 is selected.
1	0	PCLK/4 is selected.
1	1	PCLK/8 is selected.

CAUTION

This register should be set when the timer is stopped (TPBAnTE = 0). If this register is erroneously rewritten, set the register again after stopping the timer.

22.3.3 TPBAnRDM — TPBAn Reload Data Mode Register

This register controls the reload timing of the TPBAn period setting register and TPBAn timer output level register values.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 118_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnRDM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 22.10 TPBAnRDM Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnRDM0	Controls the reload timing of the TPBAn period setting register (TPBAnCMP0) and TPBAn timer output level register (TPBAnTOL) values. 0: Reloads the values synchronously with a number-of-patterns matched detection interrupt (INTTPBAnIPAT). 1: Reloads the values synchronously with a period-matched detection interrupt (INTTPBAnIPRD).

CAUTION

Although this register can be rewritten during operation, the rewritten value is reflected at any time. Accordingly, during operation, this register should be rewritten when the reload request flag (TPBAnRSF) is 0.

22.3.4 TPBAnRSF — TPBAn Reload Status Register

This register indicates whether or not reload requests from the corresponding registers have been generated.

Access: This register can be read in 8-bit units.

Address: <TPBAn_base> + 110_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TPBAnRSF1	TPBAnRSF0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 22.11 TPBAnRFS Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read.
1	TPBAnRSF1	Indicates whether or not a reload request from TPBAnCMP1 has been generated. 0: No reload request generated or reload completed. 1: Reload request has been generated. This bit is set to 1 when 1 is written to the TPBAnRDT1 bit in the TPBAnRDT register. This bit is cleared at the timing when reload is performed.
0	TPBAnRSF0	Indicates whether or not a reload request from TPBAnCMP0 and TPBAnTOL has been generated. 0: No reload request generated or reload completed. 1: Reload request has been generated. This bit is set to 1 when 1 is written to the TPBAnRDT0 bit in the TPBAnRDT register. This bit is cleared at the timing when reload is performed.

22.3.5 TPBAnRDT — TPBAn Reload Data Trigger Register

This register enables reload of the register values.

Access: This register can be written in 8-bit units. It is always read as 0.

Address: <TPBAn_base> + 114_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TPBAnRDT1	TPBAnRDT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W

Table 22.12 TPBAnRDT Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When writing, write the value after reset.
1	TPBAnRDT1	Enables reload of the TPBAnCMP1 values. 0: Write access is ignored. 1: Reload is enabled (TPBAnRSF1 is set to 1). The values are updated simultaneously at the next reload timing (reload).
0	TPBAnRDT0	Enables reload of the TPBAnCMP0 and TPBAnTOL values. 0: Write access is ignored. 1: Reload is enabled (TPBAnRSF0 is set to 1). The values are updated simultaneously at the next reload timing (reload).

22.3.6 TPBAnTOE — TPBAn Timer Output Enable Register

This register enables or disables the timer output.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 120_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTOE0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 22.13 TPBAnTOE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnTOE0	Enables or disables the timer output (TPBAnO). 0: Disables the timer output based on counter operation. 1: Enables the timer output based on counter operation. • When the timer output is disabled, the level specified in TPBAnTO is output from the TPBAnO pin, and can be controlled by software. • When the timer output is enabled, TPBAnTO is set or cleared by the timer operation, and a PWM signal is output. Write access is prohibited (ignored).

22.3.7 TPBAnTO — TPBAn Timer Output Register

This register controls or reads timer output level.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + $11C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTO0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 22.14 TPBAnTO Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnTO0	Sets or indicates the output level of TPBAnO pin • When the timer output is disabled (TPBAnTOE.TPBAnTOE0 = 0) 0: Outputs low level. 1: Outputs high level. The output level can be controlled by rewriting this register during stopping of the timer output. • When the timer output is enabled (TPBAnTOE.TPBAnTOE0 = 1) 0: Low level is being output by the timer output. 1: High level is being output by the timer output. When the timer output is enabled, rewrite to this register is ignored.

22.3.8 TPBAnTOL — TPBAn Timer Output Level Register

This register controls the timer output level.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 124_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTOL0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 22.15 TPBAnTOL Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnTOL0	Specifies the active level of the timer output. 0: High 1: Low • Setting of this bit is enabled when the timer output is enabled (TPBAnTOE.TPBAnTOE0 = 1). • Setting of this bit is reflected when the timer output is started, and change of the output level is reflected at the next reload timing.

CAUTION

This register is a register to be reloaded. Rewrite during timer operation is reflected at the next reload timing. For details on reload, see **Section 22.4.2, Compare Register Rewrite Operation**.

22.3.9 TPBAnCMP0 — TPBAn Period Setting Register

This is a 16-bit compare register for setting the PWM period.

Access: This register can be read/written in 16-bit units.

Address: <TPBAn_base> + 100_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPBAnCMP0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.16 TPBAnCMP0 Register Setting

Operating Mode	PWM Period	Minimum Value (Period)	Maximum Value (Period)
8 bits	TPBAnCMP0 + 1	1	100_H
16 bits	TPBAnCMP0 + 1	1	10000_H

CAUTION

- The PWM period is (TPBAnCMP0 + 1) count clock periods. Accordingly, for PWM output with 100% duty cycle, the maximum settable value is $FFFE_H$ (FE_H).
- This register is a register to be reloaded. Rewrite during timer operation is reflected at the next reload timing. For details on reload, see **Section 22.4.2, Compare Register Rewrite Operation**.

22.3.10 TPBAnBUFm — TPBAn Duty Setting Register

This register is a 16 × 64 buffer register for duty setting.

Access: This register can be read/written in 16-bit units.

Address: <TPBAn_base> + 000_H to $0FC_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPBAnBUFm															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.17 TPBAnBUFm Register Contents

Bit Position	Bit Name	Function
15 to 0	TPBAnBUFm15 to TPBAnBUFm0	Sets the duty value. This register can set the duty value either in 16 bits × 64 patterns mode (TPBAnDPS = 0) or 8 bits × 128 patterns mode (TPBAnDPS = 1) by setting the TPBAnDPS bit. In either mode, this register is accessed in 16-bit units by the CPU. For details, see **Section 22.4.3, Duty Rewrite Operation.**

CAUTION

The value set to this register is transferred to the duty setting buffer register (TPBAnCB2) synchronously with a period-matched detection interrupt (INTTPBAnIPRD). Rewrite during timer operation is reflected at any time. For details, see **Section 22.4.3, Duty Rewrite Operation**.

- When duty setting register with 8 bits × 128 patterns is used, the duty is set in the range from 00_H to FF_H.
  The formula to output a waveform of duty 100% is: TPBAnBUFm = TPBAnCMP0 + 1 ≤ $00FF_H$. Therefore, when PWM output of duty 100% is required, the maximum value of TPBAnCMP0 is $00FE_H$. When TPBAnBUFm is greater than TPBAnCMP0 + 1, duty value exceeds 100%, but the output becomes 100% in total.
- When duty setting register with 16 bits × 64 patterns is used, the duty is set in the range from 0000_H to $FFFF_H$.
  The formula to output a waveform of duty 100% is: TPBAnBUFm = TPBAnCMP0 + 1 ≤ $FFFF_H$. Therefore, when PWM output of duty 100% is required, the maximum value of TPBAnCMP0 is $FFFE_H$. When TPBAnBUFm is greater than TPBAnCMP0 + 1, the duty value exceeds 100%, but the output becomes 100% in total.

22.3.11 TPBAnCMP1 — TPBAn Pattern Number Setting Register

This register sets the number of PWM output patterns.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 104_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	TPBAnCMP1						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.18 TPBAnCMP1 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 0	TPBAnCMP1 [6:0]	Sets the number of patterns within the following range. TPBAnDPS = 0: 0 to 63 TPBAnDPS = 1: 0 to 127

CAUTION

- This register is a register to be reloaded. Rewrite during timer operation is reflected at the next reload timing. For details on reload, see **Section 22.4.2, Compare Register Rewrite Operation**.
- If 64 or a greater number is set as the number of patterns when the duty setting pattern is in 16 bits × 64 patterns mode (TPBAnDPS = 0), the address pointer changes from 63 to 00, and the duty value is transferred from 00 again. A number-of-patterns matched detection interrupt signal (INTTPBAnIPAT) is output by the match of the specified number of patterns and the lower 7-bit values of TPBAnCNT1.

22.3.12 TPBAnCNT0 — TPBAn Timer Counter Register

This is a timer counter register that generates PWM output.

Access: This register can be read in 16-bit units.

Address: <TPBAn_base> + 108_H

Value after reset: $FFFF_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPBAnCNT0															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

16-bit counter

This register is a counter register through which the 16-bit counter value can be read.

22.3.13 TPBAnCNT1 — TPBAn Address Counter Register

This register is a counter register that indicates the address pointer to the duty setting register.

Access: This register can only be read in 8-bit units.

Address: <TPBAn_base> + $10C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	TPBAnCNT1						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

7-bit counter

This register is a counter register that indicates the address of the TPBAnBUFm register.

22.3.14 TPBAnTE — TPBAn Enable Status Register

This register indicates whether the timer counter is operating or stopped.

Access: This register can only be read in 8-bit units.

Address: <TPBAn_base> + 128_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTE0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 22.19 TPBAnTE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read.
0	TPBAnTE0	Indicates whether the timer counter is operating or stopped. 0: The timer counter is stopped. 1: The timer counter is operating. • The TPBAnTE0 bit is set to 1 when 1 is written to the TPBAnTS bit or when a synchronous start trigger is input. • The TPBAnTE0 bit is cleared to 0 when 1 is written to the TPBAnTT bit.

22.3.15 TPBAnTS — TPBAn Start Trigger Register

This register controls the timer counter start trigger.

Access: This register can only be written in 8-bit units. It is always read as 0.

Address: <TPBAn_base> + $12C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTS0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 22.20 TPBAnTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TPBAnTS0	This bit is a trigger bit that enables the timer counter. 0: Write access is ignored. 1: Starts counting (TPBAnTE = 1).

CAUTION

Write access to this register during counting (TPBAnTE = 1) is ignored.

22.3.16 TPBAnTT — TPBAn Stop Trigger Register

This register controls the timer counter stop trigger.

Access: This register can only be written in 8-bit units. It is always read as 0.

Address: <TPBAn_base> + 130_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 22.21 TPBAnTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TPBAnTT0	This bit is a trigger bit that disables the timer counter. 0: Write access is ignored. 1: Disables counting (TPBAnTE = 0).

22.4 Function

22.4.1 Basic Operation

22.4.1.1 Basic Operation of 16-Bit Counter (TPBAnCNT0)

Counting start

The 16-bit counter (TPBAnCNT0) starts counting from the value after reset $FFFF_H$.

Counter clear

The 16-bit counter is cleared by the match of the counter value and the buffer register (TPBAnCB0) set value of TPBAnCMP0.

Counter read during counting

The 16-bit counter value during counting can be read through TPBAnCNT0.

22.4.1.2 Basic Operation of 7-Bit Counter (TPBAnCNT1)

Counting start

The 7-bit counter (TPBAnCNT1) is initialized to 00_H and starts counting. Subsequently, the counter value is incremented synchronously with a period-matched detection interrupt (INTTPBAnIPRD).

Counter clear

The 7-bit counter is cleared by the match of the counter value and the buffer register (TPBAnCB1) set value of TPBAnCPM1.

Counter read during counting

The 7-bit counter value during counting can be read through TPBAnCNT1. The read value indicates TPBAnBUFm in which the duty value to be transferred next is stored.

22.4.2 Compare Register Rewrite Operation

The following registers are rewritten by reload.

- TPBAnCMP0
- TPBAnCMP1
- TPBAnTOL

Reload mode (simultaneous rewrite function)

Writing to TPBAnRDT enables reload of the registers corresponding to the set bits (sets the reload request flag (TPBAnRSF.TPBAnRSFk)), and the values of all the pertinent registers are updated simultaneously at the next reload timing (reload).

The reload timing of TPBAnCMP0 and TPBAnTOL is set by TPBAnRDM.

The reload timing of TPBAnCMP1 is the match timing (INTTPBAnIPAT) of the 7-bit counter (TPBAnCNT1) and the buffer register (TPBAnCB1) of TPBAnCMP1.

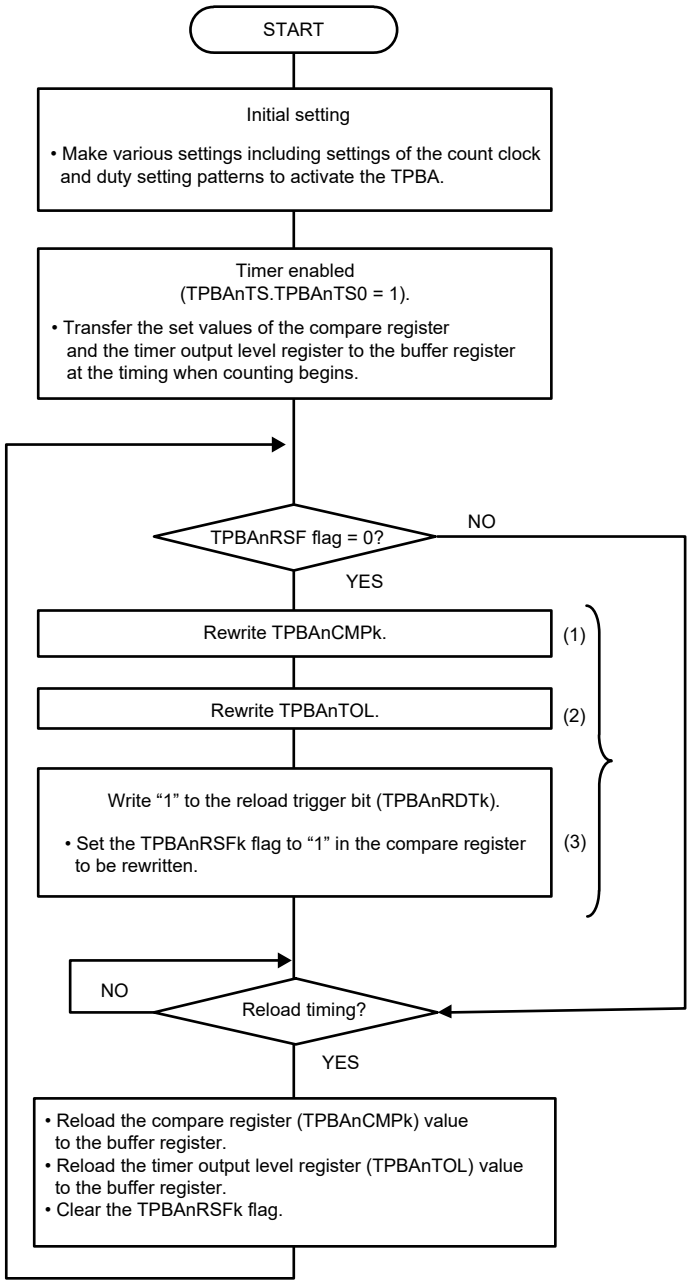
The registers to be reloaded should be rewritten when the reload request flag (TPBAnRFS.TPBAnRSFk) is 0.

NOTE

k = 0, 1

Setting Flow for Registers to Be Reloaded

The rewritten values of the registers to be reloaded (TPBAnCMP0, TPBAnCMP1, and TPBAnTOL) can be transferred to the respective buffer registers simultaneously at the reload timing.



Remark: k = 0, 1

Note: Setting the TPBAnRDTk bit to 1 enables reload. Accordingly, the TPBAnRDTk bit should be rewritten after the registers to be reloaded have been rewritten.

Figure 22.2 Basic Operation Flow of Reload (Simultaneous Rewrite Function)

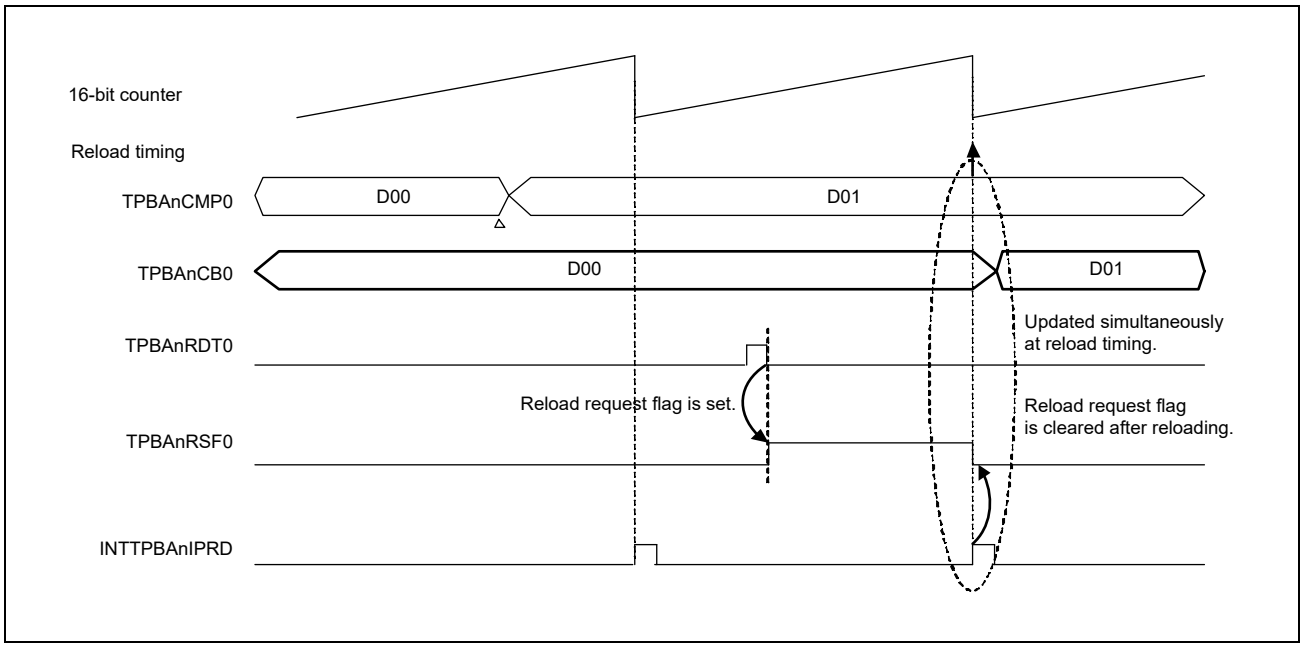


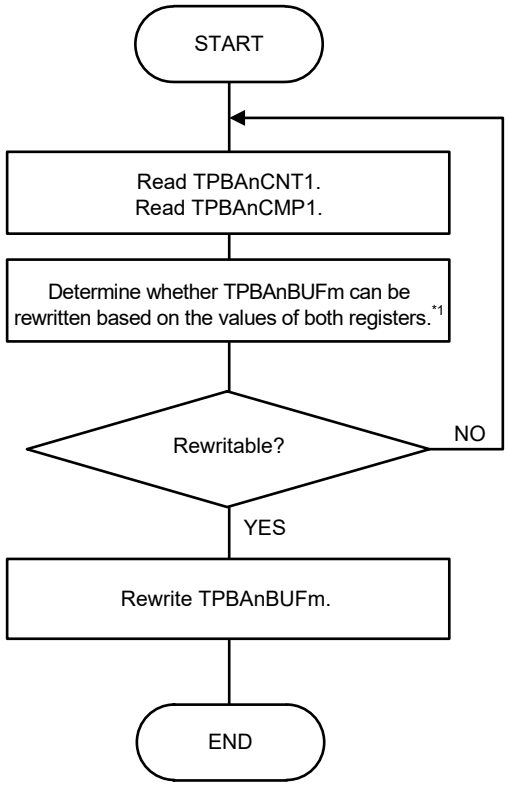
Figure 22.3 Simultaneous Rewrite Timing (TPBAnDPS = 0, TPBAnRDM = 0, and TPBAnTOL = 0)

22.4.3 Duty Rewrite Operation

TPBAnBUFm can be rewritten during operation.

The rewritten setting is reflected immediately.

22.4.3.1 TPBAnBUFm Setting Flow



Note 1. Since TPBAnBUFm can be written at any time, it is recommended that this register is rewritten during last duty cycle output (TPBAnCNT1 = 00_H) of the pattern period (the number of patterns specified by TPBAnCMP1).

Figure 22.4 Basic Rewrite Flow of TPBAnBUFm

22.4.3.2 Access to TPBAnBUFm

TPBAnBUFm is accessed in 16 bit units.

The following shows the access in 16 bits × 64 patterns mode and the access in 8 bits × 128 patterns mode.

- In 16 bits × 64 patterns mode (TPBAnDPS = 0)
  This register is accessed by the CPU in units of one 16-bit pattern.

15 0	
Pattern 64	$00FC_H$
Pattern 63	$00F8_H$
:	:
Pattern 3	0008_H
Pattern 2	0004_H
Pattern 1	0000_H

- In 8 bits × 128 patterns mode (TPBAnDPS = 1)
  This register is accessed by the CPU in units of two 8-bit patterns.

15 8	7 0	
Pattern 128	Pattern127	$00FC_H$
Pattern 126	Pattern125	$00F8_H$
:	:	:
Pattern 6	Pattern 5	0008_H
Pattern 4	Pattern 3	0004_H
Pattern 2	Pattern 1	0000_H

22.4.3.3 Relationship between TPBAnCNT1 Read Value and TPBAnBUFm

The duty value of the currently output PWM waveform can be obtained by reading the TPBAnCNT1 count value during operation. TPBAnBUFm in which the currently output duty value is stored can be found by one of the following formulas.

	Formula		
TPBAnDPS Bit	TPBAnCNT1 ≠ 00_H		TPBAnCNT1 = 00_H
0: 16 bits × 64 patterns mode	TPBAnCNT1 – 01_H*1		TPBAnCMP1*2
1: 8 bits × 128 patterns mode	TPBAnCNT1 value is an odd number	TPBAnCNT1 / 2*3	TPBAnCMP1 / 2*5
	TPBAnCNT1 value is an even number	(TPBAnCNT1 / 2) – 01_H*4	

Note 1. When TPBAnDPS = 0 and the TPBAnCNT1 ≠ 00_H
The applicable register is found by the formula TPBAnCNT1 – 01_H.
(Example) When TPBAnCNT1 = 08_H: 08_H – 01_H = 07_H → TPBA0BUF07

Note 2. When TPBAnDPS = 0 and the TPBAnCNT1 = 00_H
The applicable register is found by the TPBAnCMP1 value.
(Example) When TPBAnCMP1 = 08_H: TPBAnBUF08

Note 3. When TPBAnDPS = 1 and the TPBAnCNT1 = an odd number
The applicable register is found by the formula TPBAnCNT1 / 2
(Example) When TPBAnCNT1 = 07_H: 07_H / 02_H = 03_H → TPBAnBUF03 (lower 8 bits)

Note 4. When TPBAnDPS = 1 and the TPBAnCNT1 = an even number
The applicable register is found by the formula (TPBAnCNT1 / 2) – 01_H.
(Example) When TPBAnCNT1 = 08_H: (08_H / 02_H) – 01_H = 03_H → TPBAnBUF03 (upper 8 bits)

Note 5. When TPBAnDPS = 1 and the TPBAnCNT1 = 00_H
The applicable register is found by the formula TPBAnCMP1 / 2.
(Example) When TPBAnCMP1 = 08_H: 08_H / 2 = 04_H → TPBAnBUF04 (lower 8 bits)

22.4.4 Basic Operation Example

Overview

A PWM signal is output from the TPBAnO pin according to the PWM period set in the TPBAnCMP0 register and duty cycle set in the TPBAnBUF00 to TPBAnBUF63 registers.

Prerequisites

- Select 16 bits × 64 patterns mode or 8 bits × 128 patterns mode by setting TPBAnDPS.
- Set the duty cycle to TPBAnBUF00 to TPBAnBUF63.
- Set the number of patterns to TPBAnCMP1.

Functional description

Set the PWM period, the number of patterns, duty cycle, and level to be output. Set TPBAnTS.TPBAnTS0 = 1 (or input a synchronous start trigger) to start incrementing the timer counter value.

The TPBAnO output is set to the active level at the same time the counting begins. TPBAnCNT1 is incremented, and points to the address of the buffer in which the subsequent duty value is stored.

The output is set to the inactive level by the match of the 16-bit counter and the TPBAnBUFm buffer register (TPBAnCB2).

The duty value is then transferred from TPBAnBUFm to the buffer register (TPBAnCB2) by the match of the 16-bit counter and the TPBAnCMP0 buffer register (TPBAnCB0). Then, TPBAnCNT1 is incremented, and a period-matched detection interrupt (INTTPBAnIPRD) is generated. The TPBAnO output is set to the active level after one count clock.

During counting, a duty-cycle-matched detection interrupt (INTTPBAnIDTY) is generated by the match of the 16-bit counter and the buffer register (TPBAnCB2) of TPBAnBUFm.

A number-of-patterns matched detection interrupt (INTTPBAnIPAT) is generated by the match of the 7-bit counter and the TPBAnCMP1 buffer register (TPBAnCB1).

22.4.4.1 List of Operations

Table 22.22 16-Bit Counter Function

Operation		Setting Condition
16-bit counter	Start	Writing 1 to TPBAnTS or set 1 to simultaneous start trigger.
	Clear	Compare match of TPBAnCMP0 buffer register and 16-bit counter
	Stop	Writing 1 to TPBAnTT

Table 22.23 7-Bit Counter Function

Operation		Setting Condition
7-bit counter	Start	Writing 1 to TPBAnTS or set 1 to simultaneous start trigger.
	Clear	Compare match of TPBAnCMP1 buffer register and 7-bit counter
	Stop	Writing 1 to TPBAnTT

Table 22.24 Functions of Compare Registers and Buffer Registers

Register (Data)	Buffer Register	Rewrite Method	Rewrite During Operation	Function
TPBAnCMP0	TPBAnCB0	Reload	Possible	Setting period
TPBAnCMP1	TPBAnCB1	Reload	Possible	Setting number of patterns
TPBAnBUFm	TPBAnCB2	Rewrite at any time	Possible	Setting duty
TPBAnTOL	TPBAnTOLB	Reload	Possible	Setting output level

Buffer Registers

The registers that specify period, the number of patterns, duty, and timer output level consist of data registers that a user can directly set and buffer registers that a user cannot directly set.

Table 22.25 Timer Output Function

Pin	Function
TPBAnO	• When output is enabled (TPBAnTOE = 01_H) PWM output by compare match of the TPBAnBUFm buffer register (TPBAnCB2) and the 16-bit counter • When output is disabled (TPBAnTOE = 00_H) TPBAnTO set value

Table 22.26 Interrupt Requests

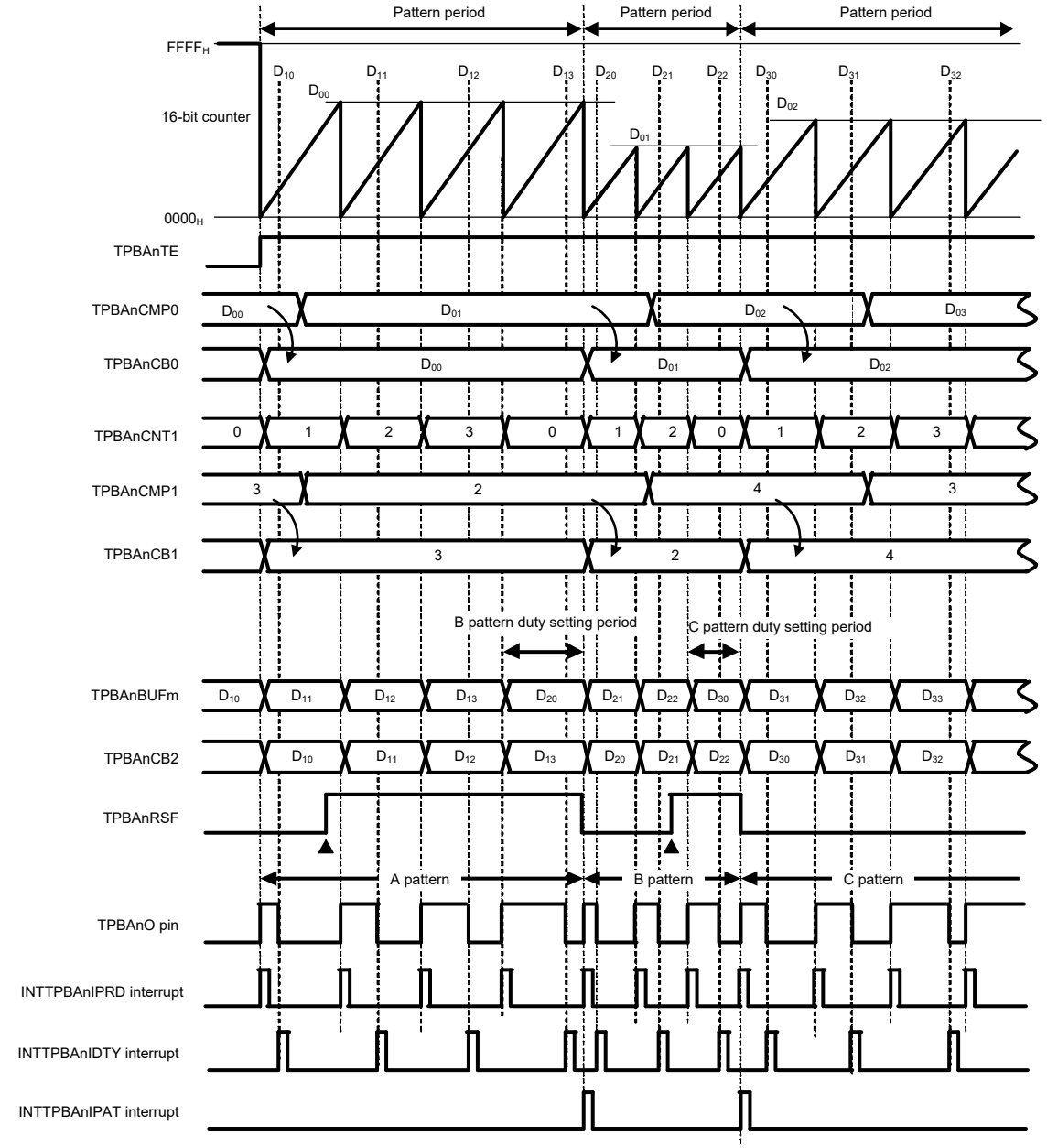
Interrupt	Function
INTTPBAnIPRD	Period-matched detection interrupt
INTTPBAnIDTY	Duty-cycle-matched detection interrupt
INTTPBAnIPAT	Number-of-patterns matched detection interrupt

Table 22.27 Compare Match Timing

Compare Match	Timing
TPBAnCMP0	When the value in the 16-bit counter matches with the value in TPBAnCMP0 and changed to 0000_H.
TPBAnCMP1	When the value in the 7-bit counter matches with the value in TPBAnCMP1 and changed to 01_H.
TPBAnBUFm	When the value in the 16-bit counter matches with the value in the buffer register (TPBAnCB2).

Table 22.28 Example of Setting Each Timer Output Condition

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TPBAnO	PWM output	(TPBAnCMP0 + 1) × count clock	Outputs an inactive level throughout one period (duty cycle 0%).	TPBAnBUFm = 0000_H
			Outputs an active level of one count clock in one period.	TPBAnBUFm = 0001_H
			Outputs an inactive level of one count clock in one period	TPBAnBUFm = TPBAnCMP0
			Outputs an active level throughout one period (duty cycle 100%).	TPBAnBUFm ≥ TPBAnCMP0 + 1



Note 1. PPWM duty cycle = (TPBAnBUFm) / (TPBAnCMP0 set value + 1)
PWM period = (TPBAnCMP0 set value + 1) × (count clock period)

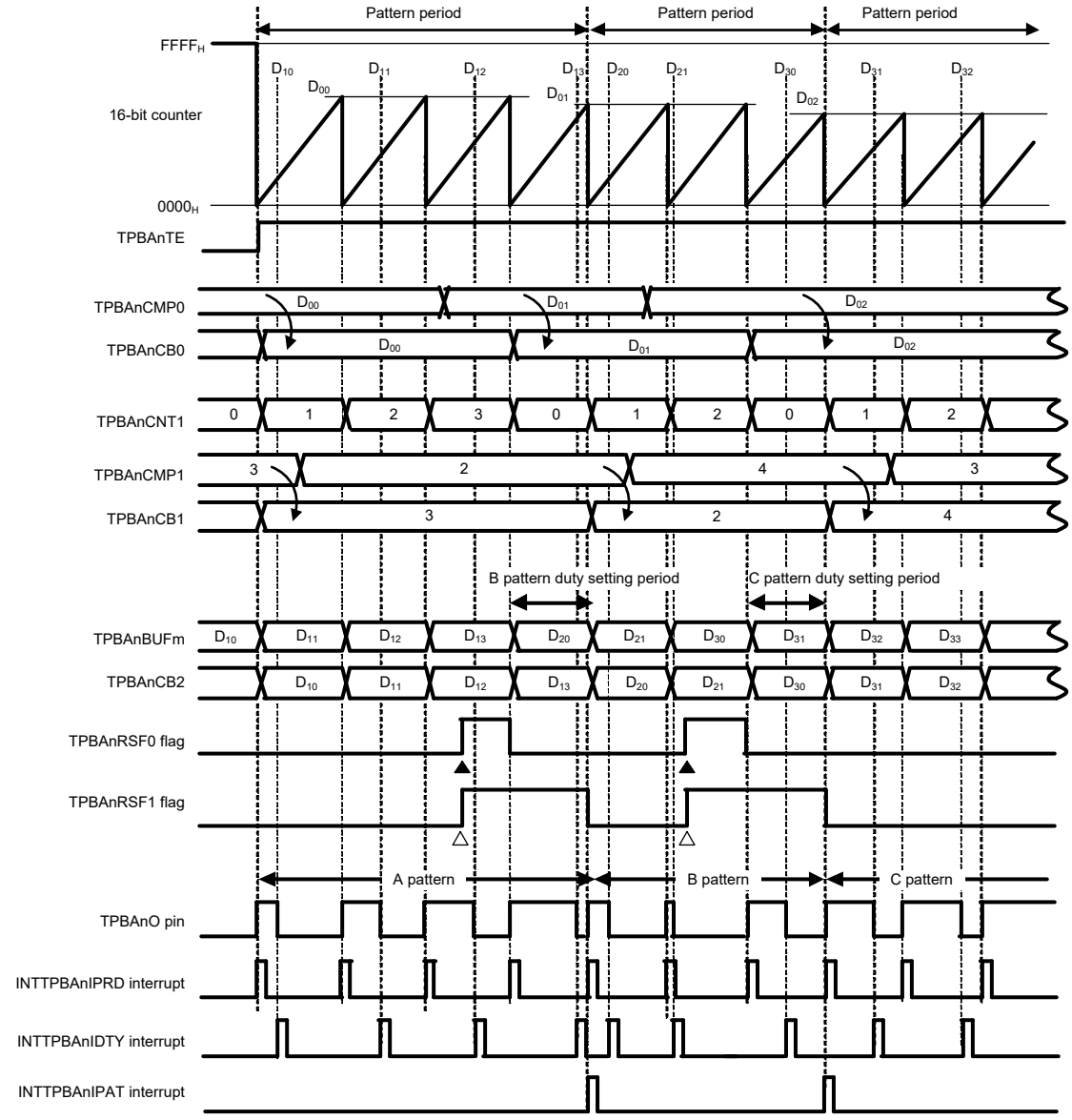
Note 2. ▲: Writing 1 to TPBAnRDT

Figure 22.5 Example of Basic Timing (1/2)

CAUTION

TPBAnO outputs active level 1 count clock after output of INTTPBAnIPRD and outputs inactive level at INTTPBAnIDTY output timing.

When a number-of-patterns matched detection interrupt is used as a trigger of the TPBAnCMP0 and TPBAnTOL reload timing (TPBAnIRDM.TPBAnRDM0 = 0 and TPBAnTOL = 0)



Note 3. PWM duty cycle = (TPBAnBUFm) / (TPBAnCMP0 set value + 1)
PWM period = (TPBAnCMP0 set value + 1) × (count clock period)

Note 4. ▲: Writing 1 to TPBAnRDT

Note 5. △: Writing 1 to TPBAnRDT1

Figure 22.5 Example of Basic Timing (2/2)

CAUTION

TPBAnO outputs active level 1 count clock after output of INTTPBAnIPRD and outputs inactive level at INTTPBAnIDTY output timing.

When a period-matched detection interrupt is used as a trigger of the TPBAnCMP0 and TPBAnTOL reload timing (TPBAnIRDM.TPBAnRDM0 = 1 and TPBAnTOL = 0)

Section 23 Encoder Timer (ENCA)

This section contains a generic description of the encoder timer (ENCA).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of the ENCA.

23.1 Features of RH850/C1M-A ENCA

23.1.1 Number of Units

This microcontroller has the following number of ENCA units.

Table 23.1 Number of Units

Product Name	RH850/C1M-A
Number of Units	2
Name	ENCAn (n = 0, 1)

Table 23.2 Index

Index	Meaning
n	Throughout this section, the individual ENCA units are identified by the index “n” (n = 0, 1); for example, ENCAnCTL is the ENCAn control register.

23.1.2 Register Base Address

ENCA base addresses are listed in the following table.

ENCA register addresses are given as offsets from the base addresses in general.

Table 23.3 Register Base Address

Base Address Name	Base Address
<ENCA0_base>	FFE8 0000_H
<ENCA1_base>	FF88 1000_H

23.1.3 Clock Supply

The ENCA clock supply is shown in following table.

Table 23.4 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
ENCAn	PCLK	CLKC_HSB (non-modulated high-speed peripheral clock)

23.1.4 Interrupts and DMA / DTS

ENCA interrupt requests are listed in the following table.

Table 23.5 Interrupt Requests

Unit Interrupt Name	Outline	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
ENCA0						
INTENCA0IOV	Overflow interrupt	166	53	—	53	—
INTENCA0IUD	Underflow interrupt	168	55	—	55	—
INTENCA0I0	Compare match 0 or capture 0 interrupt	164	51	—	51	—
INTENCA0I1	Compare match 1 or capture 1 interrupt	167	54	—	54	—
INTENCA0IEC	Interrupt to indicate clearing due to clearing input from the encoder	169	56	—	56	—
ENCA1						
INTENCA1IOV	Overflow interrupt	170	57	—	57	—
INTENCA1IUD	Underflow interrupt	172	59	—	59	—
INTENCA1I0	Compare match 0 or capture 0 interrupt	165	52	—	52	—
INTENCA1I1	Compare match 1 or capture 1 interrupt	171	58	—	58	—
INTENCA1IEC	Interrupt to indicate clearing due to clearing input from the encoder	173	60	—	60	—

Note 1. 1st: Primary channel, 2nd: Secondary channel

— : No number is assigned.

23.1.5 Reset Sources

ENCA reset sources are listed in the following table. ENCA is initialized by these reset sources.

Table 23.6 Reset Sources

Unit Name	Reset Source
ENCAn	All reset sources

23.1.6 External Input/Output Signals

External input/output signals of ENCA are listed in the following table.

Table 23.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
ENCA0		
ENCA0I0	ENCA0 capture trigger input 0	ENCA0TIN0
ENCA0I1	ENCA0 capture trigger input 1*1	ENCA0TIN1
ENCA0E0	ENCA0 encoder input 0*1	ENCA0E0
ENCA0E1	ENCA0 encoder input 1*1	ENCA0E1
ENCA0EC	ENCA0 encoder clear input*1	ENCA0EC
ENCA1		
ENCA1I0	ENCA1 capture trigger input 0	ENCA1TIN0
ENCA1I1	ENCA1 capture trigger input 1*1	ENCA1TIN1
ENCA1E0	ENCA1 encoder input 0*1	ENCA1E0
ENCA1E1	ENCA1 encoder input 1*1	ENCA1E1
ENCA1EC	ENCA1 encoder clear input*1	ENCA1EC

Note 1. These signals are input via the PIC.

23.2 Overview

23.2.1 Functional Overview

- Generation of the counter control signal from the encoder input signal, and counter operation in synchronization with PCLK.
- Capture function for capturing the counter value with an external trigger signal
- Compare function for compare match judgment with the counter value
- Two capture compare registers that can be set separately for capture operation and for compare operation
- Interrupt mask function for masking the interrupt request signal output as a result of compare match judgment during compare operation
- Function for loading the value of the capture compare register to the counter upon underflow occurrence
- Encoder input signal can be applied to the timer counter clearing condition
- Edge or level for clearing the encoder input signal of the timer counter clearing condition can be selected
- Detection of counter overflow and underflow and output of error flags and error interrupts
- Five interrupts: two capture compare interrupts, one counter clear interrupt, one overflow interrupt, and one underflow interrupt.

23.2.2 Block Diagram

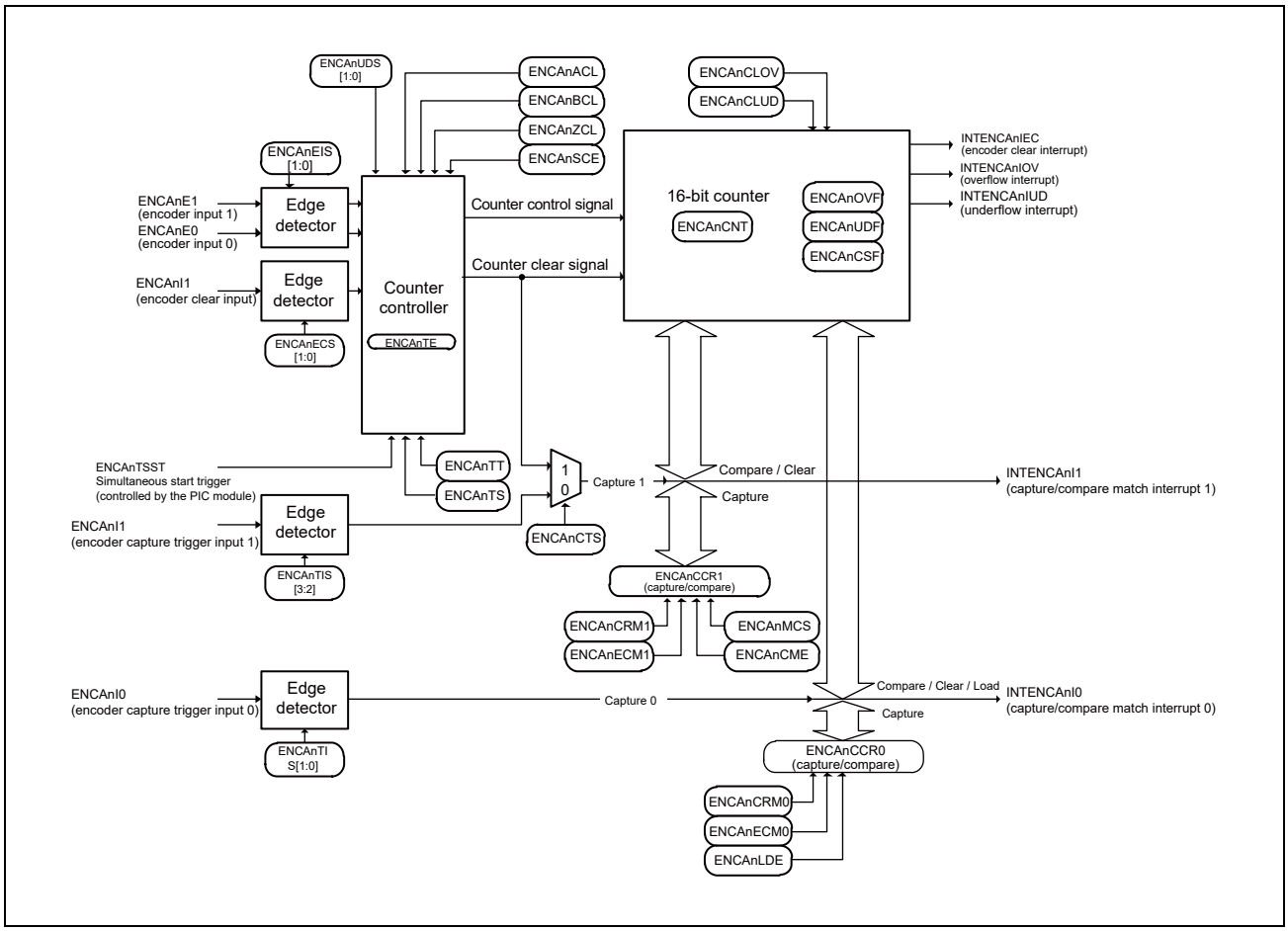


Figure 23.1 Block Diagram of ENCA

23.3 Registers

23.3.1 List of Registers

ENCA registers are listed in the following table.

For details about <ENCAn_base>, see Section **23.1.2, Register Base Address**.

Table 23.8 List of Registers

Module Name	Register Name	Symbol	Address
ENCAn	ENCAn capture compare register 0	ENCAnCCR0	<ENCAn_base>
ENCAn	ENCAn capture compare register 1	ENCAnCCR1	<ENCAn_base> + 04_H
ENCAn	ENCAn counter register	ENCAnCNT	<ENCAn_base> + 08_H
ENCAn	ENCAn status flag register	ENCAnFLG	<ENCAn_base> + $0C_H$
ENCAn	ENCAn status flag clear register	ENCAnFGC	<ENCAn_base> + 10_H
ENCAn	ENCAn timer enable status register	ENCAnTE	<ENCAn_base> + 14_H
ENCAn	ENCAn timer start trigger register	ENCAnTS	<ENCAn_base> + 18_H
ENCAn	ENCAn timer stop trigger register	ENCAnTT	<ENCAn_base> + $1C_H$
ENCAn	ENCAn I/O control register 0	ENCAnIOC0	<ENCAn_base> + 20_H
ENCAn	ENCAn control register	ENCAnCTL	<ENCAn_base> + 40_H
ENCAn	ENCAn I/O control register 1	ENCAnIOC1	<ENCAn_base> + 44_H

23.3.2 ENCAnCTL — ENCAn Control Register

This register is used to configure various operation settings for ENCAn.

Access: This register can be read/written in 16-bit units.

Writing to this register during operation is prohibited.

Address: <ENCAn_base> + 40_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ENCAn CME	ENCAn MCS	—	—	—	—	ENCAn CRM1	ENCAn CRM0	ENCAn CTS	—	—	ENCAn LDE	ENCAn ECM1	ENCAn ECM0	ENCAnUDS[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Table 23.9 ENCAnCTL Register Contents (1/2)

Bit Position	Bit Name	Function
15	ENCAnCME	Encoder Clear Mask Enable This bit is used to enable/disable masking of compare match interrupt detection when the compare function is used. 0: Disables the compare match interrupt (INTENCAnI1) mask function for the ENCAnCCR1 register 1: Enables the compare match interrupt (INTENCAnI1) mask function for the ENCAnCCR1 register. This bit is valid only when ENCAnCRM1 = 0. When this bit is set to 1, setting ENCAnECM1 to 1 is prohibited.
14	ENCAnMCS	Encoder Mask Clear Select This bit is used to select the trigger for cancelling masking of compare match interrupt detection when the compare function is used. This bit is valid only when ENCAnCRM1 = 0. 0: Masking of compare match interrupt detection is cancelled when the ENCAnCCR1 register is written. 1: Masking of compare match interrupt detection is cancelled when one of the following three operations is performed. – Timer counter clearing by encoder clear input – Timer counter clearing upon compare match between ENCAnCNT and ENCAnCCR0 when ENCAnECM0 = 1 – Loading from ENCAnCCR0 to timer counter upon underflow detection when ENCAnLDE = 1
13 to 10	Reserved	When read, the value after reset is read. When writing, write the value after reset.
9	ENCAnCRM1	ENCAnCCR1 Register Mode 0: ENCAnCCR1 used as compare register. 1: ENCAnCCR1 used as capture register.
8	ENCAnCRM0	ENCAnCCR0 Register Mode 0: ENCAnCCR0 used as compare register. 1: ENCAnCCR0 used as capture register.

Table 23.9 ENCAnCTL Register Contents (2/2)

Bit Position	Bit Name	Function
7	ENCAnCTS	ENCAnCCR1 Capture Trigger Select This is a trigger selection bit for the capture operation to the ENCAnCCR1 register. This bit is valid only when ENCAnCRM1 = 1. 0: Uses ENCAnI1 of capture trigger 1 signal as the trigger for capturing to the ENCAnCCR1 register. 1: Uses the counter clear signal selected with ENCAnSCE as the trigger for capturing to the ENCAnCCR1 register.
6, 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	ENCAnLDE	ENCAn Counter Load Enable This bit is used to enable/disable loading of the setting value to the counter upon underflow occurrence. This bit is valid only when ENCAnCRM0 = 0. When ENCAnCRM0 = 1, loading of the ENCAnCCR0 register setting value to the counter upon occurrence of an underflow is not performed, regardless of the value of this bit. 0: Disable loading of ENCAnCCR0 register setting value to counter upon occurrence of a counter underflow. 1: Enable loading of ENCAnCCR0 register setting value to counter upon occurrence of a counter underflow.
3	ENCAnECM1	Encoder Clear Mode 1 This bit is used to set the counter clear operation upon match between the counter value and ENCAnCCR1 setting value. This bit is valid only when ENCAnCRM1 = 0. 0: Does not clear the counter to 0000_H upon match of timer counter value and ENCAnCCR1 setting value. 1: Clears the counter to 0000_H upon match of timer counter value and ENCAnCCR1 setting value if the next counting operation is down-counting.
2	ENCAnECM0	Encoder Clear Mode 0 This bit is used to set the counter clear operation upon match between the counter value and ENCAnCCR0 setting value. This bit is valid only when ENCAnCRM0 = 0. 0: Does not clear the counter to 0000_H upon match of timer counter value and 0: Does not clear the counter to 0000_H upon match of timer counter value and ENCAnCCR0 setting value. 1: Clears the counter to 0000_H upon match of timer counter value and ENCAnCCR0 setting value if the next counting operation is up-counting.
1, 0	ENCAnUDS[1:0]	Up/Down Count Selection 1 and 0 These bits are the counter up/down control bits using ENCAnE0 and ENCAnE1. 00: Upon detection of effective edge of ENCAnE0, – down-count when ENCAnE1 = H, – up-count when ENCAnE1 = L 01: Upon detection of effective edge of ENCAnE0, up-count, Upon detection of effective edge of ENCAnE1, down-count 10: At rising edge of ENCAnE0, down-count At falling edge of ENCAnE0, up-count However, counting is performed only when ENCAnE1 = L. 11: Detection of both edges of ENCAnE0, ENCAnE1. Judgment of counter operation combining both detected edge and level.

23.3.3 ENCAnIOC0 — ENCAn I/O Control Register 0

This register is used to select the input edge of capture triggers 0 and 1 (ENCAnI0, ENCAnI1).

Access: This register can be read/written in 8-bit units.

Address: <ENCAn_base> + 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	ENCAnTIS[3:2]		ENCAnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 23.10 ENCAnIOC0 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3, 2	ENCAnTIS[3:2]	Input Edge Selection for Capture Trigger 1 These bits are valid only when the ENCAnCTL register's ENCAnCRM1 = 1 and ENCAnCTS = 0. All other settings of ENCAnCRM1 and ENCAnCTS are invalid. 00: No edge detection 01: Detection of the rising edge 10: Detection of the falling edge 11: Detection of both edges
1, 0	ENCAnTIS[1:0]	Input Edge Selection for Capture Trigger 0 These bits are valid only when ENCAnCTL.ENCAnCRM0 = 1. 00: No edge detection 01: Detection of the rising edge 10: Detection of the falling edge 11: Detection of both edges

23.3.4 ENCAnIOC1 — ENCAn I/O Control Register 1

This register is used to perform the clear condition setting and edge selection upon encoder input.

Access: This register can be read/written in 8-bit units.
Writing to this register during operation is prohibited.

Address: <ENCAn_base> + 44_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	ENCAnSCE	ENCAnZCL	ENCAnBCL	ENCAnACL	ENCAnECS[1:0]		ENCAnEIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.11 ENCAnIOC1 Register Contents (1/2)

Bit Position	Bit Name	Function
7	ENCAnSCE	Encoder Special-Clear Enable When setting this bit to 1, set both ENCAnUDS1 and ENCAnUDS0 to either 10_B or 11_B. The operation is not guaranteed if this bit is set to 1 with ENCAnUDS1 and ENCAnUDS0 set to 00_B or 01_B. 0: Clears the counter upon detection of ENCAnEC effective edge (set with ENCAnECS1 and ENCAnECS0). 1: Clears the counter upon detection of input level condition of ENCAnEC, ENCAnE1 and ENCAnE0 (set with ENCAnZCL bit, ENCAnBCL bit, and ENCAnACL bit).
6	ENCAnZCL	Input-Z Clear Condition Selection This bit is used to set the condition for clearing by encoder clear input (ENCAnEC) when using the encoder special clear function. This bit is valid only when ENCAnSCE = 1; it is invalid when ENCAnSCE = 0. 0: Clear condition: Low level 1: Clear condition: High level
5	ENCAnBCL	Input-B Clear Condition Selection This bit is used to set the condition for clearing by encoder input 1 (ENCAnE1) when using the encoder special clear function. This bit is valid only when ENCAnSCE = 1; it is invalid when ENCAnSCE = 0. 0: Clear condition: Low level 1: Clear condition: High level
4	ENCAnACL	Input-A Clear Condition Selection This bit is used to set the condition for clearing by encoder input 0 (ENCAnE0) when using the encoder special clear function. This bit is valid only when ENCAnSCE = 1; it is invalid when ENCAnSCE = 0. 0: Clear condition: Low level 1: Clear condition: High level

Table 23.11 ENCAnIOC1 Register Contents (2/2)

Bit Position	Bit Name	Function
3, 2	ENCAnECS[1:0]	Encoder Clear Input Edge Selection 1 and 0 These are the encoder clear input edge selection bits. These bits are valid only when ENCAnSCE = 0; they are invalid when ENCAnSCE = 1. 00: No edge detection 01: Detection of the rising edge 10: Detection of the falling edge 11: Detection of both edges
1, 0	ENCAnEIS[1:0]	Encoder Edge Input Selection 1 and 0 These are the encoder input edge selection bits. These bits are valid when ENCAnUDS1 and ENCAnUDS0 = 00_B or 01_B, and are invalid when ENCAnUDS1 and ENCAnUDS0 = 10_B or 11_B. 00: No edge detection 01: Detection of the rising edge 10: Detection of the falling edge 11: Detection of both edges

23.3.5 ENCAnFLG — ENCAn Status Flag Register

This register holds the status flags of the timer counter of ENCAn.

Access: This register can only be read in 8-bit units.

Address: <ENCAn_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENCAnCSF	ENCAnUDF	ENCAnOVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 23.12 ENCAnFLG Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read.
2	ENCAnCSF	Counter Status Flag This bit reflects the current timer counter operation. 0: Timer counter is in up-count status 1: Timer counter is in down-count status
1	ENCAnUDF	Underflow Flag This bit reflects the occurrence of an underflow during the timer counter operation. This bit is cleared at the start of counting. 0: This flag is cleared upon any of the following events: – 1 is written to ENCAnFGC.ENCAnCLUD – The flag is cleared to 0 by setting the ENCAnTS bit to 1 when ENCAnTE = 0 or by setting the simultaneous start trigger input (ENCAnTSST signal) to “High”. 1: This flag is set to 1 upon occurrence of an underflow during the encoder timer counting.
0	ENCAnOVF	Overflow Flag This bit reflects the occurrence of an overflow during the timer counter operation. This bit is cleared at the start of counting. 0: This flag is cleared upon any of the following events: – 1 is written to ENCAnFGC.ENCAnCLOV – The flag is cleared to 0 by setting the ENCAnTS bit to 1 when ENCAnTE = 0 or by setting the simultaneous start trigger input (ENCAnTSST signal) to “High”. 1: This flag is set to 1 upon occurrence of an overflow during the encoder timer counting.

23.3.6 ENCAnFGC — ENCAn Status Flag Clear Register

This register is used to clear the timer counter status flags of ENCAnFLG.

Access: This register can only be written in 8-bit units.
This register is always read as 00_H.

Address: <ENCAn_base> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ENCAnCLUD	ENCAnCLOV
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W

Table 23.13 ENCAnFGC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When writing, write the value after reset.
1	ENCAnCLUD	Underflow Flag Clear This bit clears the underflow flag. 0: Writing is ignored. 1: Clears ENCAnUDF of the ENCAnFLG register (clears underflow detection).
0	ENCAnCLOV	Overflow Flag Clear This bit clears the overflow flag. 0: Writing is ignored. 1: Clears ENCAnOVF of the ENCAnFLG register (clears overflow detection).

23.3.7 ENCAnCCR0 — ENCAn Capture Compare Register 0

This register is a 16-bit capture compare register 0.

Access: This register can be read/written in 16-bit units.
When used as a capture register, this register is only readable. Writing to this register is ignored.
When used as a compare register, this register is readable/writable.

Address: <ENCAn_base> + 00_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ENCAnCCR0[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.14 ENCAnCCR0 Register Contents

Bit Position	Bit Name	Function
15 to 0	ENCAnCCR0 [15:0]	Capture Compare Register 0 Upon occurrence of an underflow, the setting value of this register may be loaded to the counter according to the ENCAnCTL.ENCAnLDE setting. See the description of the ENCAnLDE bit in ENCA control register ENCAnCTL for details. • If ENCAnCTL.ENCAnCRM0 = 0: ENCAnCCR0 is compare register. Set the value to be compared with the timer counter value. • If ENCAnCTL.ENCAnCRM0 = 1: ENCAnCCR0 is capture register. The captured timer counter value is stored.

23.3.8 ENCAnCCR1 — ENCAn Capture Compare Register 1

This register is a 16-bit capture compare register 1.

Access: This register can be read/written in 16-bit units.
When used as a capture register, this register is only readable. Writing to this register is ignored.
When used as a compare register, this register is readable/writable.

Address: <ENCAn_base> + 04_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ENCAnCCR1[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.15 ENCAnCCR1 Register Contents

Bit Position	Bit Name	Function
15 to 0	ENCAnCCR1 [15:0]	Capture Compare Register 1 During capture operation, the trigger for capturing to this register differs according to the ENCAnCTL.ENCAnCTS setting. See the description of the ENCAnCTS bit in ENCA control register ENCAnCTL for details. • If ENCAnCTL.ENCAnCRM1 = 0: ENCAnCCR1 is compare register. Set the value to be compared with the timer counter value. • If ENCAnCTL.ENCAnCRM1 = 1: ENCAnCCR1 is capture register. The captured timer counter value is stored.

23.3.9 ENCAnCNT — ENCAn Counter Register

This register is the 16-bit timer counter register.

Access: This register can be read/written in 16-bit units.
This register can be written only when the operation is stopped.

Address: <ENCAn_base> + 08_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ENCAnCNT[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.16 ENCAnCNT Register Contents

Bit Position	Bit Name	Function
15 to 0	ENCAnCNT [15:0]	Counter Register • ENCAnTE.ENCAnTE status: 0 (initial setting): Counting stops. An arbitrary value can be set to timer counter. • ENCAnTE.ENCAnTE status: 0 → 1 (operation start)): Counting starts. Counting up or down is started with the set arbitrary value. • ENCAnTE.ENCAnTE status: 1 (operating): Counting Counting up or down is performed. • ENCAnTE.ENCAnTE status: 1 → 0 (stopped): Counting stops. The counter value immediately before the operation was stopped is held, and counting is stopped.

23.3.10 ENCAnTE — ENCAn Timer Enable Status Register

This register indicates the operating status of ENCAn.

Access: This register can only be read in 8-bit units.

Address: <ENCAn_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ENCAnTE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 23.17 ENCAnTE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read.
0	ENCAnTE	Timer Status Enable This is a status bit that indicates the operation enabled/stopped status of ENCAn. This bit is cleared to 0 when 1 is written to ENCAnTT.ENCAnTT. This bit is set to 1 when 1 is written to ENCAnTS.ENCAnTS or when the input signal of ENCAnTSST is set to the high level. 0: Operation stopped status 1: Operation enabled status

23.3.11 ENCAnTS — ENCAn Timer Start Trigger Register

This register provides the trigger bit for setting the ENCAn to the operation enabled state.

Access: This register can only be written in 8-bit units.
This register is always read as 00_H. This register can be written only when ENCAnTE.ENCAnTE is 0.

Address: <ENCAn_base> + 18_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ENCAnTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 23.18 ENCAnTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	ENCAnTS	Timer Start Trigger This is the trigger bit that sets the ENCAn to the operation enabled state. 0: Writing is ignored. 1: The ENCAn is set to the operation enabled state by setting ENCAnTE.ENCAnTE = 1.

23.3.12 ENCAnTT — ENCAn Timer Stop Trigger Register

This register provides the trigger bit for setting the ENCAn to the operation stopped state.

Access: This register can only be written in 8-bit units.
This register is always read as 00_H.

Address: <ENCAn_base> + $1C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ENCAnTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 23.19 ENCAnTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	ENCAnTT	Timer Stop Trigger This is the trigger bit that sets the ENCAn to the operation stopped state. 0: Writing is ignored. 1: Clears ENCAnTE.ENCAnTE to 0 to set the ENCAn to the counter operation stopped state.

23.4 Functions

The ENCAn operates the timer counter with counter up/down control and clear control by encoder inputs. The ENCAnCCR0 and ENCAnCCR1 registers can be used as dedicated compare registers or as dedicated capture registers.

23.4.1 Timer Counter Operation

The timer counter operations of the ENCAn are described below.

The figure below shows the operation phases. See the corresponding section with the section number for detailed descriptions on each operation.

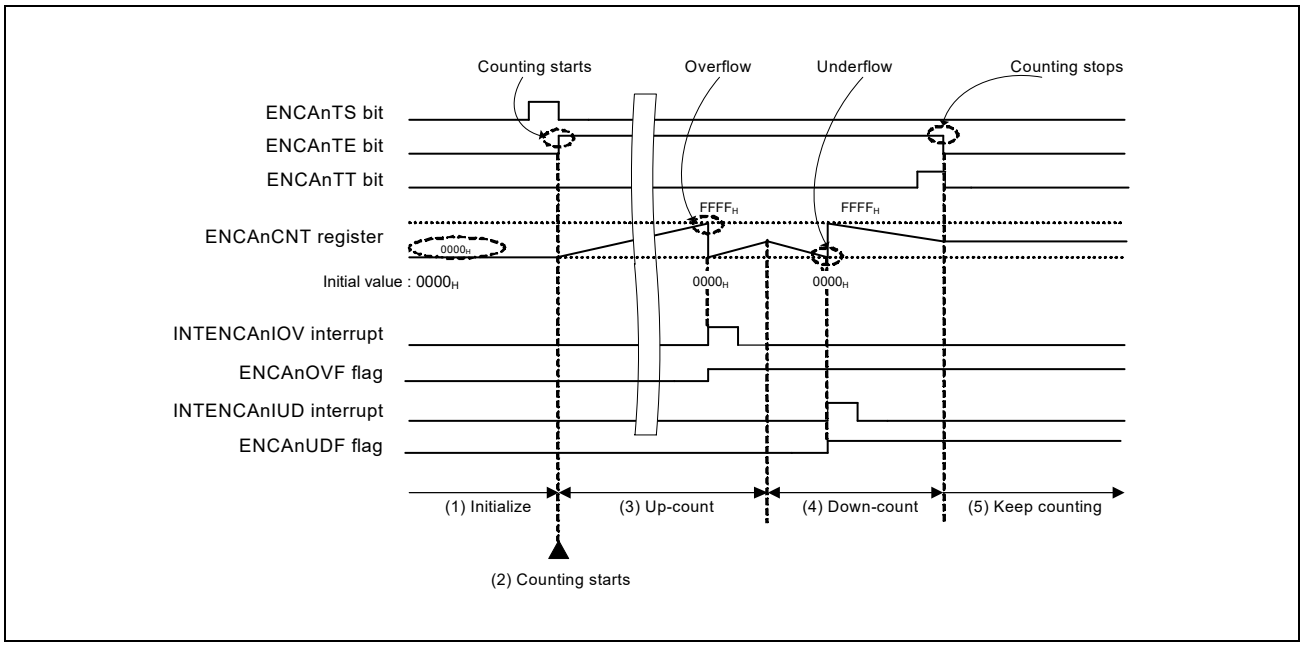


Figure 23.2 Timer Counter Initial Value Setting/Start/Stop

(1) Timer counter initial value setting

The initial value of the ENCAn counter register (ENCAnCNT) can be set in the counter operation stopped status (ENCAnTE = 0).

(2) Timer counter startup

By writing 1 to the timer start trigger bit (ENCAnTS), the timer status enable bit (ENCAnTE) is set to 1, the counter operation is enabled, and counting operation is performed upon detection of the effective edge of the encoder input.

(3) Overflow operation

An overflow occurs when up-counting is performed when the counter value is $FFFF_H$. If the counter value changes from $FFFF_H$ to 0000_H, an overflow interrupt (INTENCAnIOV) is generated, and the overflow flag (ENCAnOVF) is set to 1. The overflow flag (ENCAnOVF) is cleared to 0 when 1 is set to the overflow flag clear bit (ENCAnCLOV). For details about the operation, see **Section 23.6.6, Overflow Occurrence and Overflow Flag Clear Operation**.

(4) Underflow operation

An underflow occurs when down-counting is performed when the counter value is 0000_H. If the counter value changes from 0000_H to $FFFF_H$, an underflow interrupt (INTENCAnIUD) is generated, and the underflow flag (ENCAnUDF) is set to 1. The underflow flag (ENCAnUDF) is cleared to 0 when 1 is set to the underflow flag clear bit (ENCAnCLUD). For details about the operation, see **Section 23.6.7, Underflow Occurrence and Underflow Flag Clear Operation**.

(5) Timer counter stop

By writing 1 to the timer stop trigger bit (ENCAnTT), the timer status enable bit (ENCAnTE) is cleared to 0, and counting is stopped. At this time, the timer counter is not reset to 0000_H and holds the value before counting stops.

23.4.2 Up/Down Control of Timer Counter

Up/down control is performed by judging the phase of the encoder inputs (ENCAnE0, ENCAnE1) according to the settings of the ENCAnUDS1 and ENCAnUDS0 bits.

23.4.2.1 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 00_B

Table 23.20 When ENCAnUDS1 and ENCAnUDS0 Bits = 00_B

ENCAnUDS1	ENCAnUDS0	Operation Description		
		ENCAnE0 Pin	ENCAnE1 Pin	Counting Operation
0	0	Rising edge	High level	Down
		Falling edge		
		Both edges		
		Rising edge	Low level	Up
		Falling edge		
		Both edges		

The effective edge of the signal on the ENCAnE0 pin is specified by setting the ENCAnEIS1 and ENCAnEIS0 bits.

The timer starts counting up/down according to the condition of the edges and levels of the ENCAnE0 and ENCAnE1 pins.

The following timing chart shows the counter operation when the ENCAnUDS1 and ENCAnUDS0 bits = 00_B.

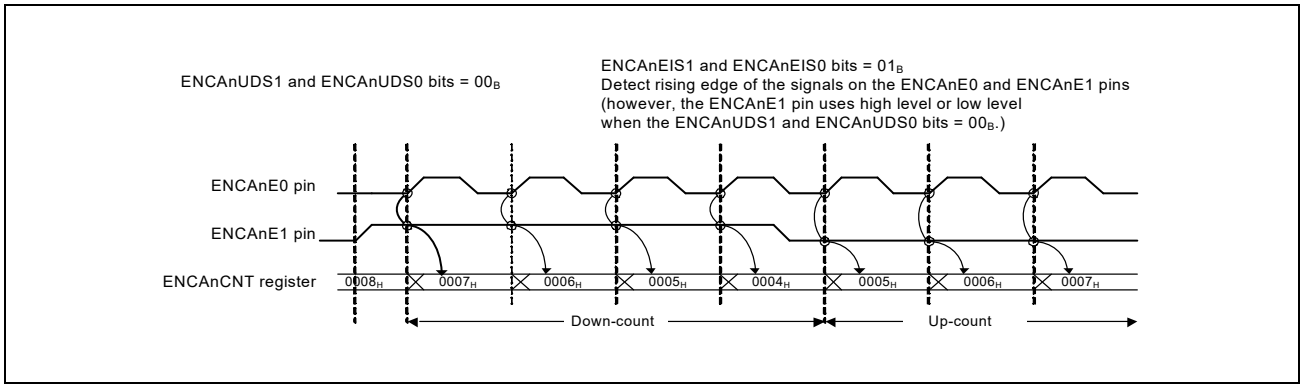


Figure 23.3 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 00_B

23.4.2.2 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 01_B

Table 23.21 When ENCAnUDS1 and ENCAnUDS0 Bits = 01_B

ENCAnUDS1	ENCAnUDS0	Operation Description		
		ENCAnE0 Pin	ENCAnE1 Pin	Counting operation
0	1	Low level	Rising edge	Down
			Falling edge	
			Both edges	
		High level	Rising edge	
			Falling edge	
			Both edges	
		Rising edge	Low level	Up
		Falling edge		
		Both edges		
		Rising edge	High level	
		Falling edge		
		Both edges		
		Simultaneous input		Hold

The effective edges of the signals on the ENCAnE0 and ENCAnE1 pins are specified by setting the ENCAnEIS1 and ENCAnEIS0 bits.

The timer starts counting up/down according to the condition of the edges and levels of the ENCAnE0 and ENCAnE1 pins. When effective edges coincide, the counter keeps counting.

The following timing chart shows the counter operation when the ENCAnUDS1 and ENCAnUDS0 bits = 01_B.

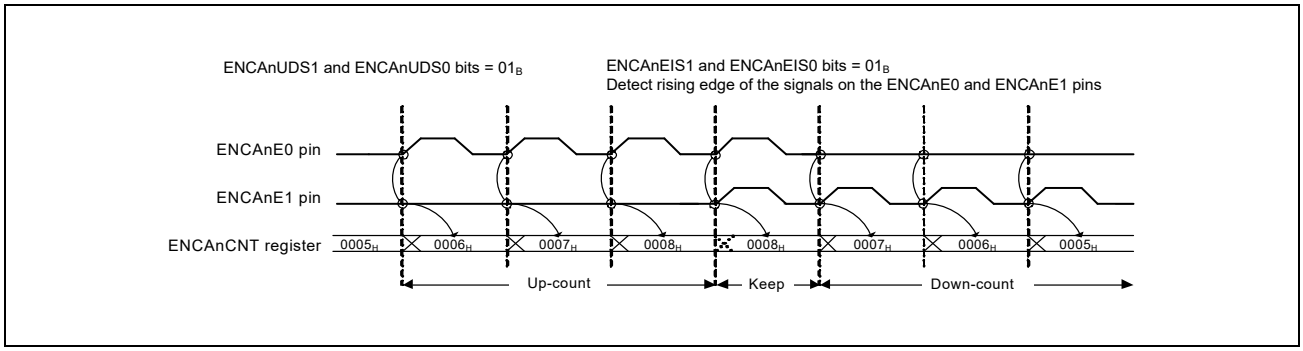


Figure 23.4 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 01_B

23.4.2.3 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 10_B

Table 23.22 When ENCAnUDS1 and ENCAnUDS0 Bits = 10_B

		Operation Description		
ENCAnUDS1	ENCAnUDS0	ENCAnE0 Pin	ENCAnE1 Pin	Counting Operation
1	0	Rising edge	Low level	Down
		Rising edge	Falling edge	
		Falling edge	Low level	Up
		Falling edge	Falling edge	
		Low level	Rising edge	Hold
		Rising edge	Rising edge	
		High level	Rising edge	
		Falling edge	Rising edge	
		Low level	Falling edge	
		Rising edge	High level	
		High level	Falling edge	
		Falling edge	High level	

Specifying effective edges of the signals on the ENCAnE0 and ENCAnE1 pins (by setting the ENCAnEIS1 and ENCAnEIS0 bits) is invalid.

The following timing chart shows the counter operation when the ENCAnUDS1 and ENCAnUDS0 bits = 10_B.

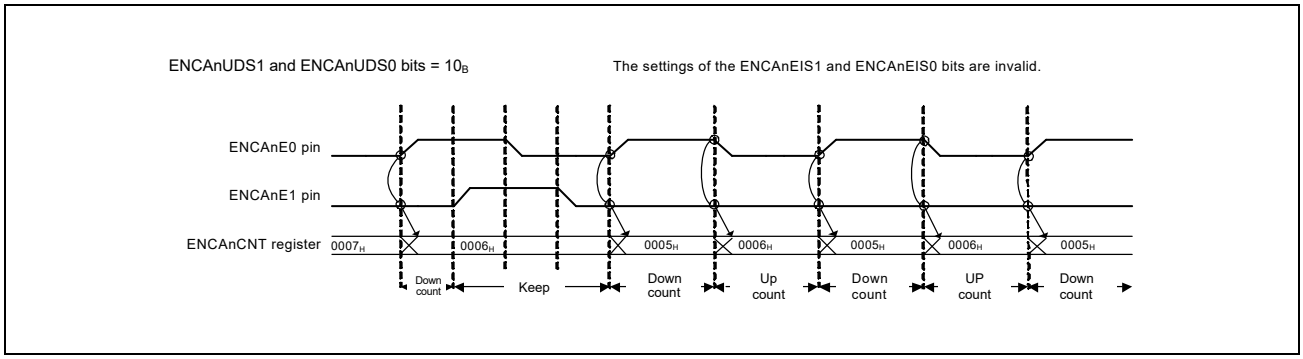


Figure 23.5 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 10_B

23.4.2.4 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 11_B

Table 23.23 When ENCAnUDS1 and ENCAnUDS0 Bits = 11_B

ENCAnUDS1	ENCAnUDS0	Operation Description		
		ENCAnE0 Pin	ENCAnE1 Pin	Counter Operation
1	1	Low level	Falling edge	Down
		Rising edge	Low level	
		High level	Rising edge	
		Falling edge	High level	
		Rising edge	High level	Up
		High level	Falling edge	
		Falling edge	Low level	
		Low level	Rising edge	
		Simultaneous input		Hold

Specifying effective edges of the signals on the ENCAnE0 and ENCAnE1 pins (by setting the ENCAnEIS1 and ENCAnEIS0 bits) is invalid. The counter value is held when the effective edges of the signals on the ENCAnE0 and ENCAnE1 pins coincide.

The following timing chart shows the counter operation when the ENCAnUDS1 and ENCAnUDS0 bits = 11_B.

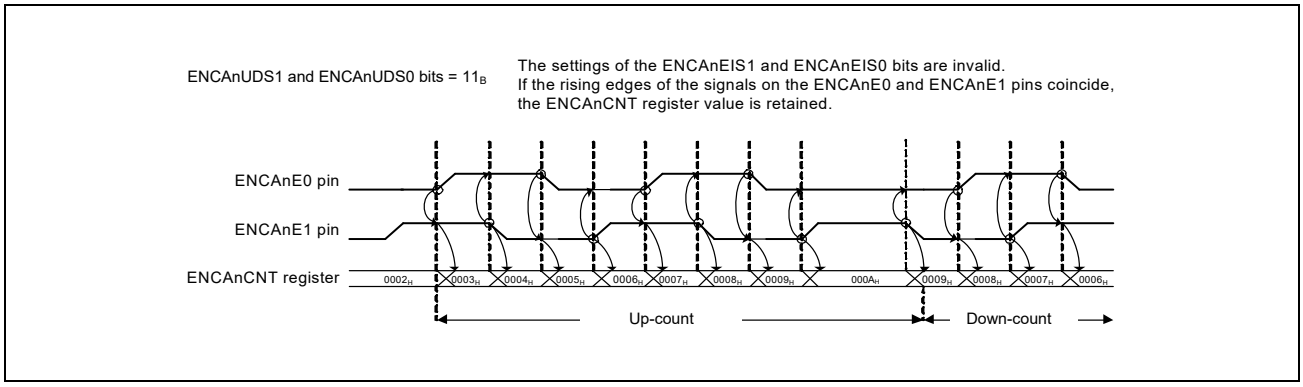


Figure 23.6 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 11_B

23.4.3 Timer Counter Clear Control by Encoder Input

The timer counter is cleared to 0000_H by the encoder clearing input signal (signal on ENCAnEC).
Two types of clearing methods can be selected by controlling the ENCAnSCE, ENCAnZCL, ENCAnBCL, ENCAnACL, ENCAnECS1, and ENCAnECS0 bits of the ENCAnIOC1 register.

Table 23.24 Timer Counter Clear Control by Encoder Input

Clearing Method	ENCAnSCE	ENCAnZCL	ENCAnBCL	ENCAnACL	ENCAnECS1, ENCAnECS0
See **23.4.3.1**	0	Invalid	Invalid	Invalid	Valid
See **23.4.3.2**	1	Valid	Valid	Valid	Invalid

23.4.3.1 Clearing Method when ENCAnSCE = 0

- Upon detection of the effective edge of ENCAnEC, the timer counter is cleared to 0000_H in synchronization with the operation clock.
- The effective edge of ENCAnEC is specified by the setting of the ENCAnECS1 and ENCAnECS0 bits.
- The settings of the ENCAnZCL, ENCAnBCL, and ENCAnACL bits are invalid.
- An encoder clear interrupt request signal (INTENCAnIEC) is output simultaneously with timer counter clearing.

For details about clear operation when ENCAnSCE = 0, see the timing chart in **Section 23.6.24, Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0**.

23.4.3.2 Clearing Method when ENCAnSCE = 1

- When the clear levels of the ENCAnEC, ENCAnE1, ENCAnE0 inputs are detected, the timer counter is cleared to 0000_H in synchronization with the operating clock.
- Specify the clear levels of the ENCAnEC, ENCAnE1, ENCAnE0 inputs by setting the ENCAnZCL, ENCAnBCL, and ENCAnACL bits.
- The settings of the ENCAnECS1 and ENCAnECS0 bits are invalid.
- An encoder clear interrupt request signal (INTENCAnIEC) is output simultaneously with timer counter clearing.
- For details about the clear operation when ENCAnSCE = 1, see the timing charts from **Section23.6.23.2, When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Up-count** to **Section 23.6.23.5, When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Down-count**.

The clearing conditions of the timer counter according to the ENCAnZCL, ENCAnBCL, and ENCAnACL settings are listed in the table below.

Table 23.25 Clearing Conditions of the Timer Counter

Counter Clearing Condition Setting			Encoder Pin Input Level		
ENCAnZCL	ENCAnBCL	ENCAnACL	ENCAnEC	ENCAnE1	ENCAnE0
0	0	0	Low	Low	Low
0	0	1	Low	Low	High
0	1	0	Low	High	Low
0	1	1	Low	High	High
1	0	0	High	Low	Low
1	0	1	High	Low	High
1	1	0	High	High	Low
1	1	1	High	High	High

23.4.4 Functions of ENCAnCCR0

23.4.4.1 Compare Function

- When ENCAnCRM0 = 0, the ENCAnCCR0 register functions as a dedicated compare register.
- Upon compare match between the value of the timer counter and the ENCAnCCR0 setting value, a compare 0 match interrupt (INTENCAnI0) is output.
- When ENCAnECM0 = 1, the timer counter is cleared to 0000_H in synchronization with the operating clock upon compare match if the next counting operation is up-count.

Table 23.26 Compare Function of ENCAnCCR0

ENCAnCCR0 Function	Compare Match Clear Control		Timer Counter Clearing Upon Compare Match with ENCAnCCR0
ENCAnCRM0	ENCAnECM0	Next Counting Operation	
0 (Compare)	0	Up-count	Does not clear (continues counter operation).
		Down-count	
	1	Up-count	Clears timer counter to 0000_H.
		Down-count	Does not clear (continues counter operation).

When ENCAnLDE = 1

- Upon occurrence of an underflow, the setting value of the ENCAnCCR0 register is loaded to the timer counter.
- An underflow interrupt (INTENCAnIUD) is output.

NOTE

For details about the timing chart when ENCAnLDE = 1, see the description from **Section 23.6.13, Using the ENCAnLDE Function Immediately after Startup** to **Section 23.6.17, Up-counting after Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input**.

23.4.4.2 Capture Function

- When ENCAnCRM0 = 1, the ENCAnCCR0 register functions as a dedicated capture register.
- Upon effective edge detection of the capture trigger input 0 (ENCAnI0), the value of the timer counter is stored into ENCAnCCR0.
- A capture 0 interrupt (INTENCAnI0) is output during capture operation.

NOTE

For details about capture operation for ENCAnCCR0, see the timing charts in **Section 23.6.19, Capture Operation between Counter Clocks (ENCAnCCR0)** and **Section 23.6.22, Encoder Operation when Compare Match Clear Control is Disabled**.

23.4.5 Functions of ENCAnCCR1

23.4.5.1 Compare Function

- When ENCAnCRM1 = 0, the ENCAnCCR1 register functions as a dedicated compare register.
- Upon compare match between the value of the timer counter and the ENCAnCCR1 setting value, a compare 1 match interrupt (INTENCAnI1) is output.
- When ENCAnECM1 = 1, the timer counter is cleared to 0000_H in synchronization with the operating clock upon compare match if the next counting operation is down-count.

Table 23.27 Compare Function of ENCAnCCR1

ENCAnCCR1 Function	Compare Match Clear Control		Timer Counter Clearing upon Compare Match with ENCAnCCR1
ENCAnCRM1	ENCAnECM1	Next Counting Operation	
0 (Compare)	0	Up-count	Does not clear (continues counting).
		Down-count	
	1	Up-count	Does not clear (continues counting).
		Down-count	Clears timer count to 0000_H.

Compare match interrupt mask function

- When ENCAnCME = 1, the compare 1 match interrupt mask function is enabled. In this state, the compare 1 match interrupt is output upon the first match of the value of the timer counter and the ENCAnCCR1 setting value, and interrupts are then masked for the second and subsequent compare matches.
- When ENCAnCME = 1 and ENCAnMCS = 0, a compare 1 match interrupt is output once upon the first compare match by writing to the ENCAnCCR1 register (interrupts are masked for the second and subsequent matches until the cancel trigger occurs again).
- When ENCAnCME = 1 and ENCAnMCS = 1, a compare 1 match interrupt is output once upon the first compare match by a timer counter clear operation by the encoder clearing input signal or by a timer counter clear operation upon match between the ENCAnCCR0 register value and the timer counter value (interrupts are masked for the second and subsequent matches until the cancel trigger occurs again).
- When ENCAnCME = 1, ENCAnMCS = 1 and ENCAnLDE = 1, a compare 1 match interrupt is output once upon the first compare match by a loading operation of the ENCAnCCR0 register to the timer counter upon underflow detection (interrupts are masked for the second and subsequent matches until the cancel trigger occurs again).
- Setting ENCAnECM1 to 1 is prohibited when enabling the compare 1 match interrupt mask function.

Table 23.28 Compare Match Interrupt Mask Function

ENCAnCCR1 Function	Compare 1 Match Interrupt Mask	Interrupt Mask Cancel Trigger	Compare 1 Match Interrupt Output upon Compare Match with ENCAnCCR1
ENCAnCRM1	ENCAnCME	ENCAnMCS	
0 (Compare)	0 (Mask function disabled)	— (Setting invalid)	Outputs compare 1 match interrupt upon each compare match.
	1 (Mask function enabled)	0 (Write operation to ENCAnCCR1)	Outputs compare 1 match interrupt once upon the first compare match. (Interrupts are masked for the second and subsequent matches until the cancel trigger occurs.)
		1 (Timer counter clear operation) (Loading from ENCAnCCR0 to timer counter upon underflow when ENCAnLDE = 1)	

23.4.5.2 Capture Function

When ENCAnCRM1 = 1, the ENCAnCCR1 register functions as a dedicated capture register.

NOTE

For details about capture operation to ENCAnCCR1, see the timing chart in **Section 23.6.18, Capture Operation between Counter Clocks (ENCAnCCR1)**.

The operations for each of the ENCAnCTS settings are shown in the table below.

Table 23.29 Capture Function of ENCAnCTS

ENCAnCCR1 Function	Capture Trigger Selection	Capture Trigger Signal	Timer Counter Clearing	Interrupt Occurrence
ENCAnCRM1	ENCAnCTS			
1 (Capture)	0	Capture trigger 1 input (ENCAnI1)	Does not clear timer counter.	(1) Capture 1 interrupt (INTENCAnI1)
	1	Encoder clear input (set with ENCAnSCE)	Clears timer counter.	(1) Capture 1 interrupt (INTENCAnI1) (2) Encoder clear interrupt (INTENCAnIEC)

NOTE

For details about the timing chart when ENCAnCTS = 0 or ENCAnCTS = 1, see the following:

Section 23.6.8, Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin), **Section 23.6.9,** Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin), **Section 23.6.10,** Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin), **Section 23.6.16,** Conflict between ENCAnLDE Function (Loading Counter Value) and Clear Operation by Encoder Clear Input (ENCAnEC pin), and **Section 23.6.17, Up-counting after Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input**.

23.4.5.3 Timer Counter Clearing upon Compare Register Match

Clearing of the timer counter upon compare match between the value of the timer counter and the setting of ENCAnCCR0 or ENCAnCCR1, according to the settings of the ENCAnECM1 and ENCAnECM0 bits in ENCAnCTL, is detailed in the following table.

Table 23.30 Timer Counter Clearing Operation upon Compare Register Match

ENCAnECM1 and ENCAnECM0	Next Counting Operation	Timer Counter Clearing upon Compare Match with ENCAnCCR1	Timer Counter Clearing upon Compare Match with ENCAnCCR0
00	Up-count	Does not clear (continues counting).	Does not clear (continues counting).
	Down-count	Does not clear (continues counting).	Does not clear (continues counting).
01	Up-count	Does not clear (continues counting).	Clears timer counter to 0000_H.
	Down-count	Does not clear (continues counting).	Does not clear (continues counting).
10	Up-count	Does not clear (continues counting).	Does not clear (continues counting).
	Down-count	Clears timer counter to 0000_H.	Does not clear (continues counting).
11	Up-count	Does not clear (continues counting).	Clears timer counter to 0000_H.
	Down-count	Clears timer counter to 0000_H.	Does not clear (continues counting).

23.4.6 Starting and Stopping the Timer Counter

23.4.6.1 Starting the Timers

This product has two encoder timers, for which independent or synchronized operation is selectable.

In the case of independent operation, operation is started by setting the ENCAnTS bits in the respective ENCAnTS registers to 1.

Synchronous operation and simultaneous start with other timers are possible by setting the PIC. For details, see **Section 24.2.3.1, Simultaneous Start Trigger Function**.

23.4.6.2 Stopping the Timers

Setting the ENCAnTT bit in the ENCAnTT register of a given encoder timer to 1 causes the ENCAnTE bit in the ENCAnTE register to be set to 0.

Writing to the ENCAnTT bits in the ENCAnTT registers of the individual encoder timers will lead to each encoder timer stopping with different timing, creating a possible margin of error in the held counter values. Therefore, when resuming operation after it has been stopped, the value in the counter must be re-set or corrected as in the examples below.

Example 1: Remaking the Counter Setting Prior to Restarting Operation

Condition:	Input on the ENCAnEC pin while two encoder timers are operating with common ENCAnE0 and ENCAnE1 pins.
Restart procedure:	Execute simultaneous restarting by setting the encoder timers to the same value. Setting the timers to the same value eliminates errors in the counted value due to operation being stopped and allows restarting of the operation.

Example 2: Correcting the Counter Setting Prior to Restarting Operation

Condition:	Input on the ENCAnEC pin while two encoder timers are operating with separate ENCAnE0 and ENCAnE1 pins.
Restart procedure:	Execute simultaneous restarting by calculating the difference between the values of the counters in the respective encoder timers and compensate for the difference in setting the counters for simultaneous restarting. Since the information of the CPU includes the information on the differences between the values of the counters of the encoder timers, operation can be restarted by correcting the error in the counter value that arose when operation was stopped, i.e. by calculating the difference and setting corrected values.

23.4.6.3 Example of Connection when Two ENCAn Units are Used

To simultaneously operate the counters of the two ENCAn units, make the same settings in the ENCAnUDS[1:0] bits in the ENCAnIOC1 register and the ENCAnCTL register.

When using ENCAnCCR0 as the comparison register for the two ENCAn units, set the same value in the ENCAnCCR0 registers and the ENCAnECM0 and ENCAnLDE bits of the two ENCAn units.

When using ENCAnCCR1 as the comparison register for the two ENCAn units, set the same value in the ENCAnECM1 registers of the two ENCAn units.

Without the settings described above, different counter values will be used and synchronous operation of two ENCAn units is not possible.

An example where two units of ENCAn are installed is shown below.

This setting example assumes the use of the ENCAnCCR0 register as a comparison register, and the ENCAnCCR1 register as a capture register.

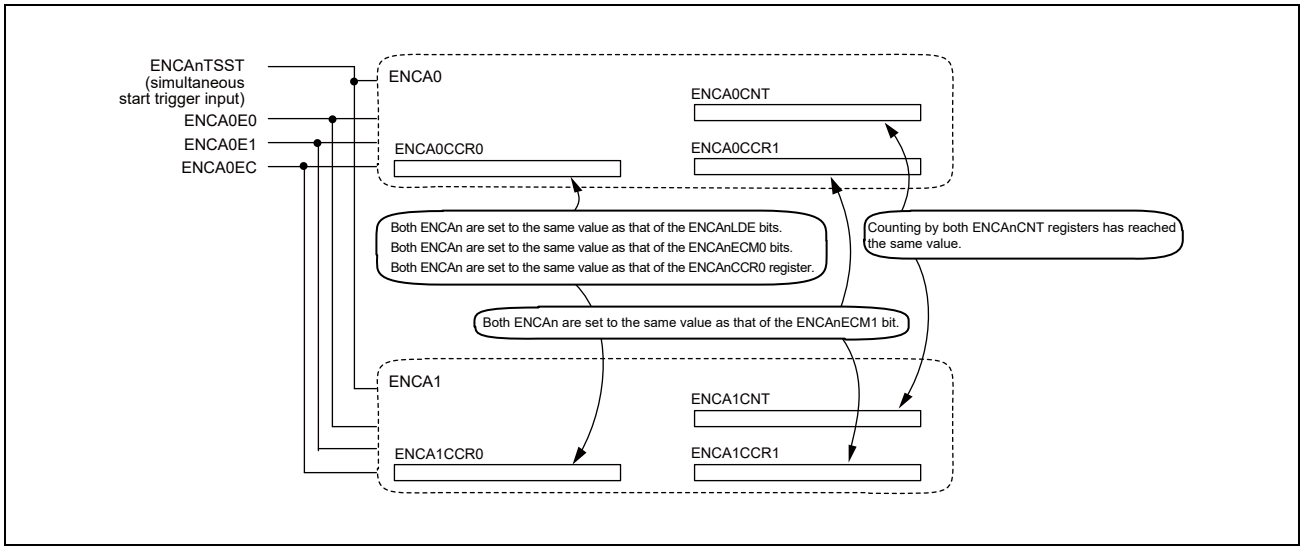


Figure 23.7 Example of Connection when Two ENCAn Units are Used

23.5 Procedure

23.5.1 ENCAn Setting Procedure

The setting procedure for ENCAn is described below.

Table 23.31 ENCAn Setting Procedure

Stage	Operation	State
Initial Setting	Release the module from the reset state	The power for the module is on and the operation is stopped. Writing to registers is enabled.
Initial Settings for ENCAn	Perform the following initial settings. • Settings for the counter • Settings for clearing the counter • Setting for ENCAnCCR0 register • Setting for ENCAnCCR1 register	The counter is not operating in this state. The ENCAnTE bit which indicates the operating state is 0.
	Make initial settings for the counter. • Set any 16-bit value to the ENCAnCNT register. (After setting this register, writing 1 to the ENCAnTS bit starts counting by the counter from the specified value.)	The value set at this stage is used as the initial value of the counter register.
Starting Operation	Make the setting to start the counter. • Set the ENCAnTS bit to 1.	The counter starts in this state. The value of the ENCAnTE bit indicating the operating state is 1, and the counter clock is supplied to the internal circuit.
During operation	Only those registers whose setting can be changed during operation can be rewritten. • Setting of the ENCAnCCR0 register • Setting of the ENCAnCCR1 register • Setting of the ENCAnIOC0 register	The counting operation set in the initial settings is performed. Counting up or down proceeds according to the settings of the ENCAnE0 and ENCAnE1 pins.
Stopping operation	Perform the setting to stop counter operation while the counter is running. • Set the ENCAnTT bit to 1.	The counter is stopped. The value of the ENCAnTE bit indicating the operating state is 0.
Stopping ENCAn	Perform reset	Settings of the registers are initialized.

23.5.1.1 Initial Setting Procedure for the Counter

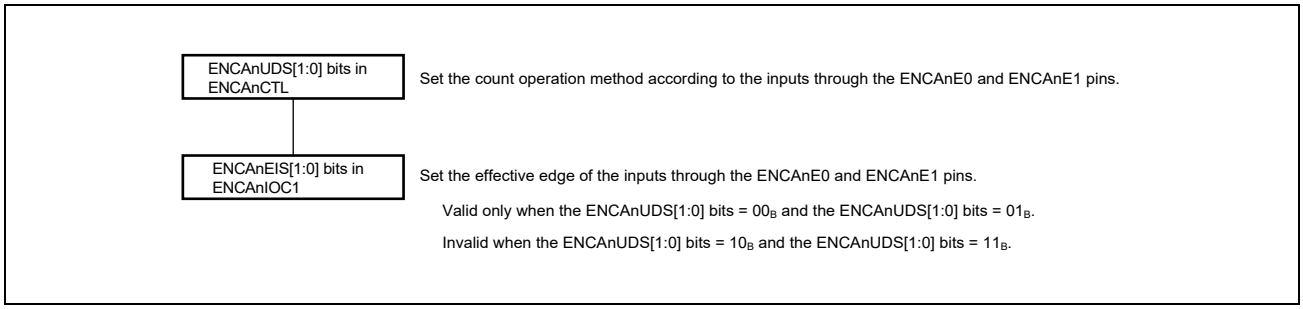


Figure 23.8 Initial Setting Procedure for the Counter

23.5.1.2 Initial Setting Procedure for Clearing the Counter

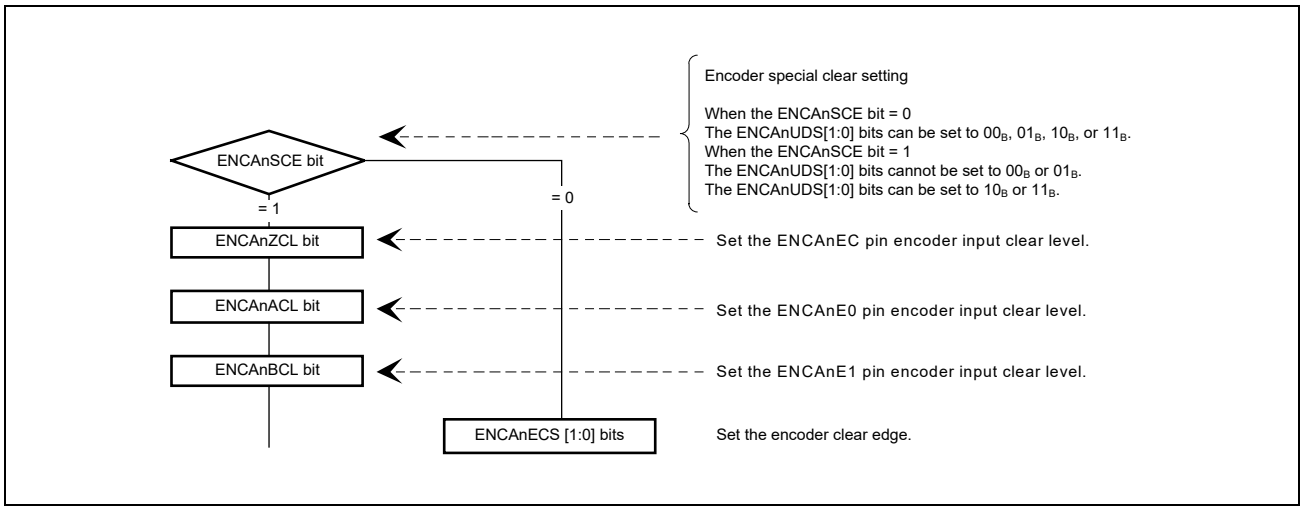


Figure 23.9 Initial Setting Procedure for Clearing the Counter

23.5.1.3 Setting Procedure for the ENCAnCCR0 Register

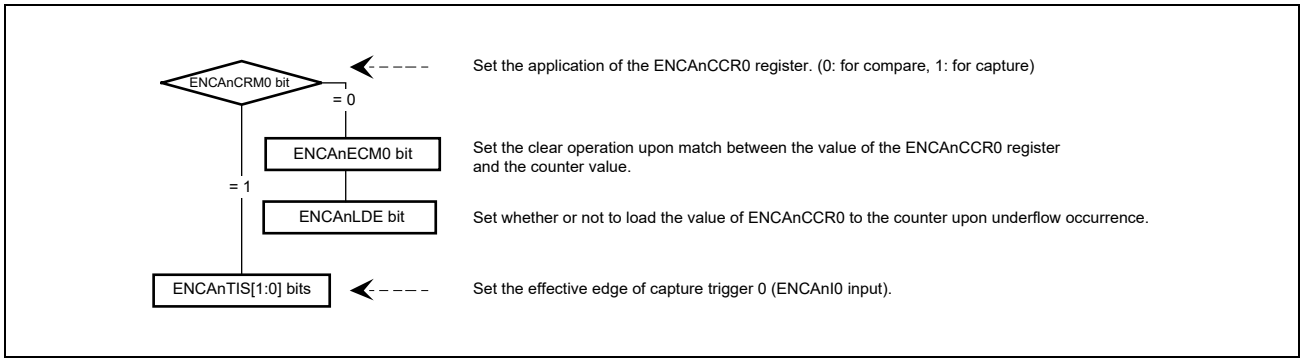


Figure 23.10 Setting Procedure for the ENCAnCCR0 Register

23.5.1.4 Setting Procedure for the ENCAnCCR1 Register

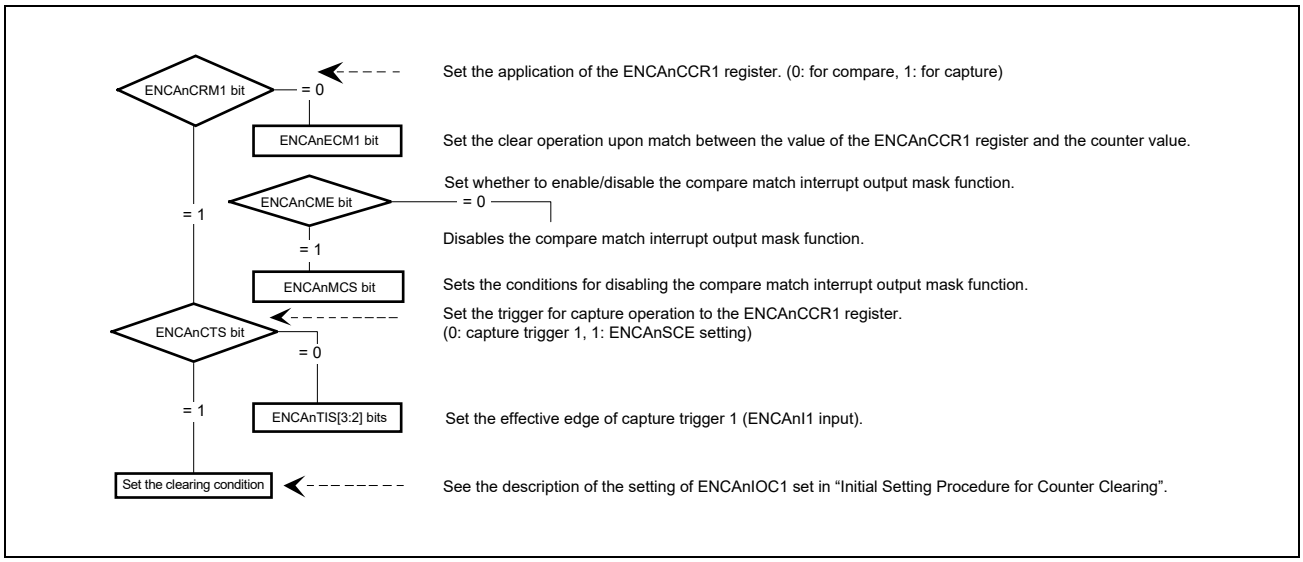


Figure 23.11 Setting Procedure for the ENCAnCCR1 Register

23.6 Timing Charts for Encoder Operations

23.6.1 Timing of Basic Encoder Operation 1 (Encoder Comparison Mode 1)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 00_B:
  Comparison is selected as the function of the ENCAnCCR0 and ENCAnCCR1 registers.
- ENCAnCTL.ENCAnECM[1:0] = 01_B:
  If the next counting after a match between the values in the counter and the ENCAnCCR0 register is up-counting, the counter is cleared.
- ENCAnCTL.ENCAnLDE = 1:
  When the counter underflows, it is loaded with the value from the ENCAnCCR0 register.

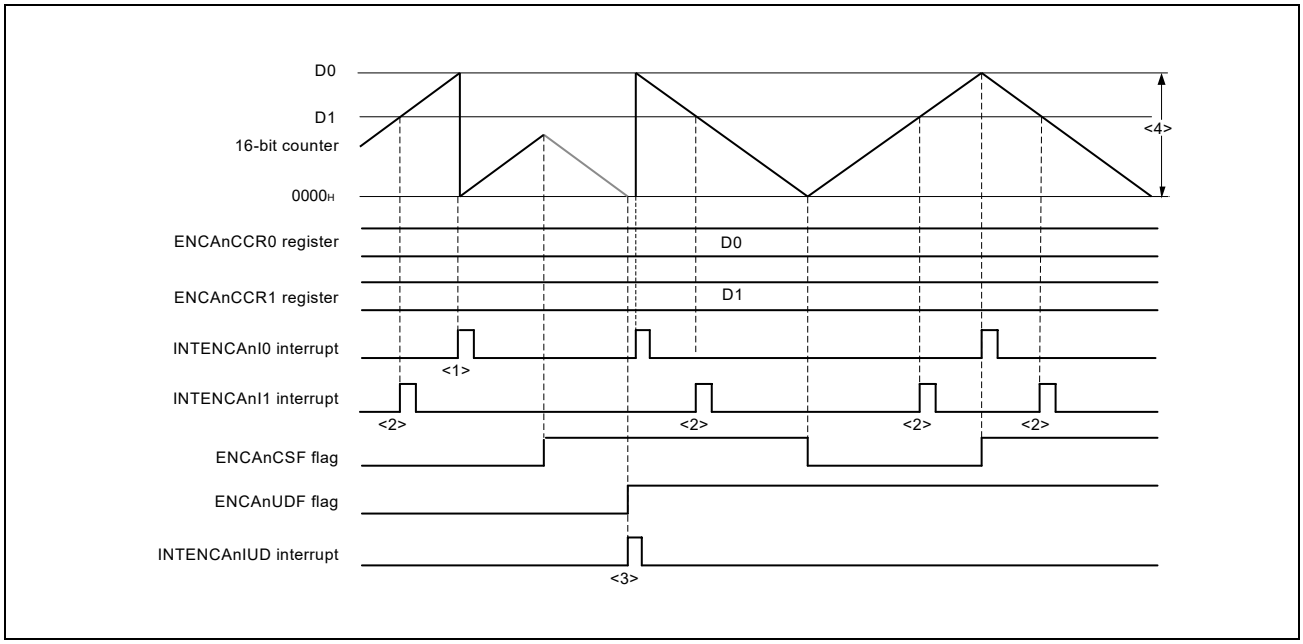


Figure 23.12 Timing of Basic Encoder Operation 1 (Encoder Comparison Mode 1)

1. A compare match interrupt (INTENCAnI0) is generated when the counter value and the ENCAnCCR0 register setting (D0) match.
   If the next counting is up-counting, the counter is cleared to 0000_H because ENCAnECM0 = 1.
2. A compare match interrupt (INTENCAnI1) is generated when the counter value and the ENCAnCCR1 register setting (D1) match.
   Counter clearing due to a match with ENCAnCCR1 does not proceed because ENCAnECM1 = 0.
3. An underflow interrupt (INTENCAnIUD) is generated when the counter underflows. ENCAnLDE = 1, so the counter is loaded with the value from the ENCAnCCR0 register (D0) when the counter underflows.
4. ENCAnLDE = 1 and ENCAnECM[1:0] = 01_B, so counting is from 0000_H to the setting of the ENCAnCCR0 register.

23.6.2 Timing of Basic Encoder Operation 2 (Encoder Comparison Mode 2)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 00_B:
  Comparison is selected as the function of the ENCAnCCR0 and ENCAnCCR1 registers.
- ENCAnCTL.ENCAnECM[1:0] = 00_B:
  The counter is not cleared on a match between its value and that of the ENCAnCCR0 register.
- ENCAnCTL.ENCAnLDE = 0:
  The counter is not loaded with the value from the ENCAnCCR0 register.

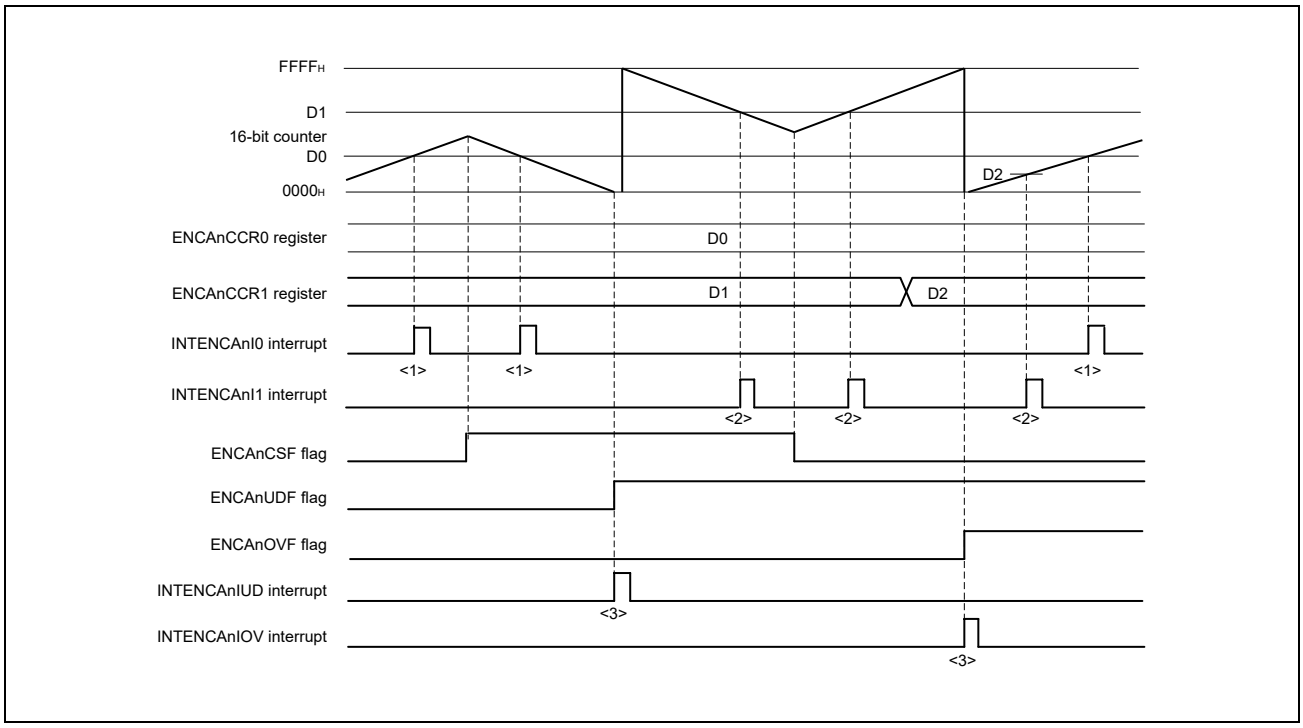


Figure 23.13 Timing of Basic Encoder Operation 2 (Encoder Comparison Mode 2)

1. A compare match interrupt (INTENCAnI0) is generated when the counter value and the ENCAnCCR0 register setting (D0) match.
   Counter clearing due to matching with ENCAnCCR0 does not proceed because ENCAnECM0 = 0.
2. A compare match interrupt (INTENCAnI1) is generated when the counter value and the ENCAnCCR1 register setting (D1, D2) match.
   Counter clearing due to matching with ENCAnCCR1 does not proceed because ENCAnECM1 = 0.
3. Overflow interrupts (INTENCAnIOV) or underflow interrupts (INTENCAnIUD) are generated in response to an overflow or underflow of the counter.

23.6.3 Timing of Basic Encoder Operation 3 (Encoder Comparison Mode 3)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 00_B:
  Comparison is selected as the function of the ENCAnCCR0 and ENCAnCCR1 registers.
- ENCAnCTL.ENCAnECM[1:0] = 11_B:
  If the next counting after a match between the values in the counter and the ENCAnCCR0 register is up-counting, the counter is cleared.
  If the next counting after a match between the values in the counter and the ENCAnCCR1 register is down-counting, the counter is cleared.
- ENCAnCTL.ENCAnLDE = 0
  The counter is not loaded with the value from the ENCAnCCR0 register.

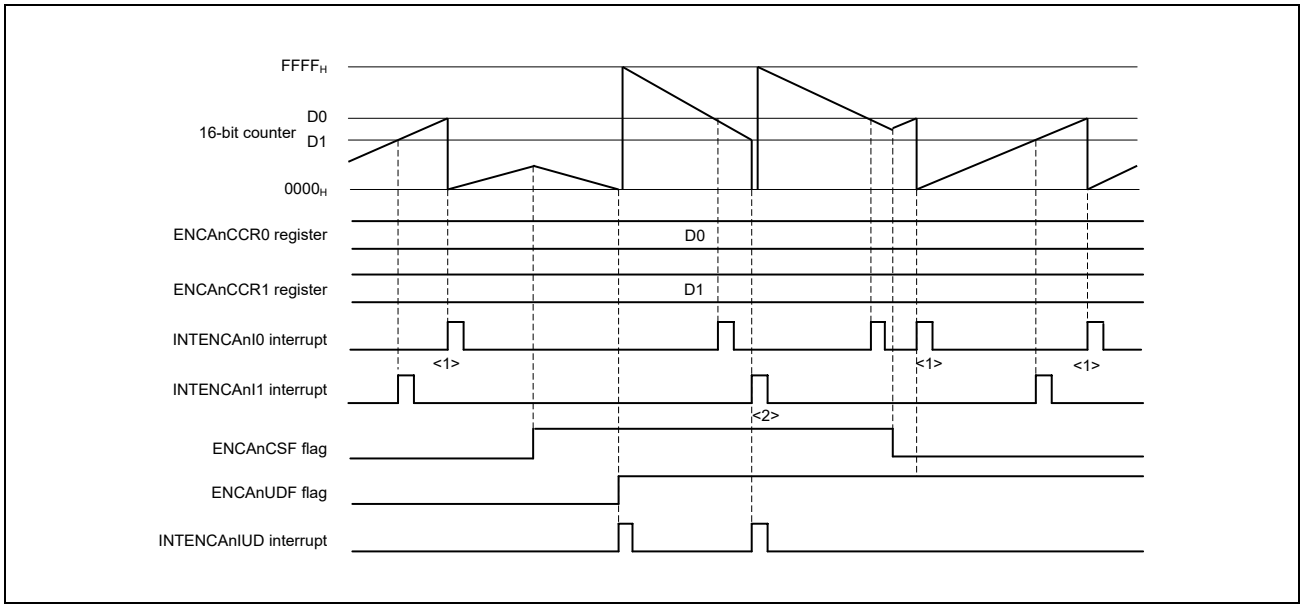


Figure 23.14 Timing of Basic Encoder Operation 3 (Encoder Comparison Mode 3)

1. A compare match interrupt (INTENCAnI0) is generated when the counter value and the ENCAnCCR0 register setting (D0) match.
   If the next counting is up-counting, the counter is cleared to 0000_H because ENCAnECM0 = 1.
2. A compare match interrupt (INTENCAnI1) is generated when the counter value and the ENCAnCCR1 register setting (D1) match.
   If the next counting is down-counting, the counter is cleared to 0000_H because ENCAnECM1 = 1.

23.6.4 Timing of Basic Encoder Operation 4 (Encoder Capture Mode)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 11_B:
  Capture is selected as the function of the ENCAnCCR0 and ENCAnCCR1 registers.
- ENCAnCTL.ENCAnECM[1:0] = 00_B:
  The counter is not cleared on a match between its value and that of the ENCAnCCR0 register.
- ENCAnCTL.ENCAnLDE = 0:
  The setting from the ENCAnCCR0 register is not loaded to the counter.
- ENCAnIOC1.ENCAnSCE = 0, ENCAnECS[1:0] = 00_B:
  Input on the ENCAnEC pin does not lead to edge detection.
- ENCAnIOC0.ENCAnTIS[3:2] = 01_B:
  Selects detection of rising edges of the signal on the ENCAnI1 pin.
- ENCAnIOC0.ENCAnTIS[1:0] = 01_B:
  Selects detection of rising edges of the signal on the ENCAnI0 pin.

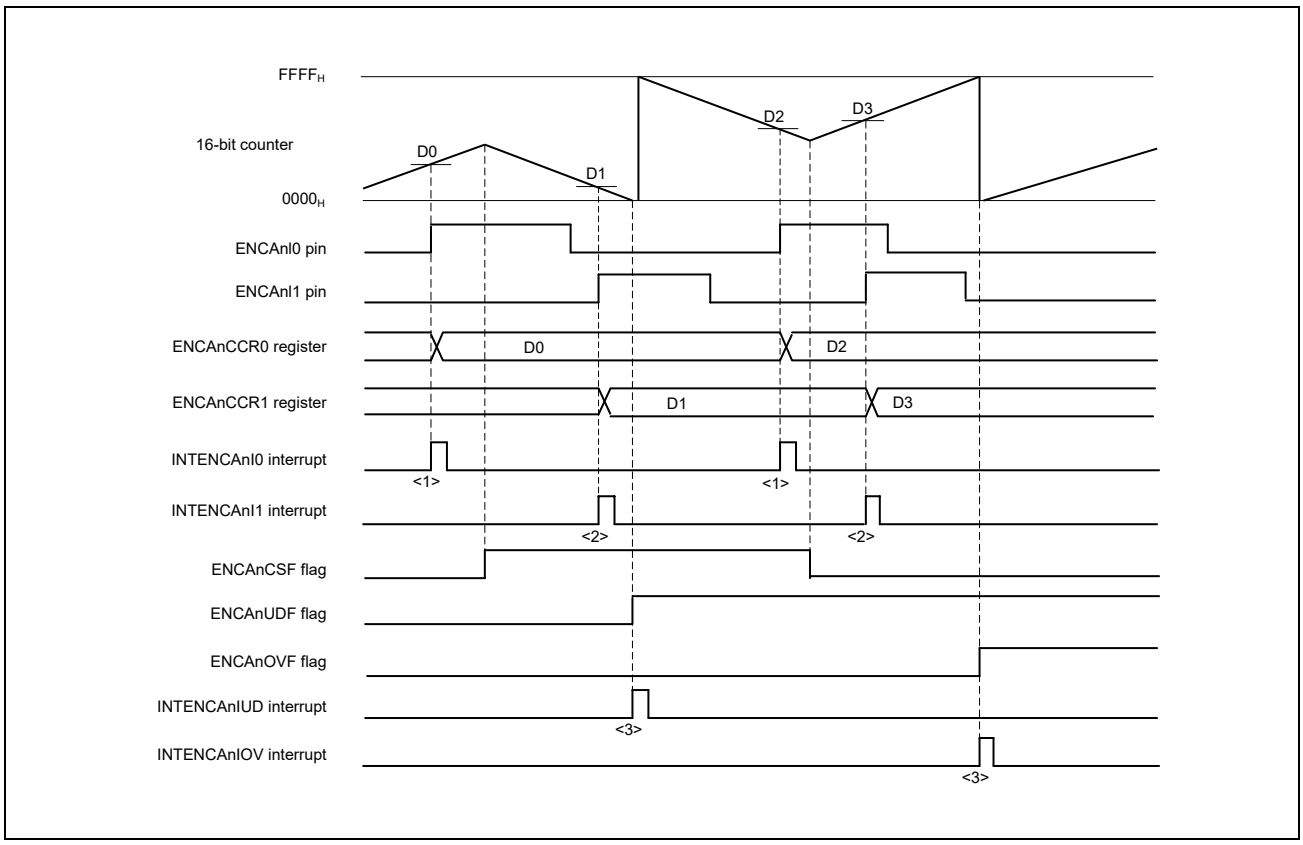


Figure 23.15 Timing of Basic Encoder Operation 4 (Encoder Capture Mode)

1. Detection of a rising edge on the ENCAnI0 pin leads to storage of the counter value (D0, D2) in the capture register (ENCAnCCR0) and the generation of a capture interrupt (INTENCAnI0).
2. Detection of a rising edge on the ENCAnI1 pin leads to storage of the counter value (D1, D3) in the capture register (ENCAnCCR1) and the generation of a capture interrupt (INTENCAnI1).
3. Overflow interrupts (INTENCAnIOV) or underflow interrupts (INTENCAnIUD) are generated in response to an overflow or underflow of the counter.

23.6.5 Timing of Basic Encoder Operation 5 (Encoder Capture and Comparison Mode)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 10_B:
  Select the comparison function for the ENCAnCCR0 register and the capture function for the ENCAnCCR1 register.
- ENCAnCTL.ENCAnECM[1:0] = 01_B:
  The counter is cleared when its value matches that of the ENCAnCCR0 register.
- ENCAnCTL.ENCAnLDE = 1:
  When the counter underflows, it is loaded with the value from the ENCAnCCR0 register.
- ENCAnIOC1.ENCAnSCE = 0, ENCAnECS[1:0] = 00_B
- ENCAnIOC0.ENCAnTIS[3:2] = 11_B
  Selects detection of both edges of the signal on the ENCAnI1 pin.

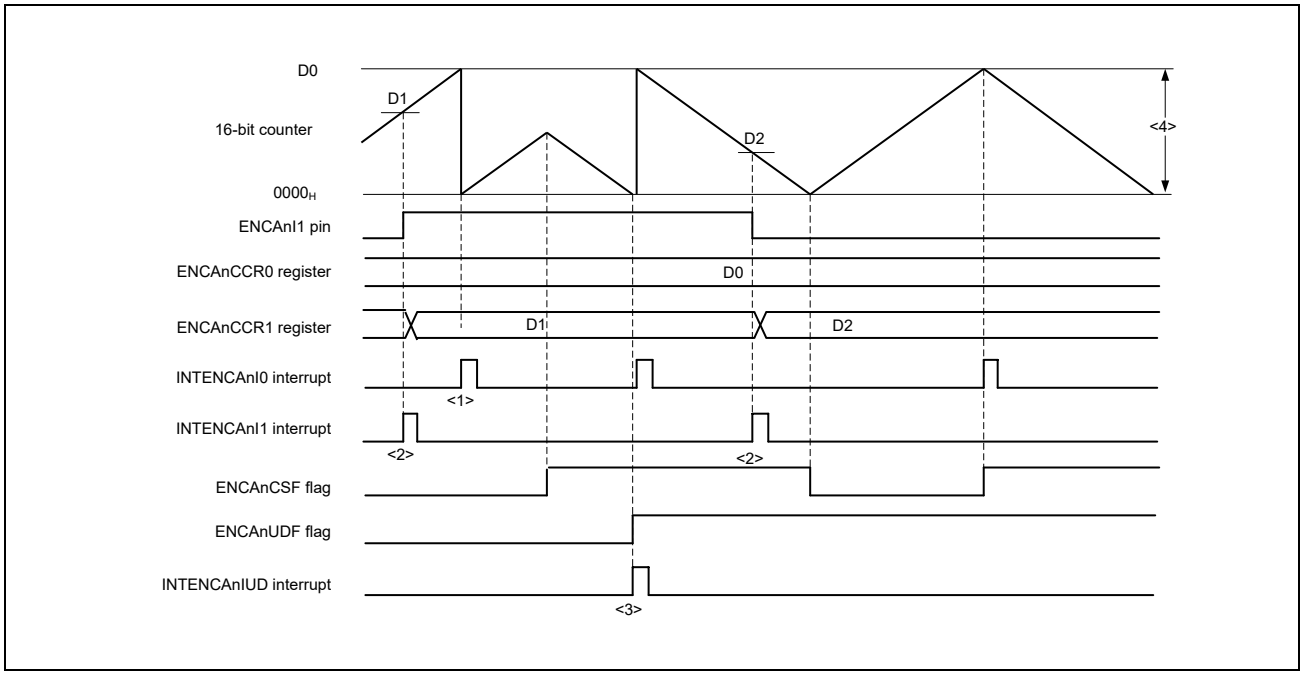


Figure 23.16 Timing of Basic Encoder Operation 5 (Encoder Capture and Comparison Mode)

1. A compare match interrupt (INTENCAnI0) is generated when the counter value and the ENCAnCCR0 register setting (D0) match.
   If the next counting is up-counting, the counter is cleared to 0000_H because ENCAnECM0 = 1.
2. Detection of both edges on the ENCAnI1 pin leads to storage of the counter value (D1) in the capture register (ENCAnCCR1) and the generation of a capture interrupt (INTENCAnI1).
3. An underflow interrupt (INTENCAnIUD) is generated when the counter underflows. ENCAnLDE = 1, so the counter is loaded with the value from the ENCAnCCR0 register (D0) when the counter underflows.
4. ENCAnLDE = 1 and ENCAnECM[1:0] = 01_B, so counting is from 0000_H to the setting of the ENCAnCCR0 register.

23.6.6 Overflow Occurrence and Overflow Flag Clear Operation

When up-counting is performed while the counter value is $FFFF_H$, an overflow occurs. If an overflow occurs, an overflow interrupt (INTENCAnIOV) is output and the overflow flag (ENCAnOVF) is set to 1. When the overflow flag clear bit (ENCAnCLOV) is set to 1, the overflow flag (ENCAnOVF) is cleared to 0.

The overflow occurrence and overflow flag clear operation are described as follows.

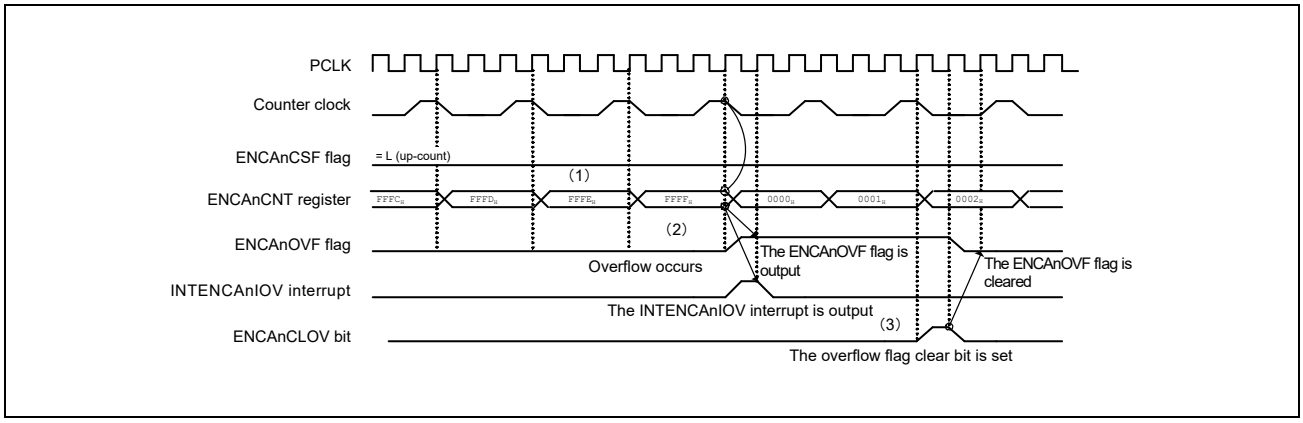


Figure 23.17 Overflow Occurrence and Setting for Clearing of the Overflow Flag

1. The counter value is counted up from $FFFE_H$ to $FFFF_H$.
2. When the counter value changes from $FFFF_H$ to 0000_H, an overflow occurs. At the same time, an overflow interrupt is output, and the overflow flag is set to 1.
3. An overflow flag is cleared to 0 by setting 1 to the ENCAnCLOV bit in the ENCAnFGC register according to the clearing procedure. In addition, an overflow flag is also cleared by setting the ENCAnTS bit in the ENCAnTS register to 1 while ENCAnTE.ENCAnTE is 0, or by setting an input signal of the ENCAnTSST (simultaneous start trigger input) to high.

23.6.7 Underflow Occurrence and Underflow Flag Clear Operation

When down-counting is performed while the counter value is 0000_H, an underflow occurs. If an underflow occurs, an underflow interrupt (INTENCAnIUD) is output and the underflow flag (ENCAnUDF) is set to 1. When the underflow flag clear bit (ENCAnCLUD) is set to 1, the underflow flag (ENCAnUDF) is cleared to 0.

The underflow occurrence and underflow flag clear operation are described as follows.

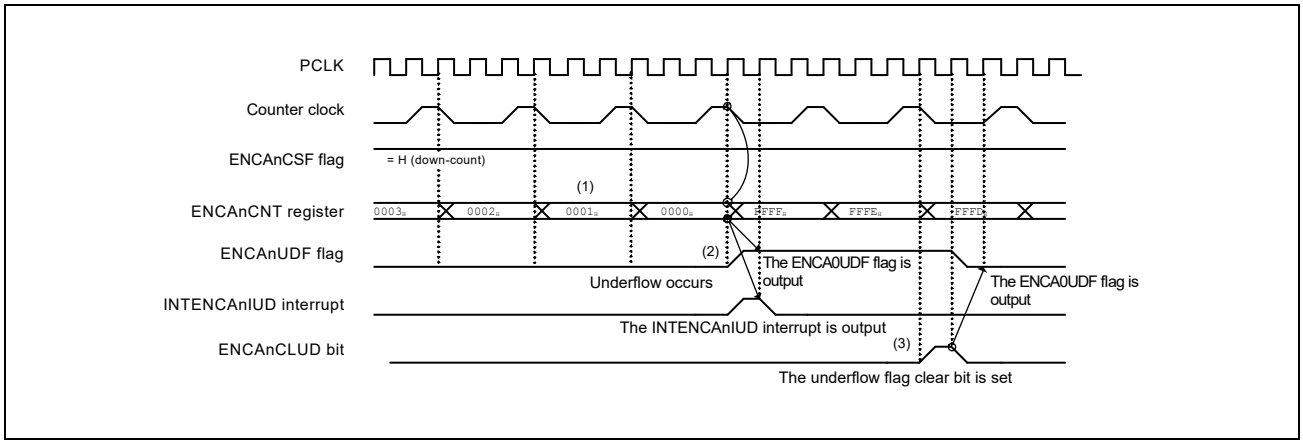


Figure 23.18 Underflow Occurrence and Setting for Clearing of the Underflow Flag

1. The counter value is decremented from 0001_H to 0000_H.
2. When the counter value changes from 0000_H to $FFFF_H$, an underflow occurs. At the same time, an underflow interrupt is output, and the underflow flag is set to 1.
3. An underflow flag is cleared to 0 by setting 1 to the ENCAnCLUD bit in the ENCAnFGC register according to the clearing procedure. In addition, an underflow flag is also cleared by setting the ENCAnTS bit in the ENCAnTS register to 1 while the ENCAnTE bit in the ENCAnTE register is 0, or by setting an input signal of the ENCAnTSST (simultaneous start trigger input) to high.

23.6.8 Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin)

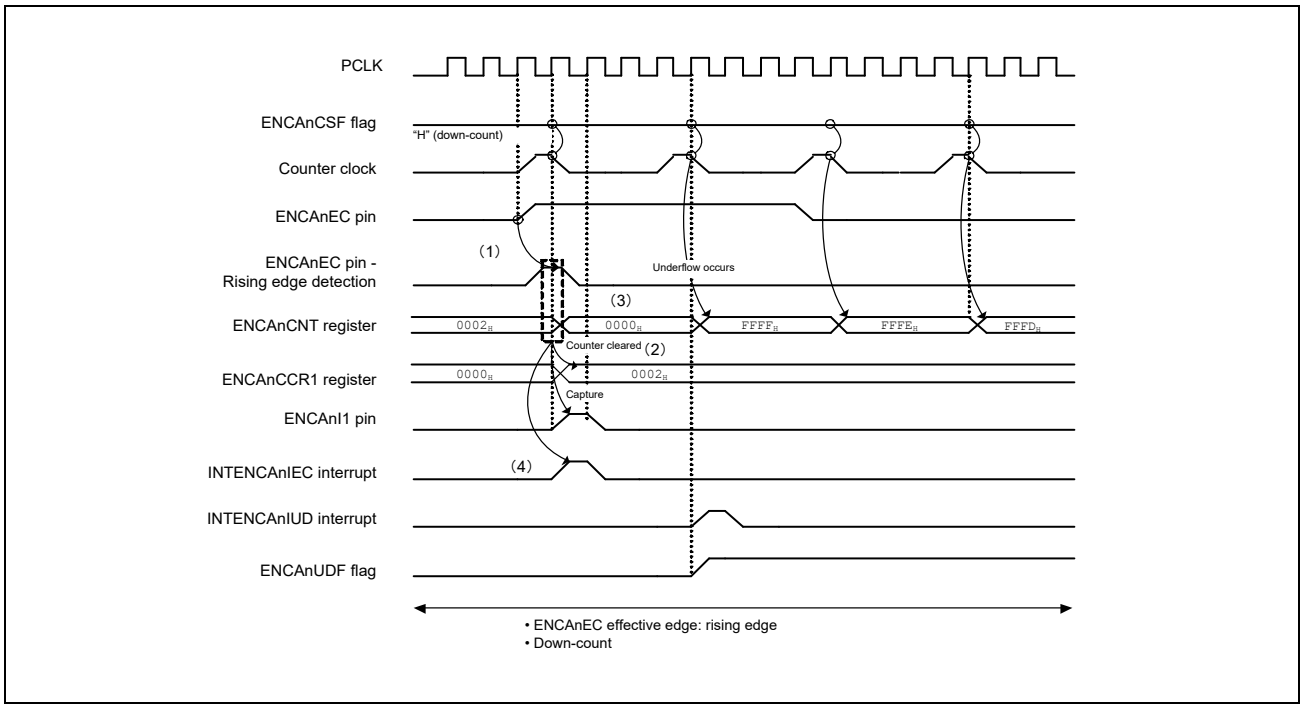


Figure 23.19 Timing Chart of Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin)

<Setting Conditions>

- ENCAnCTL.ENCAnCRM1 = 1
  (ENCAnCCR1 is selected as the capture register)
- ENCAnCTL.ENCAnCTS = 1
  (ENCAnEC pin input is selected as a capture trigger input)
- ENCAnIOC1.ENCAnECS1 and ENCAnECS0 bits = 01_B
  (A rising edge is selected as the effective edge to be detected for the ENCAnEC pin input)

1. Capture operation is performed at the rising edge of the ENCAnEC pin input trigger.
2. The counter value (0002_H) is captured at the rising edge of the ENCAnEC pin input, and it is stored in the ENCAnCCR1 register.
3. Clear operation is performed by the input through the ENCAnEC pin, and the counter value is reset to 0000_H.
4. At the same time, by the input through the ENCAnEC pin, an encoder clear interrupt (INTENCAnIEC) and capture interrupt 1 (INTENCAnI1) are output.

23.6.9 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)

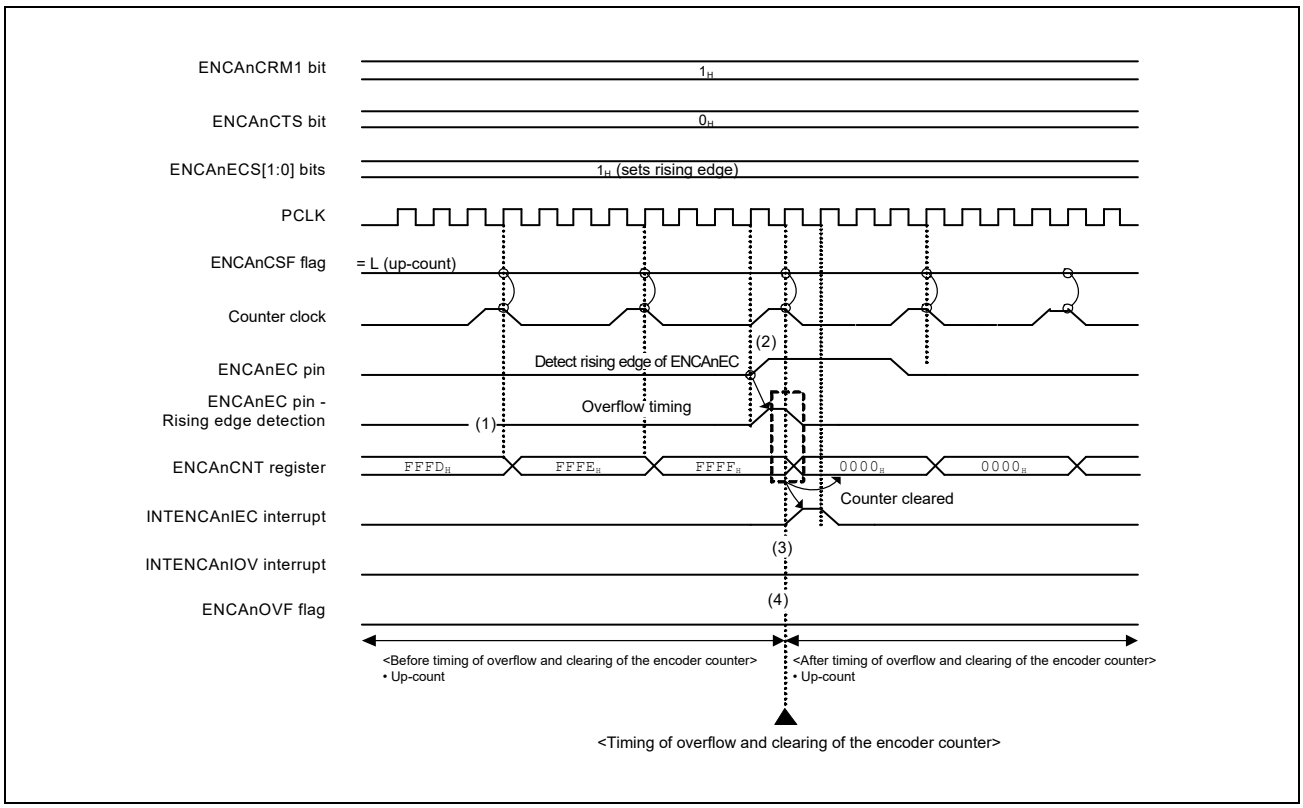


Figure 23.20 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)

1. Up-counting from $FFFD_H$ is continuously performed.
2. If an overflow occurs due to the counter value reaching $FFFF_H$, and the rising edge of ENCAnEC is detected at the same time, clearing by the encoder clear input is performed. Then, the counter value is cleared to 0000_H.
3. The encoder clear input clears the counter value and outputs a clear interrupt (INTENCAnIEC) at the same time. Here, an overflow is deemed to not have occurred and an overflow interrupt is not output. The counter has been cleared.
4. An overflow has not occurred as described in step 3, the overflow flag is not set.

23.6.10 Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)

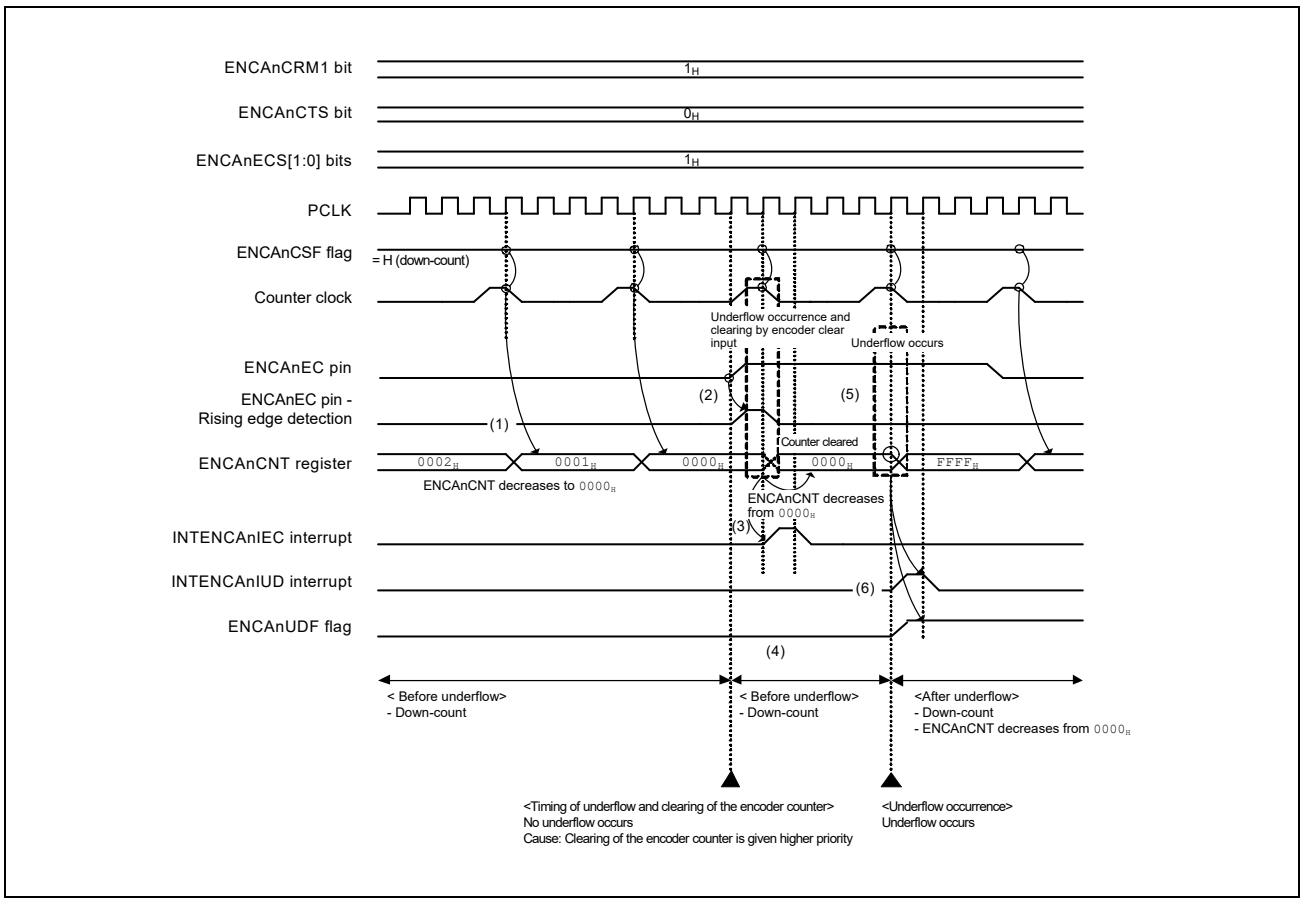


Figure 23.21 Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)

1. Down-counting from 0002_H is continuously performed.
2. If an underflow occurs due to the counter value reaching 0000_H, and the rising edge of ENCAnEC is detected at the same time, clearing by the encoder clear input is performed. Even if the next clock signal is input during clear operation, the counter value remains at 0000_H.
3. When the counter value is cleared by the encoder clear input, an encoder clear interrupt (INTENCAnIEC) is output simultaneously. Here, an underflow is deemed to not have occurred and an underflow interrupt is not output. The counter has been cleared.
4. An underflow has not occurred as described in step 3, the underflow flag is not set.
5. When down-counting is further performed after the counter value has reached 0000_H by clear operation by the encoder clear input, the counter value changes from 0000_H to $FFFF_H$, and another underflow occurs.
6. When an underflow occurs, an underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.

23.6.11 Overflow Operation Immediately after Startup

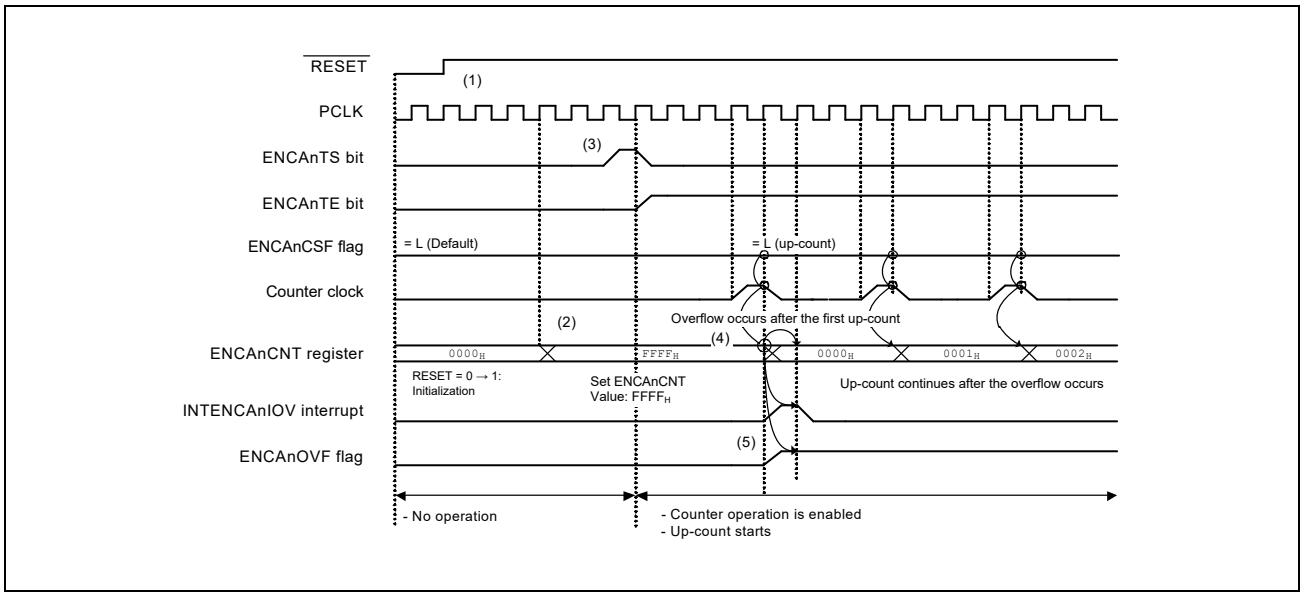


Figure 23.22 Overflow Operation Immediately after Startup

1. When the $\overline{\text{RESET}}$ value changes from 0 to 1, the state of the counter changes from “reset” to “reset release”.
2. The timer counter is set to $FFFF_H$ as the initial value.
3. ENCAnTS is set to 1, and operation starts. ENCAnTE changes to 1, which indicates that operation is enabled.
4. When up-counting is performed from $FFFF_H$ which is the initially set counter value, the counter value changes from $FFFF_H$ to 0000_H, and an overflow occurs immediately after operation starts.
5. At the same time, by an overflow occurrence immediately after operation starts, an overflow interrupt (INTENCAnIOV) is output, and the overflow flag (ENCAnOVF) is set.

23.6.12 Underflow Operation Immediately after Startup

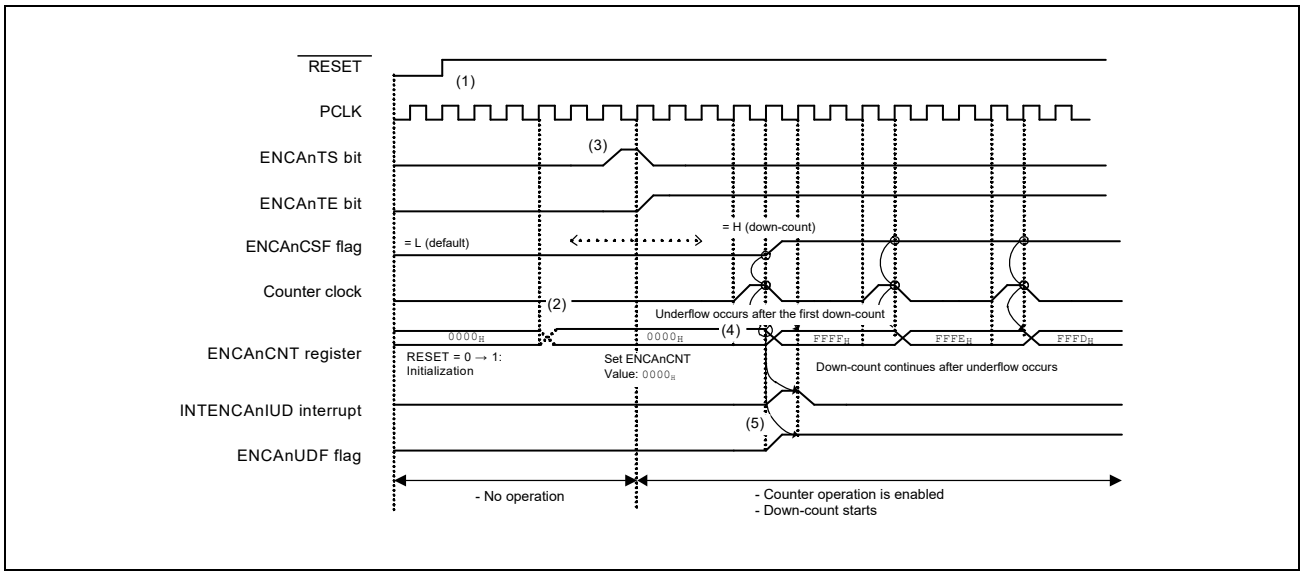


Figure 23.23 Underflow Operation Immediately after Startup

1. When the $\overline{\text{RESET}}$ value changes from 0 to 1, the state of the counter changes from “reset” to “reset release”.
2. The timer counter is set to 0000_H as the initial value.
3. ENCAnTS is set to 1, and operation starts. ENCAnTE changes to 1, which indicates that operation is enabled.
4. When down-counting is performed from 0000_H which is the initially set counter value, the counter value changes from 0000_H to $FFFF_H$, and an underflow occurs immediately after operation starts.
5. At the same time, by an underflow occurrence immediately after operation starts, an underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.

23.6.13 Using the ENCAnLDE Function Immediately after Startup

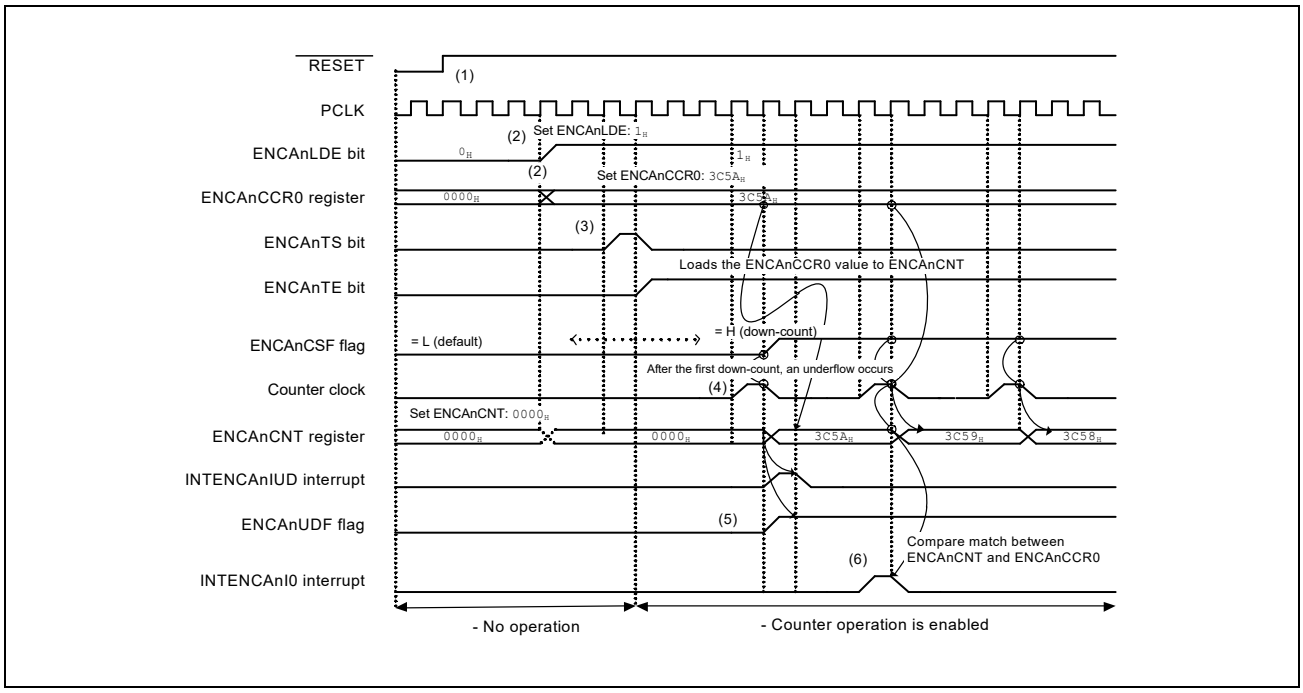


Figure 23.24 Using the ENCAnLDE Function Immediately after Startup

1. When the $\overline{\text{RESET}}$ value changes from 0 to 1, the status is changes from “reset” to “reset release”.
2. The load enable bit (ENCAnLDE) is set to 1, capture compare register 0 (ENCAnCCR0) is set to $3C5A_H$, and the timer counter is set to the initial value 0000_H.
3. ENCAnTS is set to 1, and operation starts. ENCAnTE changes to 1, which indicates that operation is enabled.
4. When down-counting is performed from 0000_H which is the initially set counter value, an underflow occurs immediately after operation starts. Because ENCAnLDE is set to 1, the ENCAnCCR0 value, $3C5A_H$, is loaded to the timer counter (INTENCAnI0 is not output during loading).
5. At the same time, by an underflow occurrence immediately after operation starts, an underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set (after an underflow occurs, down-counting from the loaded value ($3C5A_H$) continues).
6. After the ENCAnCCR0 value is loaded to ENCAnCNT, a match with ENCAnCCR0 is detected, and INTENCAnI0 is output.

23.6.14 ENCAnLDE Function (Loading Counter Value)

(1) <When ENCAnLDE = 0>

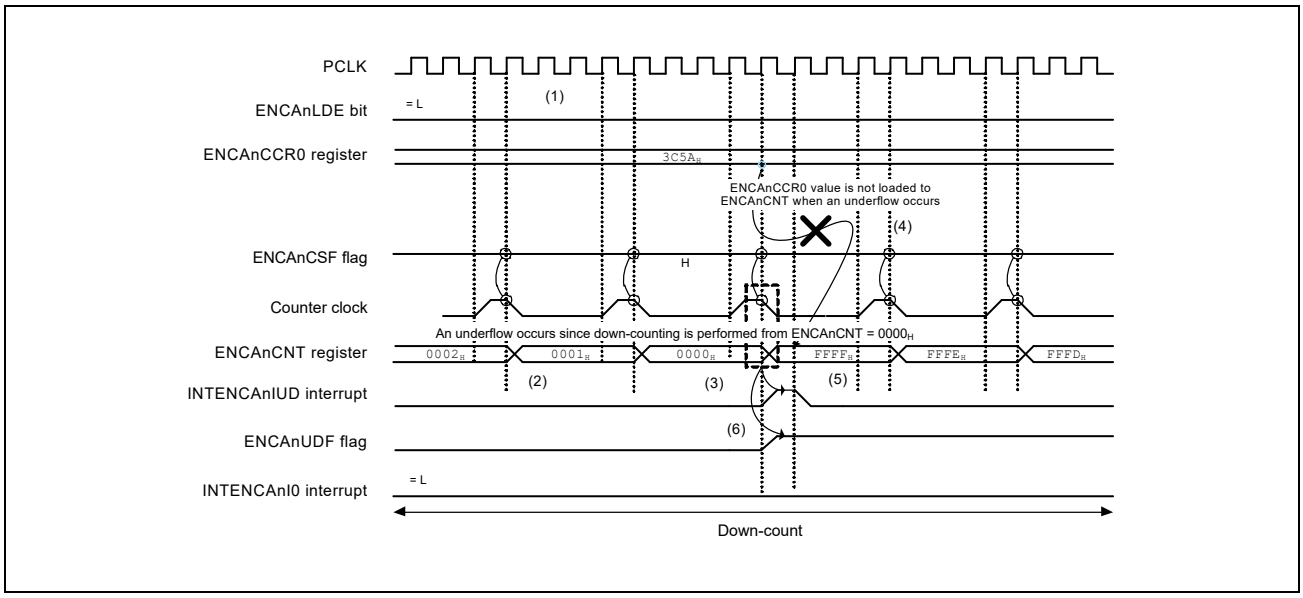


Figure 23.25 ENCAnLDE Function (when ENCAnLDE = 0)

1. ENCAnLDE is set to 0 (even if an underflow occurs, the ENCAnCCR0 value is not loaded).
2. Down-counting is performed: $0002_H \rightarrow 0001_H \rightarrow 0000_H$
3. When down-counting is further performed after the counter value changes to 0000_H, an underflow occurs.
4. Because ENCAnLDE is set to 0, the setting value of the ENCAnCCR0 register is not loaded to the counter even if an underflow occurs.
5. Operation changes to underflow operation (counter value: $0000_H \rightarrow FFFF_H$).
6. An underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.

(2) <When ENCAnLDE = 1>

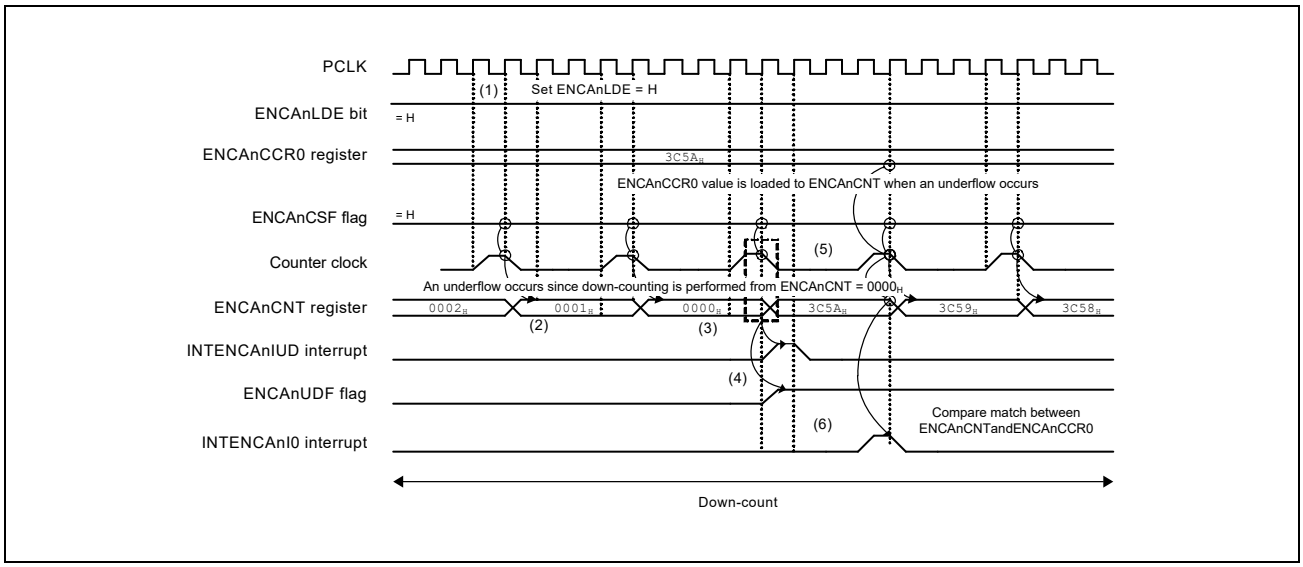


Figure 23.26 ENCAnLDE Function (when ENCAnLDE = 1)

1. ENCAnLDE is set to 1 (if an underflow occurs, the ENCAnCCR0 value is loaded to the counter).
2. Down-counting is performed: $0002_H \rightarrow 0001_H \rightarrow 0000_H$
3. When down-counting is further performed after the counter value changes to 0000_H, an underflow occurs.
4. An underflow interrupt is output, and the underflow flag is set.
5. Because ENCAnLDE is set to 1, the setting value of the ENCAnCCR0 register is loaded to the counter if an underflow occurs. ENCAnCNT is set to $3C5A_H$.
6. After the ENCAnCCR0 value is set to ENCAnCNT, if the ENCAnCNT value matches with the ENCAnCCR0 value on a counter clock, a compare match interrupt (INTENCAnI0) is output.

23.6.15 Conflict between ENCAnLDE Function (Loading Counter Value) and Rewriting of ENCAnCCR0 Register

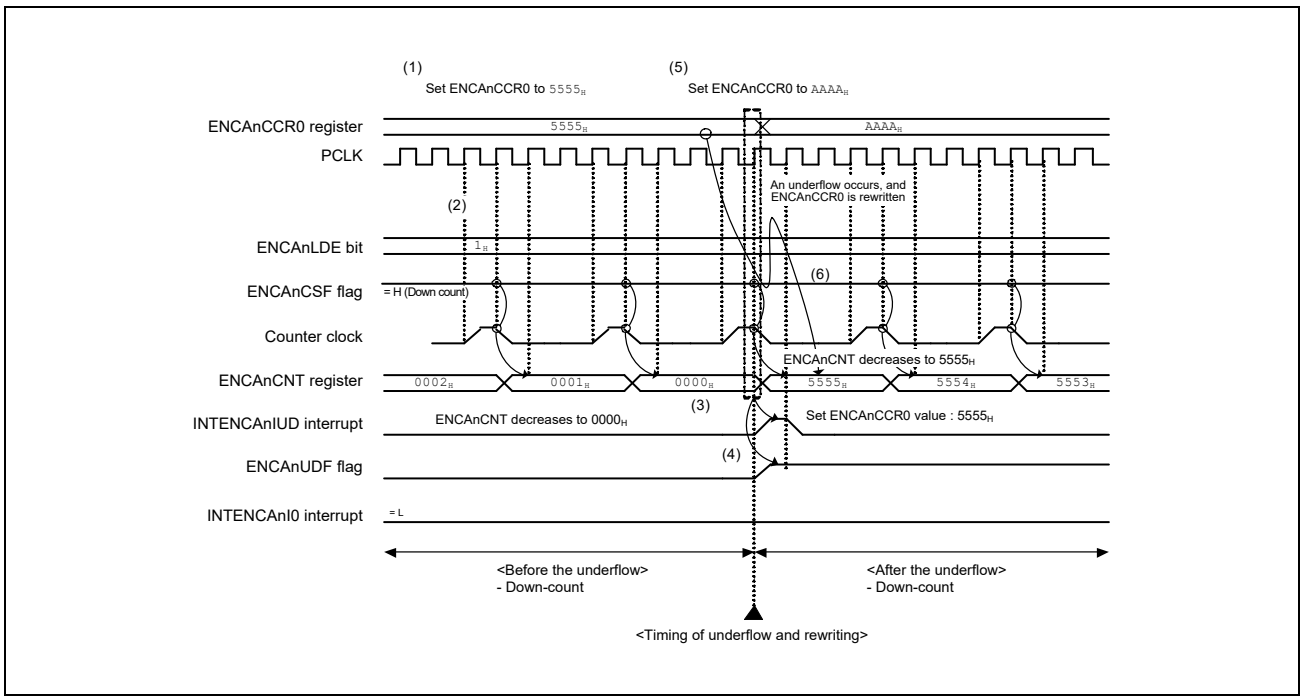


Figure 23.27 Conflict between ENCAnLDE Function and Rewriting of ENCAnCCR0 Register

1. The ENCAnCCR0 register is currently set to 5555_H.
2. ENCAnLDE is currently set to 1.
3. Down-counting is performed ($0002_H \rightarrow 0001_H \rightarrow 0000_H$), and an underflow occurs.
4. An underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.
5. When an underflow occurs, the ENCAnCCR0 register value is changed from 5555_H to $AAAA_H$.
6. Additionally, when an underflow occurs, the ENCAnCCR0 value before the rewrite was performed (5555_H) is set in ENCAnCNT.

23.6.16 Conflict between ENCAnLDE Function (Loading Counter Value) and Clear Operation by Encoder Clear Input (ENCAnEC pin)

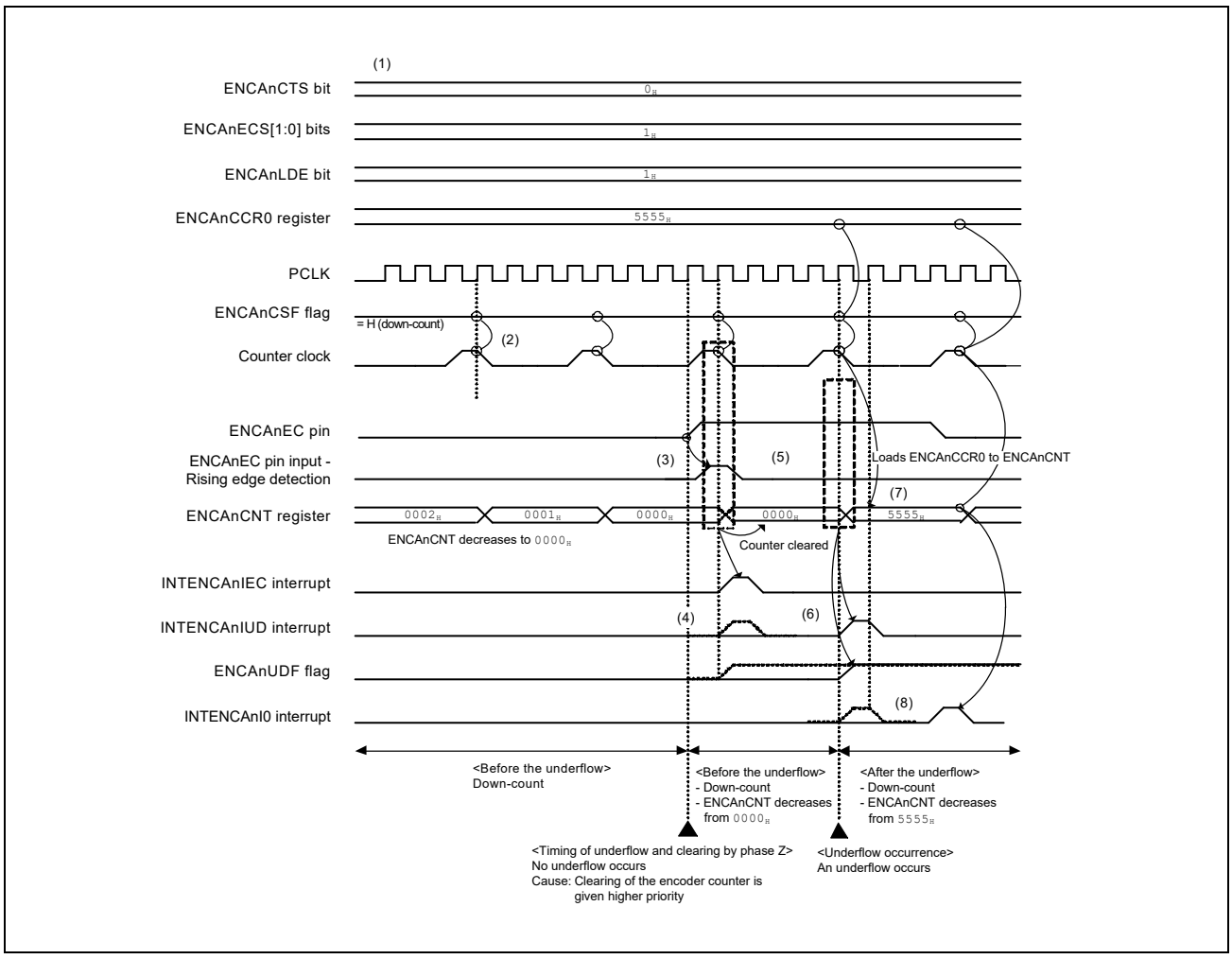


Figure 23.28 Conflict between ENCAnLDE Function and Clear Operation by Encoder Clear Input

1. The values are set as follows: ENCAnCTS = 0, ENCAnECS[1:0] = 01_B, ENCAnLDE = 1, and ENCAnCCR0 = 5555_H.
2. Down-counting is performed: $0002_H \rightarrow 0001_H \rightarrow 0000_H$
3. When the counter value becomes 0000_H, the rising edge of the ENCAnEC pin is detected, and clear operation by the encoder clear input is performed.
4. Because a count clear is performed when the counter value reaches 0000_H, a counter clear interrupt (INTENCAnIEC) by the encoder clear input is output. An underflow does not occur because down-counting is not performed when the counter value is 0000_H. Therefore, an underflow interrupt (INTENCAnIUD) is not output, and the underflow flag (ENCAnUDF) is not set.
5. After the counter value is cleared to 0000_H by clear operation by the encoder clear input, down-counting is performed and an underflow occurs.
6. An underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.
7. Because ENCAnLDE = 1, if an underflow occurs, the ENCAnCCR0 value is loaded to ENCAnCNT.
8. After the ENCAnCCR0 value is set to ENCAnCNT, a compare match is detected according to the counter clock. If the ENCAnCNT value matches with the ENCAnCCR0 value, a compare match interrupt (INTENCAnI0) is output.

23.6.17 Up-counting after Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input

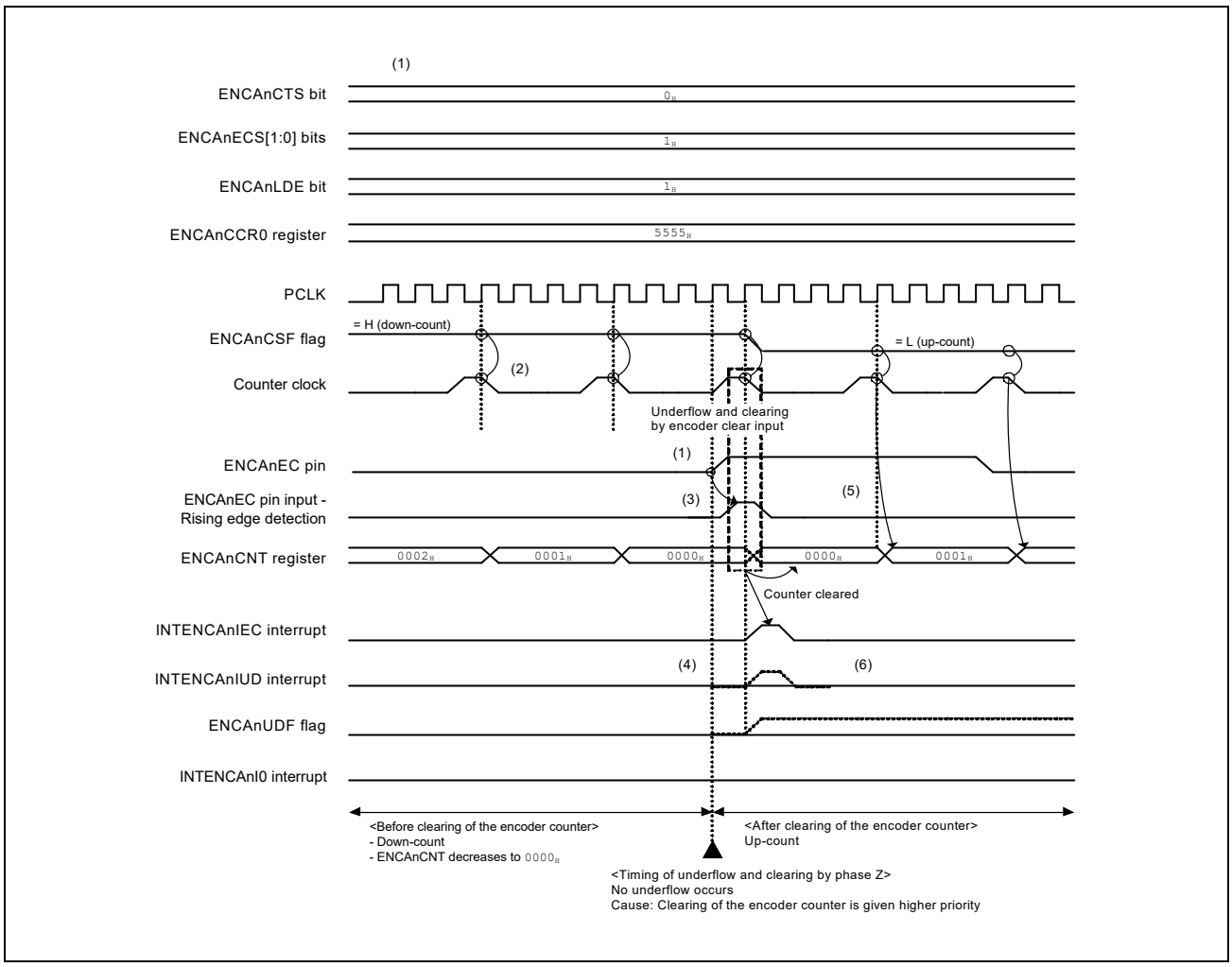


Figure 23.29 Up-counting after Conflict between ENCAnLDE Function and Clearing of the Encoder Counter

1. The values are set as follows: ENCAnCTS = 0, ENCAnECS[1:0] = 01_B, ENCAnLDE = 1, and ENCAnCCR0 = 5555_H.
2. Down-counting is performed: $0002_H \rightarrow 0001_H \rightarrow 0000_H$
3. When the counter value reaches 0000_H, the rising edge of ENCAnEC is detected, and clear operation by the encoder clear input is performed.
4. Upon clearing of the counter, a counter clear interrupt (INTENCAnIEC) triggered by the encoder clear input is output. Here, an underflow will not occur because further down-counting was not performed when the counter value was 0000_H. This means that neither an underflow interrupt (INTENCAnIUD) is output nor the underflow flag (ENCAnUDF) is set.
5. After the counter value has been cleared to 0000_H, up-counting is performed.
6. An underflow interrupt (INTENCAnIUD) is not output, and the underflow flag (ENCAnUDF) is not set.

23.6.18 Capture Operation between Counter Clocks (ENCAnCCR1)

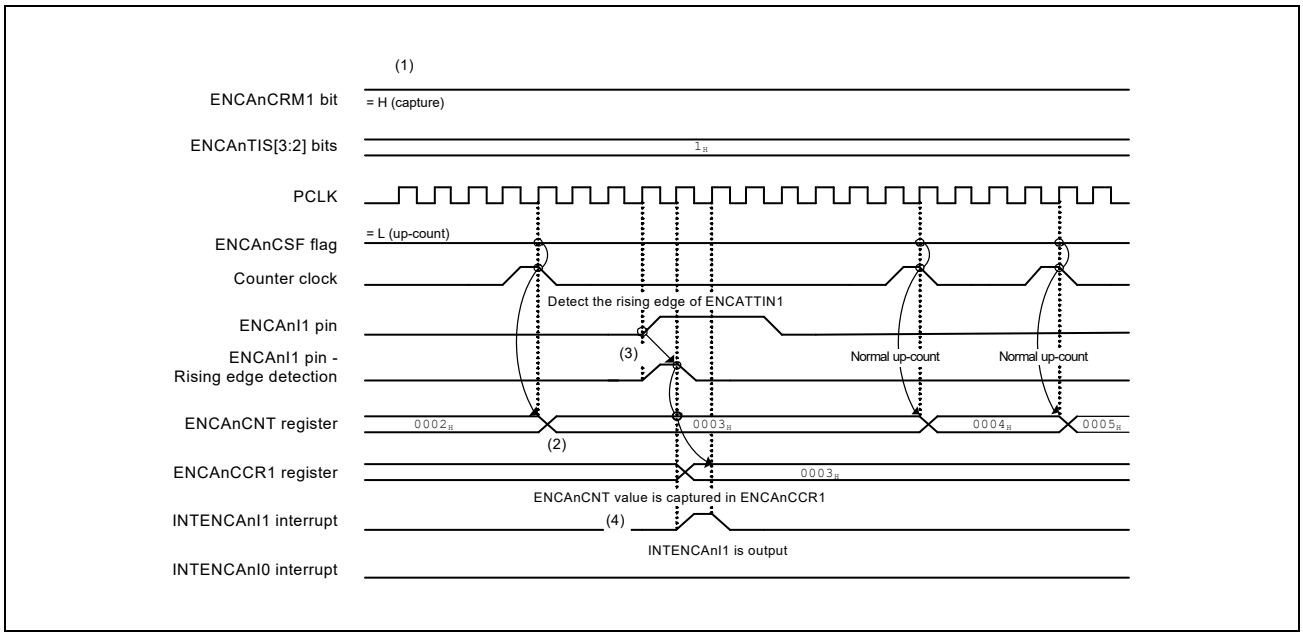


Figure 23.30 Capture Operation between Counter Clocks (ENCAnCCR1)

1. The values are set as follows: ENCAnCRM1 = 1 and ENCAnTIS[3:2] = 01_B.
2. Up-counting is performed.
3. The rising edge of the ENCAnI1 input is detected, and the counter value is captured in ENCAnCCR1.
4. An interrupt (INTENCAnI1) corresponding to the capture into the ENCAnCCR1 register is output.

23.6.19 Capture Operation between Counter Clocks (ENCAnCCR0)

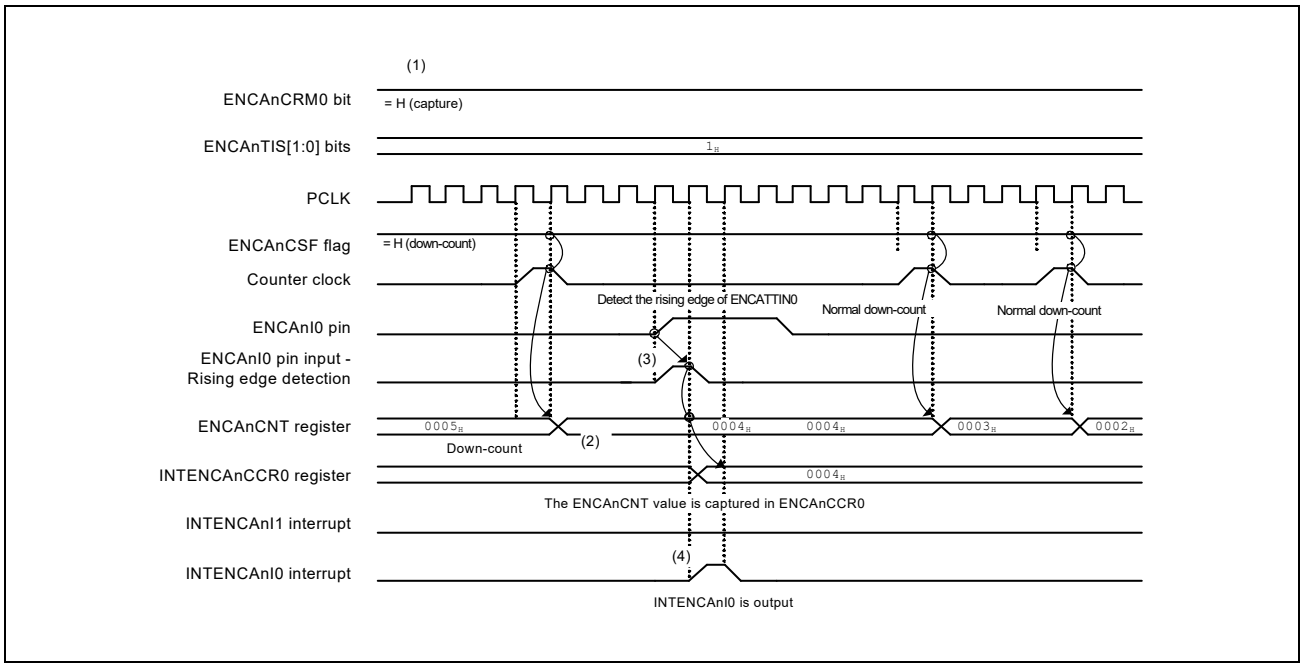


Figure 23.31 Capture Operation between Counter Clocks (ENCAnCCR0)

1. The values are set as follows: ENCAnCRM0 = 1 and ENCAnTIS[1:0] = 01_B.
2. Down-counting is performed.
3. The rising edge of the ENCAnI0 input is detected, and the counter value is captured in ENCAnCCR0.
4. An interrupt (INTENCAnI0) corresponding to the capture into the ENCAnCCR0 register is output.

23.6.20 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 0

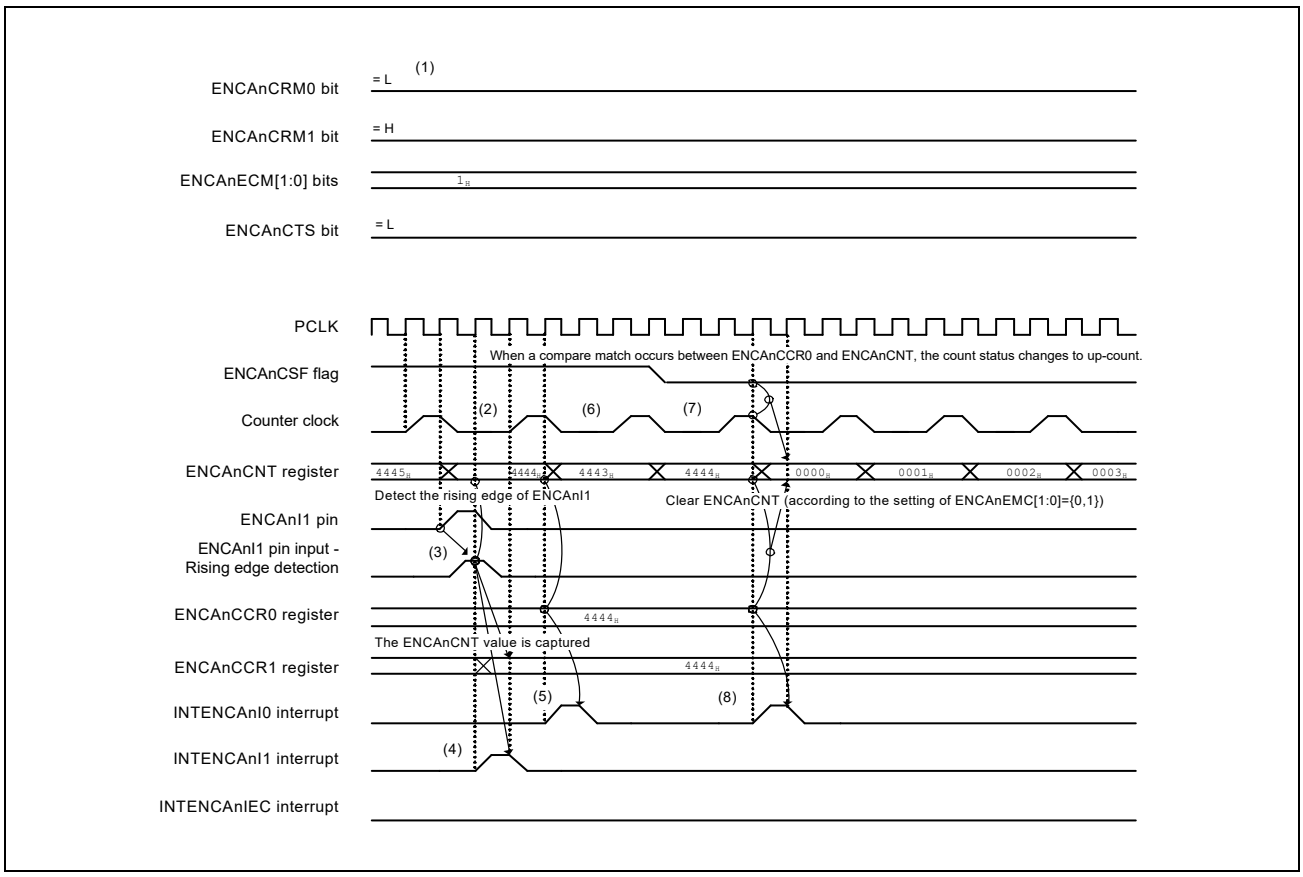


Figure 23.32 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 0

1. The values are set as follows: ENCAnCCR0 = 4444_H, ENCAnCRM0 = 0, ENCAnCRM1 = 1, ENCAnECM[1:0] = 01_B, and ENCAnCTS = 0.
2. Down-counting is performed.
3. The rising edge of the ENCAnI1 input is detected, and the ENCAnCNT value (4444_H) is captured in the ENCAnCCR1 register.
4. An interrupt signal (INTENCAnI1) corresponding to the capture into the ENCAnCCR1 register is output.
5. When a compare match occurs between ENCAnCNT (counted down from 4445_H to 4444_H) and ENCAnCCR0 (4444_H), a compare match interrupt (INTENCAnI0) with ENCAnCCR0 is output.
6. The counter operation changes to up-counting.
7. When ENCAnCNT is counted up from 4443_H to 4444_H, a compare match with ENCAnCCR0 occurs again. Because the counter operation is up-counting when the compare match occurs, the counter value is cleared according to the setting of ENCAnECM[1:0] (01_B), and the ENCAnCNT value changes to 0000_H.
8. When ENCAnCNT changes to 4444_H, a compare match interrupt (INTENCAnI0) with ENCAnCCR0 is output.

23.6.21 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 1

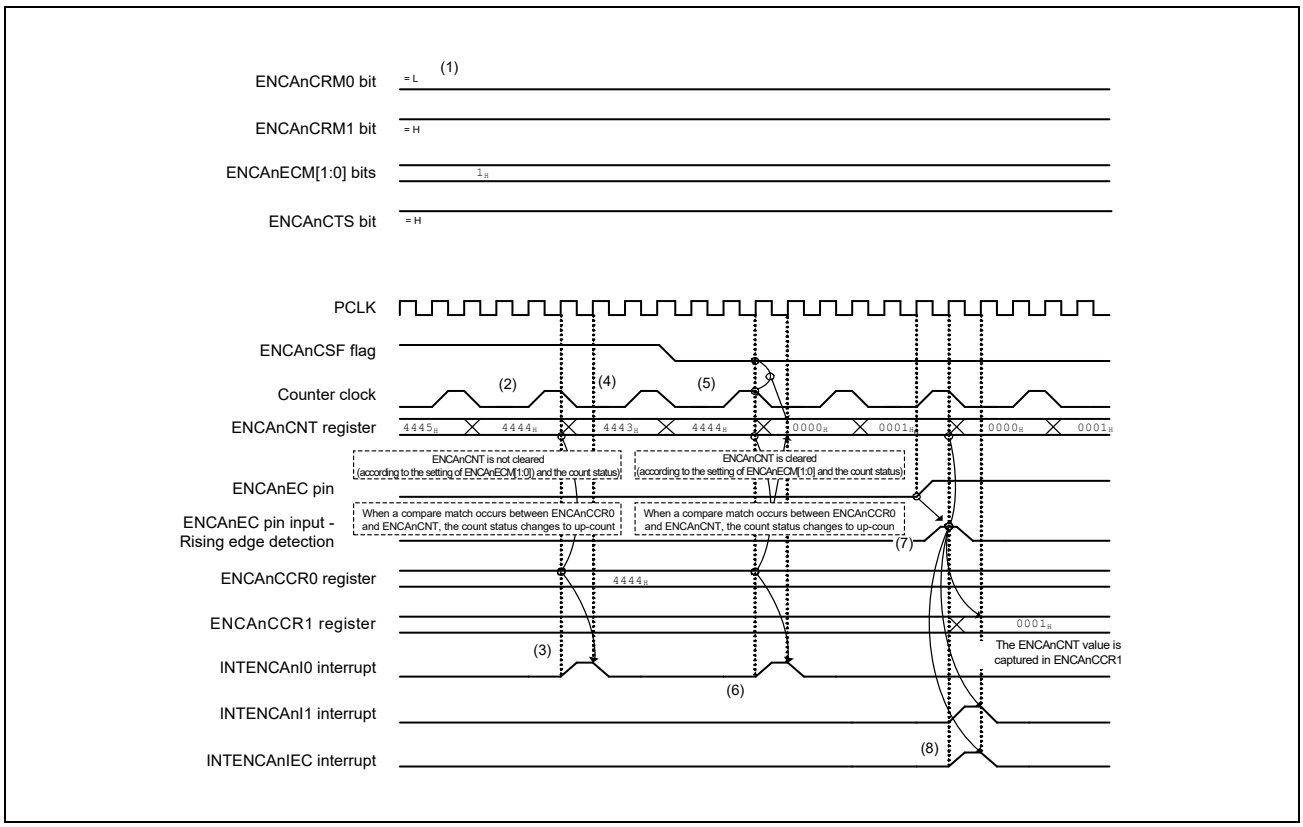


Figure 23.33 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 1

1. The values are set as follows: ENCAnCCR0 = 4444_H, ENCAnCRM0 = 0, ENCAnCRM1 = 1, ENCAnECM[1:0] = 01_B, and ENCAnCTS = 1.
2. Down-counting is performed.
3. When a compare match occurs between ENCAnCNT (counted down from 4445_H to 4444_H) and ENCAnCCR0 (4444_H), a compare/capture interrupt 0 (INTENCAnI0) is output.
4. The counter operation changes to up-counting.
5. When ENCAnCNT is counted up from 4443_H to 4444_H, a compare match with ENCAnCCR0 occurs again. Because the counter operation is up-counting when the compare match occurs, the counter value is cleared according to the setting of ENCAnECM[1:0] (01_B), and the ENCAnCNT value changes to 0000_H.
6. When ENCAnCNT changes to 4444_H, a compare match interrupt (INTENCAnI0) with ENCAnCCR0 is output.
7. After the counter value is cleared, up-counting is performed, and the counter value changes to 0001_H. At this point, the ENCAnCNT value (0001_H) is captured in ENCAnCCR1 by detecting the rising edge of the ENCAnEC signal, and the counter is cleared to 0000_H.
8. An interrupt (INTENCAnI1) corresponding to the capture to the ENCAnCCR1 register and a clear interrupt (INTENCAnIEC) by ENCAnEC are output.

23.6.22 Encoder Operation when Compare Match Clear Control is Disabled

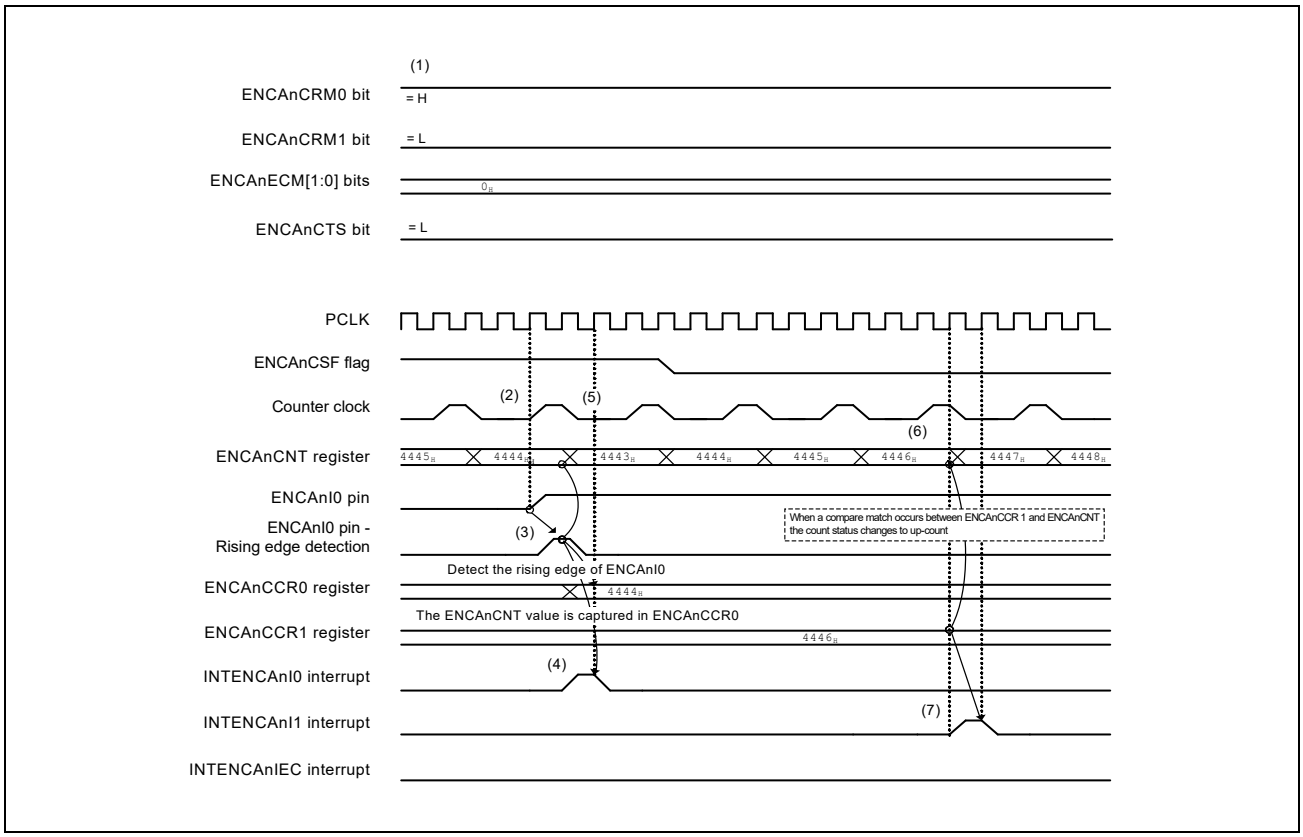


Figure 23.34 Encoder Operation when Compare Match Clear Control is Disabled

1. The values are set as follows: ENCAnCCR1 = 4446_H, ENCAnCRM0 = 1, ENCAnCRM1 = 0, ENCAnECM[1:0] = 00_B, and ENCAnCTS = 0.
2. Down-counting is performed.
3. When the rising edge of ENCAnI0 is detected, the ENCAnCNT value (4444_H) is captured in ENCAnCCR0.
4. An interrupt signal (INTENCAnI0) corresponding to the capture to the ENCAnCCR0 register is output
5. The counter operation changes to up-count.
6. When ENCAnCNT changes to 4446_H, a compare match with ENCAnCCR1 is detected.
7. A compare match interrupt (INTENCAnI1) with ENCAnCCR1 is output.

23.6.23 Capture Operation Performed upon Clearing by ENCAnEC, ENCAnE0, and ENCAnE1 when ENCAnSCE = 1

23.6.23.1 Accompanying Capture Operation

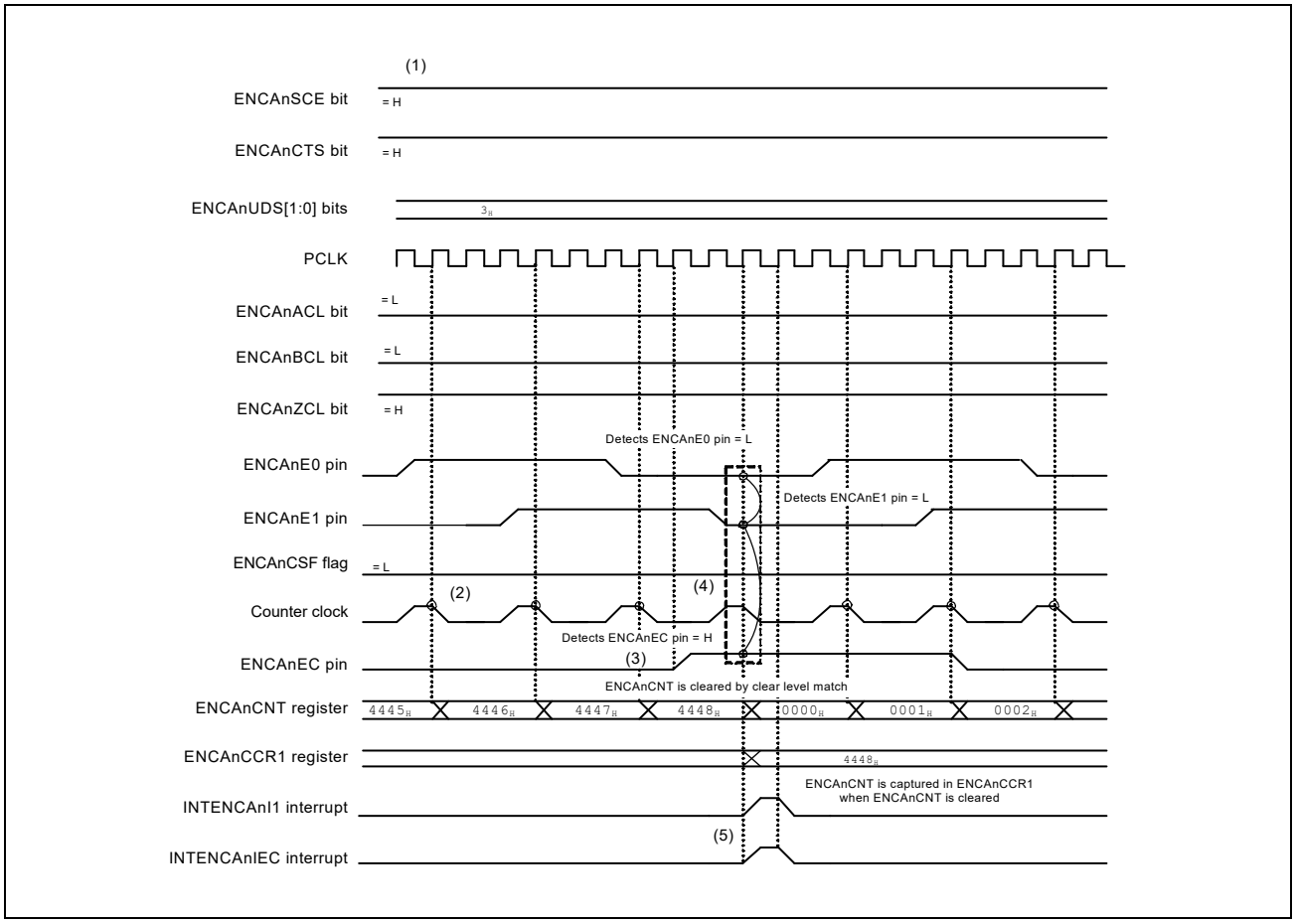


Figure 23.35 Capture Operation Performed upon Clearing by ENCAnEC, ENCAnE0, and ENCAnE1 when ENCAnSCE = 1

1. The values are set as follows: ENCAnSCE = 1, ENCAnCTS = 1, ENCAnUDS[1:0] = 11_B, ENCAnACL = 0, ENCAnBCL = 0, and ENCAnZCL = 1.
2. Up-counting is performed.
3. The counter value is not cleared upon the rising edge of ENCAnEC.
4. When ENCAnE0, ENCAnE1, and ENCAnEC reach the set clear level, the counter value is cleared. The counter value is captured in ENCAnCCR1 at the time of the clearing.
5. At the time of the clearing, an interrupt (INTENCAnI1) corresponding to the capture to the ENCAnCCR1 register and a clear interrupt (INTENCAnIEC) by ENCAnEC are output.

23.6.23.2 When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Up-count

(When ENCAnACL = 1, ENCAnBCL = 0, ENCAnZCL = 1, and ENCAnUDS[1:0] = 11_B)

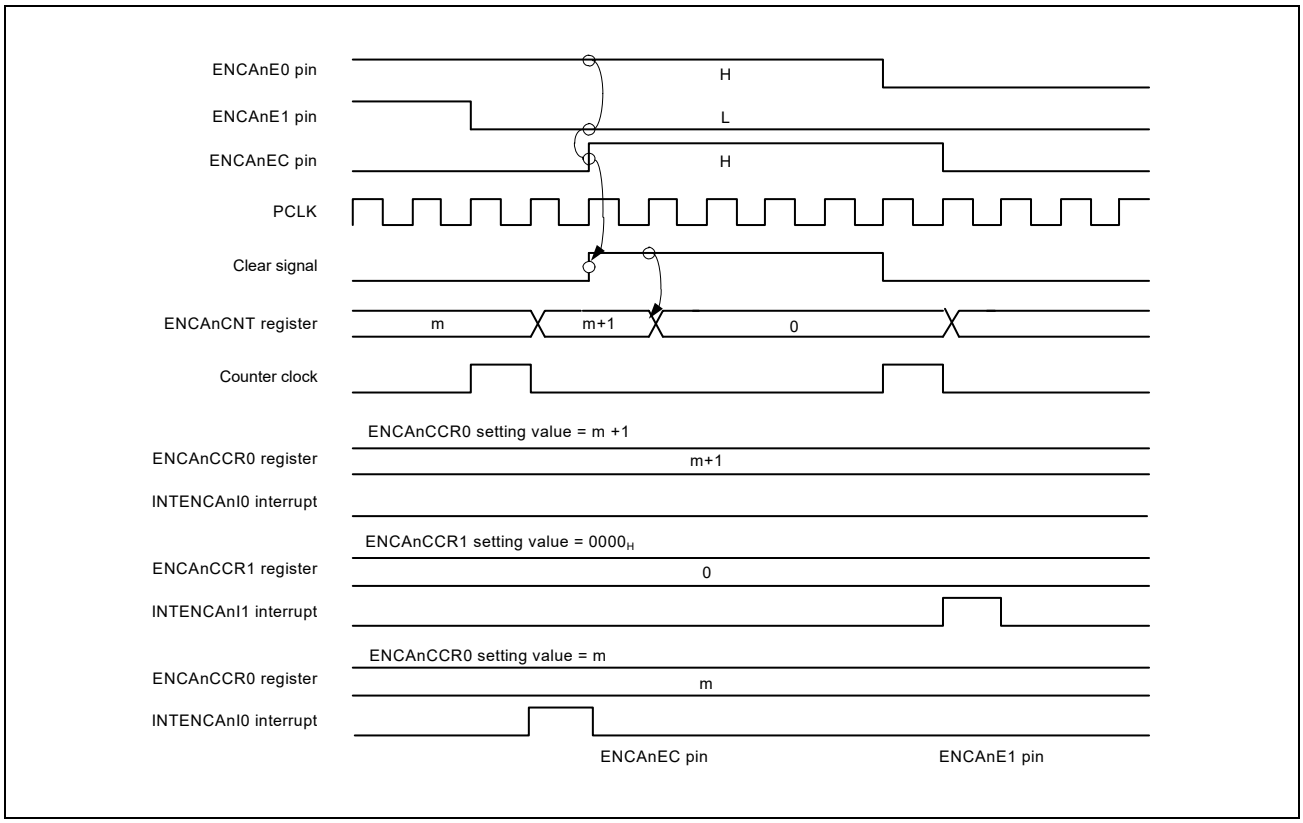


Figure 23.36 Clearing for when the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input During Up-Count

23.6.23.3 When the Timing of the ENCAnEC Input is the Same as that of the ENCAnE1 Input during Up-count

(When ENCAnACL = 1, ENCAnBCL = 0, ENCAnZCL = 1, and ENCAnUDS[1:0] = 11_B)

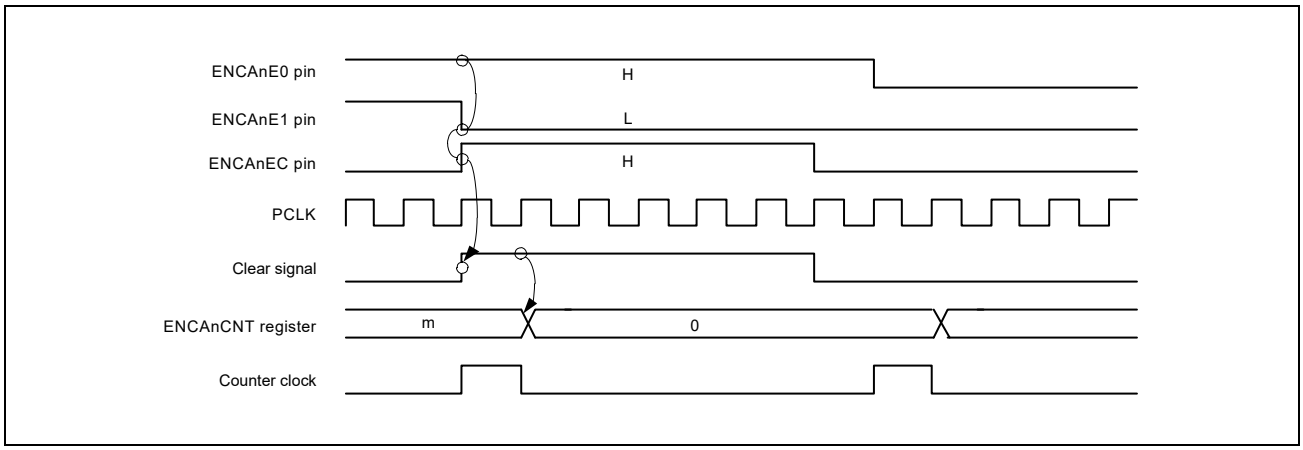


Figure 23.37 Clearing for when the Timing of the ENCAnEC Input is the Same as that of the ENCAnE1 Input During Up-Count

23.6.23.4 When the Timing of the ENCAnEC Input is Earlier than that of the ENCAnE1 Input during Up-count

(When ENCAnACL = 1, ENCAnBCL = 0, ENCAnZCL = 1, and ENCAnUDS[1:0] = 11_B)

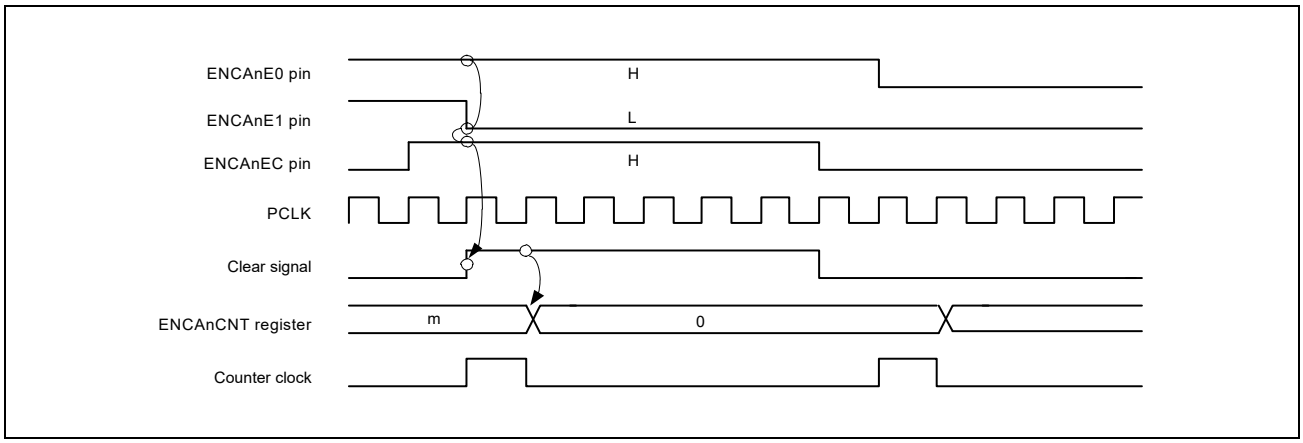


Figure 23.38 Clearing for when the Timing of the ENCAnEC Input is Earlier than that of the ENCAnE1 Input During Up-Count

23.6.23.5 When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Down-count

(When ENCAnACL = 1, ENCAnBCL = 0, ENCAnZCL = 1, and ENCAnUDS[1:0] = 11_B)

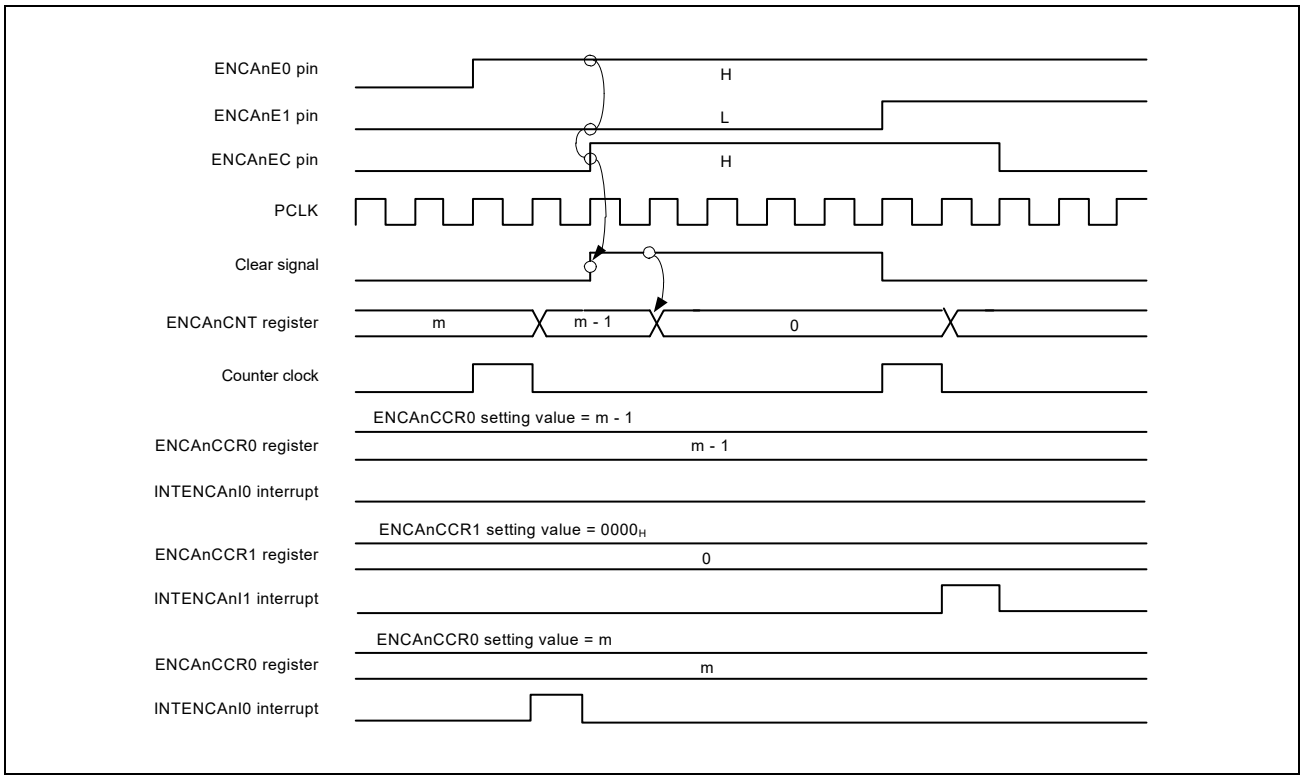


Figure 23.39 Clearing for when the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input During Down-count

23.6.24 Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0

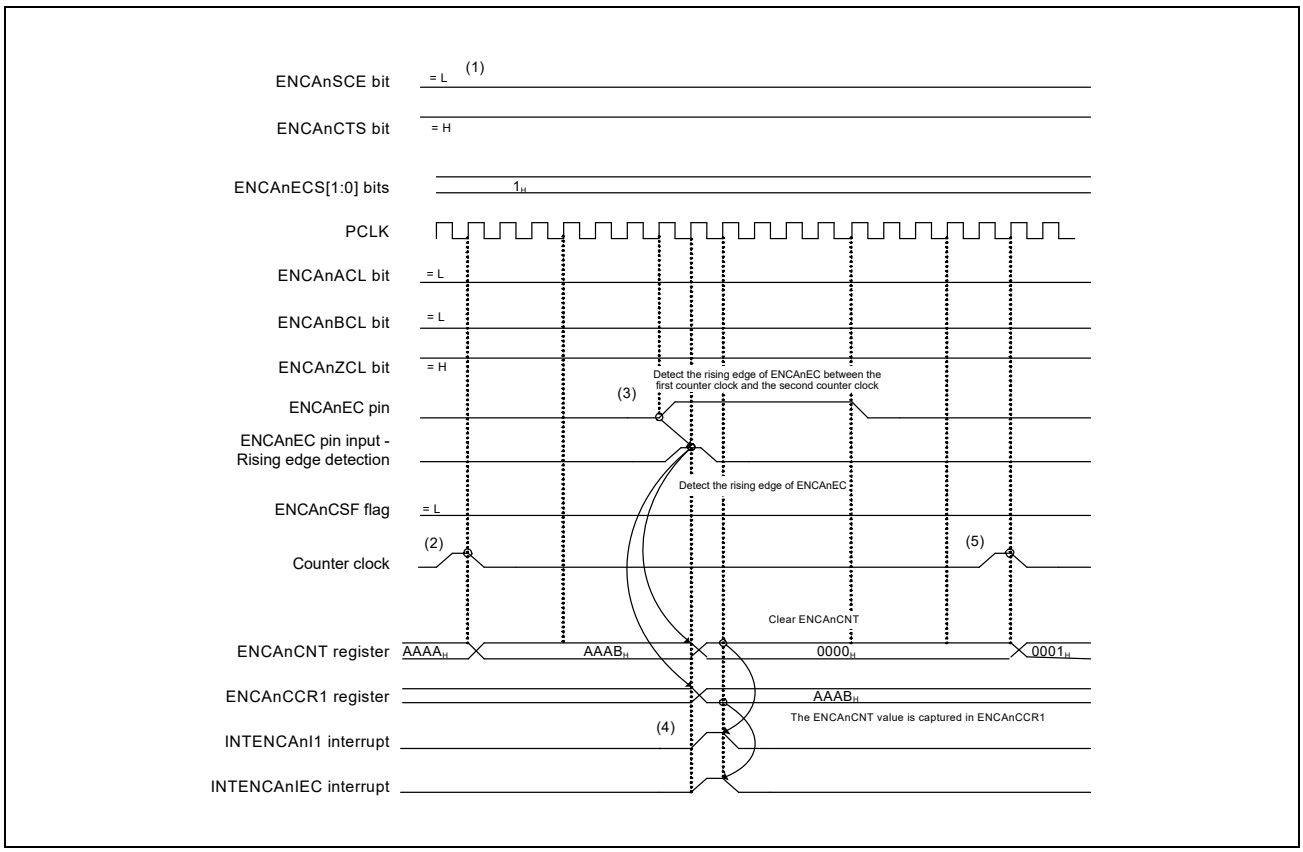


Figure 23.40 Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0

1. The values are set as follows: ENCAnSCE = 0, ENCAnCTS = 1, and ENCAnECS[1:0] = 01_B.
2. Up-counting is performed.
3. The rising edge of the ENCAnEC input is detected, and the ENCAnCNT value ($AAAB_H$) is captured in the ENCAnCCR1 register. Concurrently, clear operation by ENCAnEC is performed, and ENCAnCNT is cleared to 0000_H.
4. A capture interrupt 1 (INTENCAnI1) to the ENCAnCCR1 register and an encoder clear interrupt (INTENCAnIEC) by ENCAnEC are output.
5. After the counter value is cleared, up-counting is performed, and the counter value changes to 0001_H.

Section 24 Peripheral Interconnection (PIC)

24.1 Features of RH850/C1M-A PIC

This section contains a generic description of the peripheral interconnection (PIC).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of the PIC (PIC1B, PIC2D).

24.1.1 Number of Units

This microcontroller has the following number of units of the PIC.

Table 24.1 Number of Units (PIC1B)

Product Name	RH850/C1M-A2	RH850/C1M-A1
Number of Units	2	1
Name	PIC1Bk (k = 0, 1)	PIC1Bk (k = 0)

Table 24.2 Number of Units (PIC2D)

Product Name	RH850/C1M-A2	RH850/C1M-A1
Number of Units	1	
Name	PIC2D	

Table 24.3 Indices

Index	Meaning
k	The individual unit of PIC1B is identified by the index “k”.
n	The individual unit of each timer and A/D converter is identified by the index “n”.
m	The individual channel of each timer and A/D converter is identified by the index “m”.
x	The individual scan group of A/D converter is identified by the index “x”.
i	The variable used for descriptions is indicated by the index “i”.

24.1.2 Register Base Address

PIC base addresses are listed in the following table.

PIC register addresses are given as offsets from the base addresses in general.

Table 24.4 Register Base Address

Base Address Name	Base Address
<PIC1B0_base>	FFDD 0000_H
<PIC1B1_base>	FF7D 2000_H
<PIC2D_base>	FFDD 1000_H

24.1.3 Clock Supply

PIC clocks are listed in the following table.

Table 24.5 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
PIC1Bk	PCLK	CLKC_HSB (non-modulated high-speed peripheral clock)
PIC2D	PCLK	CLKC_HSB (non-modulated high-speed peripheral clock)

24.1.4 Reset Sources

PIC reset sources are listed in the following table. PIC is initialized by these reset sources.

Table 24.6 Reset Sources

Unit Name	Reset Source
PIC1Bk	Reset by all reset sources.
PIC2D	Reset by all reset sources.

24.1.5 External Input/Output Signals

External input/output signals of PIC are listed in the following table.

Table 24.7 PIC1B External Input/Output Signals

Unit Signal Name	Description	Alternative Port Pin Signal Name
ENCAnI1	ENCAn capture trigger input 1	ENCAnTIN1
ENCAnE0	ENCAn encoder input (count pulse 0)	ENCAnE0
ENCAnE1	ENCAn encoder input (count pulse 1)	ENCAnE1
ENCAnEC	ENCAn encoder input (clear pulse)	ENCAnEC
TAUDnTINm	TAUDn channel m	TAUDnIm
ESOn	Hi-Z control	TAPAnESO
TSG3nO1-6	TSG3n channel output 1 to 6	TSG3nO1-6
TOPnU	Motor control output U phase	TAPAnUP
TOPnUB	Motor control output UB phase	TAPAnUN
TOPnV	Motor control output V phase	TAPAnVP
TOPnVB	Motor control output VB phase	TAPAnVN
TOPnW	Motor control output W phase	TAPAnWP
TOPnWB	Motor control output WB phase	TAPAnWN
TSGTSST	TSG external simultaneous start trigger input	TSGTRG

Table 24.8 PIC2D External Input/Output Signals

Unit Signal Name	Description	Alternative Port Pin Signal Name
ADTRGnZ	ADCCn trigger	ADCCnTRG

24.2 Peripheral Interconnection 1 (PIC1B)

24.2.1 Overview

24.2.1.1 Functional Overview

The peripheral interconnection 1 (PIC1B) handles synchronous operation using multiple timers and by connecting the timer internal I/O signals between the timers.

CAUTIONS

- The signal names used in the following descriptions are abbreviations. The actual signal names corresponding to each abbreviation are as follows:
  INTm → TAUDnTINTm
  TINm → TAUDnTTINm
  TOUTm → TAUDnTTOUTm
  CDRm → TAUDnCDRm
  CNTm → TAUDnCNTm
  TSGTSST → TSGTRG
- The functions of the ENCA internal signals used in the following descriptions are as follows:
  ENCATnEQ0 : Internal signals to be output one PCLK cycle before INTENCAnI0 interrupt request signals
  ENCATnEQ1 : Internal signals to be output one PCLK cycle before INTENCAnI1 interrupt request signals
  ENCATnIEC : Internal signals of INTENCAnIEC interrupt request signals

The PIC1B has the following functions.

- Simultaneous start trigger function
- TSG simultaneous start trigger function (external trigger)
- PWM output function with dead time
- High accuracy triangle wave PWM output function with dead time
- Delay pulse output function with dead time
- Trigger and pulse width measurement function
- Encoder capture trigger select function
- Two-phase encoder control function (control method 1)
- Two-phase encoder control function (control method 2)
- Two-phase encoder control function (control method 3)
- Three-phase pulse input control function
- Three-phase encoder control function
- ENCA input select function
- TAUD input select function
- Switch function between TSG output and low/high level output
- Hi-Z control function

Table 24.9 lists the related modules by functionality.

Table 24.9 Related Modules by Functionality

Section Number	Function Name	Related Module*2 / Input Pin*2		
		PIC1B0	PIC1B1*1	Two Units Coupled*1
24.2.3.1	**Simultaneous Start Trigger Function**	TAUD0, TAUD1 TAUJ0 TSG30, TSG31 TPBA0, TPBA1 ENCA0, ENCA1 OSTM0, OSTM1, OSTM2	TAUD2, TAUD3 TAUJ1 TSG32 — — OSTM3	TAUD0, TAUD1, TAUD2, TAUD3 TAUJ0, TAUJ1 TSG30, TSG31, TSG32 TPBA0, TPBA1 ENCA0, ENCA1 OSTM0, OSTM1, OSTM2, OSTM3
24.2.3.2	**TSG Simultaneous Start Function (External Trigger)**	TSG30, TSG31	TSG32	TSG30, TSG31, TSG32
24.2.3.3	**PWM Output Function with Dead Time**	TAUD0, TAUD1	TAUD2, TAUD3	—
24.2.3.4	**High Accuracy Triangle Wave PWM Output Function with Dead Time**	TAUD0, TAUD1	TAUD2, TAUD3	—
24.2.3.5	**Delay Pulse Output Function with Dead Time**	TADU0, TAUD1	TAUD2, TAUD3	—
24.2.3.6	**Trigger and Pulse Width Measurement Function**	TADU0, TAUD1 TAUJ0 ENCA0, ENCA1	—	—
24.2.3.7	**Encoder Capture Trigger Select Function**	TAUD0, TAUD1, ENCA0, ENCA1 PIC2D	—	—
24.2.3.8	**Two-Phase Encoder Control Function (Control Method 1)**	TSG30, TSG31 ENCA0, ENCA1	—	—
24.2.3.9	**Two-Phase Encoder Control Function (Control Method 2)**	TSG30, TSG31 ENCA0, ENCA1	—	—
24.2.3.10	**Two-Phase Encoder Control Function (Control Method 3)**	TSG30, TSG31 ENCA0, ENCA1	—	—
24.2.3.11	**Three-Phase Pulse Input Control Function**	TAUD0, TAUD1 TSG30, TSG31 ENCA0, ENCA1 (pin)	—	—
24.2.3.12	**Three-Phase Encoder Function**	TSG30, TSG31 ENCA0, ENCA1	—	—
24.2.3.13	**ENCA Input Select Function**	ENCA0, ENCA1 RDC3A0, RDC3A1	—	—
24.2.3.14	**TAUD Input Select Function**	TAUD0, TAUD1	TAUD2, TAUD3	—
24.2.3.15	**Switch Function between TSG Output and Low/High Level Output**	TSG30, TSG31	TSG32	—
24.2.3.16	**Hi-Z Control Function**	TAUD0, TAUD1 TSG30, TSG31 TAPA0, TAPA1, TAPA3, TAPA4 ECM ES00, ES01, ES03, ES04 (pin)	TAUD2 TSG32 TAPA2, TAPA5 ECM ES02, ES05 (pin)	—

Note 1. PIC1B1 is not provided for the C1M-A1 and thus coupled operation of the two units are not possible.

Note 2. TSG32, TAPA5, TAPA2, TAUD3, TAUD2, TAUJ1, TPBA1, OSTM3, RDC3A1, ESO5, and ESO2 are not provided for the C1M-A1.

24.2.2 Registers

24.2.2.1 List of Registers

The registers are listed in the following table.

The bits can be accessed only in 32-bit units. Access in 16-bits or 8-bits is operated as 32-bit access.

See **Section 24.1.2, Register Base Address** for <PIC1Bk_base>.

Table 24.10 Registers (1/2)

Module Name	Register	Symbol	Address
PIC1B0	Simultaneous start trigger control register 0	PIC1BSST0	<PIC1B0_base> + 04_H
PIC1B0	TSG3 simultaneous start trigger select register 0	PIC1BSSTSGSEL0	<PIC1B0_base> + 08_H
PIC1B0	Simultaneous start trigger output control register 0*1	PIC1BSSTOUTEN0	<PIC1B0_base> + $0C_H$
PIC1B0	Simultaneous start control register 00	PIC1BSSER00	<PIC1B0_base> + 10_H
PIC1B0	Simultaneous start control register 01	PIC1BSSER01	<PIC1B0_base> + 14_H
PIC1B0	Simultaneous start control register 02	PIC1BSSER02	<PIC1B0_base> + 18_H
PIC1B0	Simultaneous start control register 03	PIC1BSSER03	<PIC1B0_base> + $1C_H$
PIC1B0	RS flip-flop circuit initialization register 00	PIC1BINI00	<PIC1B0_base> + 20_H
PIC1B0	DT initialization register 01	PIC1BINI01	<PIC1B0_base> + 24_H
PIC1B0	RS flip-flop circuit initialization register 10	PIC1BINI10	<PIC1B0_base> + $2C_H$
PIC1B0	DT initialization register 11	PIC1BINI11	<PIC1B0_base> + 30_H
PIC1B0	TSG30 output low/high level select register	PIC1BLHSEL0	<PIC1B0_base> + 60_H
PIC1B0	TSG30 output control register	PIC1BTSGOUTCTR0	<PIC1B0_base> + 64_H
PIC1B0	TSG31 output low/high level select register	PIC1BLHSEL1	<PIC1B0_base> + 68_H
PIC1B0	TSG31 output control register	PIC1BTSGOUTCTR1	<PIC1B0_base> + $6C_H$
PIC1B0	Hall sensor input select register	PIC1BTSGHALLSEL	<PIC1B0_base> + 74_H
PIC1B0	TAUD0 input select register	PIC1BTAUD0SEL	<PIC1B0_base> + 78_H
PIC1B0	TAUD1 input select register	PIC1BTAUD1SEL	<PIC1B0_base> + $7C_H$
PIC1B0	Hi-Z control register 00	PIC1BHIZCEN00	<PIC1B0_base> + 80_H
PIC1B0	Hi-Z control register 01	PIC1BHIZCEN01	<PIC1B0_base> + 84_H
PIC1B0	Hi-Z control register 02	PIC1BHIZCEN02	<PIC1B0_base> + 88_H
PIC1B0	Hi-Z control register 03	PIC1BHIZCEN03	<PIC1B0_base> + $8C_H$
PIC1B0	ENCATIN 1 input select register 400	PIC1BENCSEL400	<PIC1B0_base> + $B8_H$
PIC1B0	ENCATIN 1 input select register 410	PIC1BENCSEL410	<PIC1B0_base> + BC_H
PIC1B0	Timer input/output control register 200	PIC1BREG200	<PIC1B0_base> + $C0_H$
PIC1B0	Timer input/output control register 201	PIC1BREG201	<PIC1B0_base> + $C4_H$
PIC1B0	Timer input/output control register 202	PIC1BREG202	<PIC1B0_base> + $C8_H$
PIC1B0	Timer input/output control register 203	PIC1BREG203	<PIC1B0_base> + CC_H
PIC1B0	Timer input/output control register 210	PIC1BREG210	<PIC1B0_base> + $D4_H$
PIC1B0	Timer input/output control register 211	PIC1BREG211	<PIC1B0_base> + $D8_H$
PIC1B0	Timer input/output control register 212	PIC1BREG212	<PIC1B0_base> + DC_H
PIC1B0	Timer input/output control register 213	PIC1BREG213	<PIC1B0_base> + $E0_H$
PIC1B0	Timer input/output control register 30	PIC1BREG30	<PIC1B0_base> + $E8_H$
PIC1B0	Timer input/output control register 31	PIC1BREG31	<PIC1B0_base> + EC_H
PIC1B0	Timer input/output control register 50	PIC1BREG50	<PIC1B0_base> + $F8_H$
PIC1B0	Timer input/output control register 51	PIC1BREG51	<PIC1B0_base> + FC_H

Table 24.10 Registers (2/2)

Module Name	Register	Symbol	Address
PIC1B1	Simultaneous start trigger control register 1*1	PIC1BSST1	<PIC1B1_base> + 04_H
PIC1B1	TSG3 simultaneous start trigger select register 1*1	PIC1BSSTSGSEL1	<PIC1B1_base> + 08_H
PIC1B1	Simultaneous start trigger output control register 1*1	PIC1BSSTOUTEN1	<PIC1B1_base> + $0C_H$
PIC1B1	Simultaneous start control register 10*1	PIC1BSSER10	<PIC1B1_base> + 10_H
PIC1B1	Simultaneous start control register 11*1	PIC1BSSER11	<PIC1B1_base> + 14_H
PIC1B1	Simultaneous start control register 12*1	PIC1BSSER12	<PIC1B1_base> + 18_H
PIC1B1	Simultaneous start control register 13*1	PIC1BSSER13	<PIC1B1_base> + $1C_H$
PIC1B1	RS flip-flop circuit initialization register 20*1	PIC1BINI20	<PIC1B1_base> + 20_H
PIC1B1	RS flip-flop circuit initialization register 30*1	PIC1BINI30	<PIC1B1_base> + $2C_H$
PIC1B1	TSG32 output low/high level select register*1	PIC1BLHSEL2	<PIC1B1_base> + 60_H
PIC1B1	TSG32 output control register*1	PIC1BTSGOUTCTR2	<PIC1B1_base> + 64_H
PIC1B1	TAUD2 input select register*1	PIC1BTAUD2SEL	<PIC1B1_base> + 78_H
PIC1B1	TAUD3 input select register*1	PIC1BTAUD3SEL	<PIC1B1_base> + $7C_H$
PIC1B1	Hi-Z control register 10*1	PIC1BHIZCEN10	<PIC1B1_base> + 80_H
PIC1B1	Hi-Z control register 12*1	PIC1BHIZCEN12	<PIC1B1_base> + 88_H
PIC1B1	Timer input/output control register 220*1	PIC1BREG220	<PIC1B1_base> + $C0_H$
PIC1B1	Timer input/output control register 221*1	PIC1BREG221	<PIC1B1_base> + $C4_H$
PIC1B1	Timer input/output control register 222*1	PIC1BREG222	<PIC1B1_base> + $C8_H$
PIC1B1	Timer input/output control register 223*1	PIC1BREG223	<PIC1B1_base> + CC_H
PIC1B1	Timer input/output control register 230*1	PIC1BREG230	<PIC1B1_base> + $D4_H$
PIC1B1	Timer input/output control register 231*1	PIC1BREG231	<PIC1B1_base> + $D8_H$
PIC1B1	Timer input/output control register 232*1	PIC1BREG232	<PIC1B1_base> + DC_H
PIC1B1	Timer input/output control register 233*1	PIC1BREG233	<PIC1B1_base> + $E0_H$

Note 1. C1M-A1 is not provided with this register.

Combinations of registers used for each function are listed in the following table.

Table 24.11 Registers Used by Each Function

Section Number	Function Name	PIC1BSST	PIC1BSSTSGSEL	PIC1BSSTOUTEN	PIC1BSSER				PIC1BINI		PIC1BLHSEL	PIC1BTSGOUTCTR	PIC1BTSGHALLSEL	PIC1BTAUD0SEL	PIC1BTAUD1SEL	PIC1BTAUD2SEL	PIC1BTAUD3SEL	PIC1BHIZCEN	PIC1BBENCSEL	PIC1BREG							
		0, 1	0, 1	0, 1	00, 10	01, 11	02, 12	03, 13	00, 10, 20, 30	01, 11	0, 1, 2	0, 1, 2	—	—	—	—	—	00, 01, 02, 03, 10, 12	400, 410	200, 210, 220, 230	201, 211, 221, 231	202, 212, 222, 232	203, 213, 223, 233	30	31	50	51
24.2.3.1	Simultaneous Start Trigger Function	√	—	√	√	√	√	√	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
24.2.3.2	TSG Simultaneous Start Function (External Trigger)	—	√	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
24.2.3.3	PWM Output Function with Dead Time	—	—	—	—	—	—	—	√	—	—	—	—	√	√	√	√	—	—	—	√	√	√	—	—	—	—
24.2.3.4	High Accuracy Triangle Wave PWM Output Function with Dead Time	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	—	—	√	√	√	√	—	—	—	—
24.2.3.5	Delay Pulse Output Function with Dead Time	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	—	—	—	√	√	√	—	—	—	—
24.2.3.6	Trigger and Pulse Width Measurement Function	—	—	—	—	—	—	—	—	√	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	—	—
24.2.3.7	Encoder Capture Trigger Select Function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—	—	√	—	—	—
24.2.3.8	Two-Phase Encoder Control Function (Control Method 1)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	√	√
24.2.3.9	Two-Phase Encoder Control Function (Control Method 2)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	√	√
24.2.3.10	Two-Phase Encoder Control Function (Control Method 3)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	√	√
24.2.3.11	Three-Phase Pulse Input Control Function	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	—	—	√	—	—	—	—	—	√	√
24.2.3.12	Three-Phase Encoder Function	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—	—	—	—	—	—	—	—	√	—	√	√
24.2.3.13	ENCA Input Select Function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—
24.2.3.14	TAUD Input Select Function	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	—	—	√	—	√	—	—	√	—	—
24.2.3.15	Switch Function between TSG Output and Low/High Level Output	—	—	—	—	—	—	—	—	—	√	√	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
24.2.3.16	Hi-Z Control Function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—	—	—	—	—	—	—

24.2.2.2 PIC1BSST0 ─ Simultaneous Start Trigger Control Register 0

PIC1BSST0 is an 8-bit register that selects a simultaneous start trigger.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PIC1BSYNCTRG
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.12 PIC1BSST0 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BSYNCTRG	Generates a start trigger for the timer of PIC1B0 whose simultaneous start trigger is enabled. 0: Disabled 1: Simultaneous start trigger (a pulse with a width of 1 cycle of PCLK is output.)

Note: PIC1BSYNCTRG is always read as 0.

24.2.2.3 PIC1BSST1 ─ Simultaneous Start Trigger Control Register 1*1

PIC1BSST1 is an 8-bit register that selects a simultaneous start trigger.

Access: Readable/writable in 8-bit units.

Address: <PIC1B1_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PIC1BSYNCTRG
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.13 PIC1BSST1 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BSYNCTRG	Generates a start trigger for the timer of PIC1B1 whose simultaneous start trigger is enabled. 0: Disabled 1: Simultaneous start trigger (a pulse with a width of 1 cycle of PCLK is output.)

Note: PIC1BSYNCTRG is always read as 0.

Note 1. C1M-A1 is not provided with the PIC1BSST1 register.

24.2.2.4 PIC1BSSTSGSEL0 ─ TSG3 Simultaneous Start Trigger Select Register 0

PIC1BSSTSGSEL0 is an 8-bit register that selects a simultaneous start trigger for each of TSG31 and TSG30.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIC1BSS TSGSEL01	PIC1BSS TSGSEL00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 24.14 PIC1BSSTSGSEL0 Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1	PIC1BSSTSGSEL01	Selects a start trigger for TSG31. 0: Selects PIC1BSYNCTRG of the PIC1BSST0 register. 1: Selects an external trigger (TSGTSST).
0	PIC1BSSTSGSEL00	Selects a start trigger for TSG30. 0: Selects PIC1BSYNCTRG of the PIC1BSST0 register. 1: Selects an external trigger (TSGTSST).

24.2.2.5 PIC1BSSTSGSEL1 ─ TSG3 Simultaneous Start Trigger Select Register 1*1

PIC1BSSTSGSEL1 is an 8-bit register that selects a simultaneous start trigger for TSG32.

Access: Readable/writable in 8-bit units.

Address: <PIC1B1_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PIC1BSS TSGSEL10
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.15 PIC1BSSTSGSEL1 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BSSTSGSEL10	Selects a start trigger for TSG32. 0: Selects PIC1BSYNCTRG of the PIC1BSST1 register. 1: Selects an external trigger (TSGTSST).

Note 1. C1M-A1 is not provided with the PIC1BSSTSGSEL1 register.

24.2.2.6 PIC1BSSTOUTENk – Simultaneous Start Trigger Output Control Register k*1

The PIC1BSSTOUTENk registers enables or disables the output of simultaneous start triggers from one PIC1B unit to the other. Setting PIC1BSSTOUTEN00 of the PIC1BSSTOUTEN0 register and then PIC1BSYNCTRG0 of the PIC1BSST0 register to 1 sends a simultaneous start trigger to the PIC1B1 unit. Making the equivalent settings in the PIC1BSSTOUTEN1 and PIC1BSST1 registers sends a simultaneous start trigger to the PIC1B0 unit.

Access: Readable/writable in 8-bit units.

Address: <PIC1Bk_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PIC1BSSTOUTENk0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.16 PIC1BSSTOUTENk Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BSSTOUTENk0	Enables or disables output of simultaneous start triggers from the corresponding PIC1B unit to the other. 0: Disables the output of a simultaneous start trigger to the other PIC1B unit. 1: Enables the output of a simultaneous start trigger to the other PIC1B unit.

Note 1. This register is not provided for the C1M-A.

24.2.2.7 PIC1BSSERk0 – Simultaneous Start Control Register k0

PIC1BSSER00 enables a start trigger for each channel of TAUD0.
PIC1BSSER10 enables a start trigger for each channel of TAUD2.*[1]

Access: Readable/writable in 16-bit units.

Address: <PIC1Bk_base> + 10_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1BSSERk015	PIC1BSSERk014	PIC1BSSERk013	PIC1BSSERk012	PIC1BSSERk011	PIC1BSSERk010	PIC1BSSERk009	PIC1BSSERk008	PIC1BSSERk007	PIC1BSSERk006	PIC1BSSERk005	PIC1BSSERk004	PIC1BSSERk003	PIC1BSSERk002	PIC1BSSERk001	PIC1BSSERk000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.17 PIC1BSSERk0 Register Contents

Bit Position	Bit Name	Function
15 to 0	PIC1BSSERk015 to PIC1BSSERk000	Enable or disable a simultaneous start trigger for CHm in the TAUDn. 0: Disabled 1: Enabled

Note 1. The PIC1BSSER10 register is not provided for the C1M-A1.

24.2.2.8 PIC1BSSERk1 ─ Simultaneous Start Control Register k1

PIC1BSSER01 enables a start trigger for each channel of TAUD1.
PIC1BSSER11 enables a start trigger for each channel of TAUD3.*[1]

Access: Readable/writable in 16-bit units.

Address: <PIC1Bk_base> + 14_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1BSSERk115	PIC1BSSERk114	PIC1BSSERk113	PIC1BSSERk112	PIC1BSSERk111	PIC1BSSERk110	PIC1BSSERk109	PIC1BSSERk108	PIC1BSSERk107	PIC1BSSERk106	PIC1BSSERk105	PIC1BSSERk104	PIC1BSSERk103	PIC1BSSERk102	PIC1BSSERk101	PIC1BSSERk100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.18 PIC1BSSERk1 Register Contents

Bit Position	Bit Name	Function
15 to 0	PIC1BSSERk115 to PIC1BSSERk100	Enable or disable a simultaneous start trigger for CHm in the TAUDn. 0: Disabled 1: Enabled

Note 1. The PIC1BSSER11 register is not provided for the C1M-A1.

24.2.2.9 PIC1BSSER02 ─ Simultaneous Start Control Register 02

PIC1BSSER02 enables a start trigger for TAUJ0, TSG30, TSG31, TPBA0, TPBA1*[1], ENCA0, and ENCA1.

Access: Readable/writable in 16-bit units.

Address: <PIC1B0_base> + 18_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PIC1BSSER0213	PIC1BSSER0212	PIC1BSSER0211	PIC1BSSER0210	PIC1BSSER0209	PIC1BSSER0208	—	—	—	—	PIC1BSSER0203	PIC1BSSER0202	PIC1BSSER0201	PIC1BSSER0200
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 24.19 PIC1BSSER02 Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	Reserved	When read, the value after reset is read. When writing, write the value after reset.
13	PIC1BSSER0213	Enables or disables a simultaneous start trigger for the ENCA1. 0: Disabled 1: Enabled
12	PIC1BSSER0212	Enables or disables a simultaneous start trigger for the ENCA0. 0: Disabled 1: Enabled
11	PIC1BSSER0211	Enables or disables a simultaneous start trigger for the TPBA1.*[1] 0: Disabled 1: Enabled
10	PIC1BSSER0210	Enables or disables a simultaneous start trigger for the TPBA0. 0: Disabled 1: Enabled
9	PIC1BSSER0209	Enables or disables a simultaneous start trigger for the TSG31. 0: Disabled 1: Enabled
8	PIC1BSSER0208	Enables or disables a simultaneous start trigger for the TSG30. 0: Disabled 1: Enabled
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	PIC1BSSER0203	Enables or disables a simultaneous start trigger for the TAUJ0 CH03. 0: Disabled 1: Enabled
2	PIC1BSSER0202	Enables or disables a simultaneous start trigger for the TAUJ0 CH02. 0: Disabled 1: Enabled
1	PIC1BSSER0201	Enables or disables a simultaneous start trigger for the TAUJ0 CH01. 0: Disabled 1: Enabled

Table 24.19 PIC1BSSER02 Register Contents (2/2)

Bit Position	Bit Name	Function
0	PIC1BSSER0200	Enables or disables a simultaneous start trigger for the TAUJ0 CH00. 0: Disabled 1: Enabled

Note 1. This module is not provided for the C1M-A1.

24.2.2.10 PIC1BSSER12 – Simultaneous Start Control Register 12*1

PIC1BSSER12 enables a start trigger for TAUJ1 and TSG32.

Access: Readable/writable in 16-bit units.

Address: <PIC1B1_base> + 18_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PIC1BSSER1208	—	—	—	—	PIC1BSSER1203	PIC1BSSER1202	PIC1BSSER1201	PIC1BSSER1200
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 24.20 PIC1BSSER12 Register Contents

Bit Position	Bit Name	Function
15 to 9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	PIC1BSSER1208	Enables or disables a simultaneous start trigger for TSG32. 0: Disabled 1: Enabled
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	PIC1BSSER1203	Enables or disables a simultaneous start trigger for TAUJ1 CH03. 0: Disabled 1: Enabled
2	PIC1BSSER1202	Enables or disables a simultaneous start trigger for TAUJ1 CH02. 0: Disabled 1: Enabled
1	PIC1BSSER1201	Enables or disables a simultaneous start trigger for TAUJ1 CH01. 0: Disabled 1: Enabled
0	PIC1BSSER1200	Enables or disables a simultaneous start trigger for TAUJ1 CH00. 0: Disabled 1: Enabled

Note 1. This register is not provided for the C1M-A1.

24.2.2.11 PIC1BSSER03 ─ Simultaneous Start Control Register 03

The PIC1BSSER03 register enables a start trigger for each channel of OSTM0 to OSTM2.

Access: Readable/writable in 16-bit units.

Address: <PIC1B0_base> + $1C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PIC1BSSER0302	PIC1BSSER0301	PIC1BSSER0300
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 24.21 PIC1BSSER03 Register Contents

Bit Position	Bit Name	Function
15 to 3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2	PIC1BSSER0302	Enables or disables a simultaneous start trigger for OSTM2. 0: Disabled 1: Enabled
1	PIC1BSSER0301	Enables or disables a simultaneous start trigger for OSTM1. 0: Disabled 1: Enabled
0	PIC1BSSER0300	Enables or disables a simultaneous start trigger for OSTM0. 0: Disabled 1: Enabled

24.2.2.12 PIC1BSSER13 ─ Simultaneous Start Control Register 13*1

The PIC1BSSER13 register enables a start trigger for each channel of OSTM3.

Access: Readable/writable in 16-bit units.

Address: <PIC1B1_base> + $1C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PIC1BSSER1300
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 24.22 PIC1BSSER13 Register Contents

Bit Position	Bit Name	Function
15 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BSSER1300	Enables or disables a simultaneous start trigger for OSTM3. 0: Disabled 1: Enabled

Note 1. This register is not provided for the C1M-A1.

24.2.2.13 PIC1BINIn0 ─ RS Flip-Flop Circuit Initialization Register n0*1

The PIC1BINIn0 register enables initialization of the RS flip-flop circuits 4 to 2 (RSn4 to 2)

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 20_H (n = 0), <PIC1B0_base> + $2C_H$ (n = 1),
<PIC1B1_base> + 20_H (n = 2), <PIC1B1_base> + $2C_H$ (n = 3)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	PIC1BINIn04	PIC1BINIn03	PIC1BINIn02	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	W	W	W	R	R

Table 24.23 PIC1BINIn0 Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 2	PIC1BINIn0[4:2]	Enables or disables initialization of the RS flip-flop circuits 4 to 2 (RSn4 to RSn2) used for the PWM output function with dead time. These bits are always read as 0. 0: Disabled 1: Initialized
1, 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Note 1. TAUD2 (n = 2) and TAUD3 (n = 3) are not provided for the C1M-A1.

24.2.2.14 PIC1BINIn1 — DT Initialization Register n1

The PIC1BINIn1 register enables initialization of the latch & toggle (DT) circuit.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 24_H (n = 0), <PIC1B0_base> + 30_H (n = 1)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PIC1BINIn12	PIC1BINIn11	PIC1BINIn10
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	W	W	W

Table 24.24 PIC1BINIn1 Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2 to 0	PIC1BINIn1[2:0]	Enables or disables initialization of the DT circuit to be used for the trigger and pulse width measurement function. These bits are always read as 0. 0: Disabled 1: Initialized

24.2.2.15 PIC1BLHSEL0 — TSG30 Output Low/High Level Select Register

The PIC1BLHSEL0 register selects low/high level output of the TSG30 output.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 60_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	PIC1BLHSEL 06	PIC1BLHSEL 05	PIC1BLHSEL 04	PIC1BLHSEL 03	PIC1BLHSEL 02	PIC1BLHSEL 01	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 24.25 PIC1BLHSEL0 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	PIC1BLHSEL0m	These bits are applied to TSG30 output [6:1] and PIC1BLHSEL0 [6:1]. 0: Low level output 1: High level output
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

24.2.2.16 PIC1BTSGOUTCTR0 ─ TSG30 Output Control Register

The PIC1BTSGOUTCTR0 register selects the output type of TSG30 output signal.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 64_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	PIC1BSEL06	PIC1BSEL05	PIC1BSEL04	PIC1BSEL03	PIC1BSEL02	PIC1BSEL01	PIC1BSEL00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.26 PIC1BTSGOUTCTR0 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	PIC1BSEL0m	Select the output signal from either TSG30 output or low/high level output. 0: TSG30 output 1: Low/high level output
0	PIC1BSEL00	Switches on/off of the function to output low/high level. 0: Off (only TSG30 output is available) 1: On (TSG30 output and low/high level output can be switched)*1

Note 1. When low/high level output function of TSG30 output is turned on, a delay of one cycle of the clock (CLKC_HSB) is generated from the output when the function is turned off.

24.2.2.17 PIC1BLHSEL1 ─ TSG31 Output Low/High Level Select Register

The PIC1BLHSEL1 register selects low/high level output of the TSG31 output.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 68_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	PIC1BLHSEL 16	PIC1BLHSEL 15	PIC1BLHSEL 14	PIC1BLHSEL 13	PIC1BLHSEL 12	PIC1BLHSEL 11	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 24.27 PIC1BLHSEL1 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	PIC1BLHSEL1m	These bits are applied to TSG31 output [6:1] and PIC1BLHSEL1[6:1]. 0: Low level output 1: High level output
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

24.2.2.18 PIC1BTSGOUTCTR1 ─ TSG31 Output Control Register

The PIC1BTSGOUTCTR1 register selects the output type of TSG31 output.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + $6C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	PIC1BSEL16	PIC1BSEL15	PIC1BSEL14	PIC1BSEL13	PIC1BSEL12	PIC1BSEL11	PIC1BSEL10
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.28 PIC1BTSGOUTCTR1 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	PIC1BSEL1m	Select the output signal between TSG31 output and low/high-level output. 0: TSG31 output 1: Low/high level output
0	PIC1BSEL10	Switches on/off the function to output low/high level. 0: Off (only TSG31 output is available) 1: On (TSG31 output and low/high level output can be switched)*1

Note 1. When low/high level output function of TSG31 output is turned on, a delay of one cycle of the clock (CLKC_HSB) is generated from the output when the function is turned off.

24.2.2.19 PIC1BLHSEL2 ─ TSG32 Output Low/High Level Select Register*1

The PIC1BLHSEL2 register selects the output level (low/high) of the TSG32 output.

Access: Readable/writable in 8-bit units.

Address: <PIC1B1_base> + 60_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	PIC1BLHSEL 26	PIC1BLHSEL 25	PIC1BLHSEL 24	PIC1BLHSEL 23	PIC1BLHSEL 22	PIC1BLHSEL 21	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 24.29 PIC1BLHSEL2 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	PIC1BLHSEL2m	These bits are applied to TSG32 output[6:1] and PIC1BLHSEL2[6:1]. 0: Low level output 1: High level output
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Note 1. C1M-A1 is not provided with the PIC1BLHSEL2 register.

24.2.2.20 PIC1BTSGOUTCTR2 ─ TSG32 Output Control Register*1

The PIC1BTSGOUTCTR2 register selects the type of the signal for output from the TSG32 pin.

Access: Readable/writable in 8-bit units.

Address: <PIC1B1_base> + 64_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	PIC1BSEL26	PIC1BSEL25	PIC1BSEL24	PIC1BSEL23	PIC1BSEL22	PIC1BSEL21	PIC1BSEL20
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.30 PIC1BTSGOUTCTR2 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	PIC1BSEL2m	Select the output signal between TSG32 output and low/high-level output. 0: TSG32 output 1: Low/high level output
0	PIC1BSEL20	Switches on/off of the function to output low/high level. 0: Off (only TSG32 output is available) 1: On (TSG32 output and low/high level output can be switched)*2

Note 1. PIC1BTSGOUTCTR2 is not provided for the C1M-A1.

Note 2. When the low or high level output function of the TSG32 output pins is turned on and then off, operation as a TSG output only starts after a delay of one cycle of the clock (CLKC_HSB).

24.2.2.21 PIC1BTSGHALLSEL ─ Hall Sensor Input Select Register

The PIC1BTSGHALLSEL register sets the pin conditions to input the external hall sensor signal.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 74_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIC1BTSG1 HALLSEL	PIC1BTSG0 HALLSEL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 24.31 PIC1BTSGHALLSEL Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1	PIC1B TSG1HALLSEL	Sets pin condition to input the external hall sensor signal.*1 0: Separate input 1: Alternative input with ENCA
0	PIC1B TSG0HALLSEL	Sets pin condition to input the external hall sensor signal.*1 0: Separate input 1: Alternative input with ENCA

Note 1. For the products of C1M-A series, set this register to 1 because the external hall sensor input pins are also used as ENCA input pins. Set the bit 0 of the PIC1BREG50 register and the PIC1BREG51 register as the following tables.

Table 24.32 Settings of PIC1BTSGHALLSEL and PIC1BREG51 Register Bit 0 (PIC1BREG5100)

PIC1BTSG1HALLSEL	PIC1BREG5100	Function
1	1	Selects input pin ENCA1E0, ENCA1E1, and ENCA1EC.
Other than above		Setting prohibited.

Table 24.33 Settings of PIC1BTSGHALLSEL and PIC1BREG5000 Register Bit 0 (PIC1BREG5000)

PIC1BTSG0HALLSEL	PIC1BREG5000	Function
1	0	Selects input pin ENCA0E0, ENCA0E1, and ENCA0EC.
Other than above		Setting prohibited.

24.2.2.22 PIC1BTAUD0SEL ─ TAUD0 Input Select Register

The PIC1BTAUD0SEL register is a 32-bit register that selects TAUDTIN input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + 78_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PIC1BTAUD0IN143	PIC1BTAUD0IN142	PIC1BTAUD0IN141	PIC1BTAUD0IN140	PIC1BTAUD0IN123	PIC1BTAUD0IN122	PIC1BTAUD0IN121	PIC1BTAUD0IN120	PIC1BTAUD0IN103	PIC1BTAUD0IN102	PIC1BTAUD0IN101	PIC1BTAUD0IN100	PIC1BTAUD0IN83	PIC1BTAUD0IN82	PIC1BTAUD0IN81	PIC1BTAUD0IN80
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1BTAUD0IN63	PIC1BTAUD0IN62	PIC1BTAUD0IN61	PIC1BTAUD0IN60	PIC1BTAUD0IN43	PIC1BTAUD0IN42	PIC1BTAUD0IN41	PIC1BTAUD0IN40	PIC1BTAUD0IN23	PIC1BTAUD0IN22	PIC1BTAUD0IN21	PIC1BTAUD0IN20	PIC1BTAUD0IN03	PIC1BTAUD0IN02	PIC1BTAUD0IN01	PIC1BTAUD0IN00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.34 PIC1BTAUD0SEL Register Contents

Bit Position	Bit Name	Function
2m + 3	PIC1BTAUD0INm3	Selects the signal for output from the TAUD0TIN (m + 1) output pin. 00: TAUD0TIN (m + 1) is selected. 01: TAUD0TIN (m) is selected. 10: TAUD1TIN (m + 1) is selected. 11: TAUD1TIN (m) is selected.
2m + 2	PIC1BTAUD0INm2	
2m + 1	PIC1BTAUD0INm1	Selects the signal for output from the TAUD0TIN (m) output pin. 00: TAUD0TIN (m) is selected. 01: TAUD0TIN (m + 1) is selected. 10: TAUD1TIN (m) is selected. 11: TAUD1TIN (m + 1) is selected.
2m	PIC1BTAUD0INm0	

Note: m is an even channel number of TAUD0 (CHm_even).

24.2.2.23 PIC1BTAUD1SEL ─ TAUD1 Input Select Register

PIC1BTAUD1SEL is a 32-bit register that selects TAUDTIN input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + $7C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PIC1BTAUD1IN143	PIC1BTAUD1IN142	PIC1BTAUD1IN141	PIC1BTAUD1IN140	PIC1BTAUD1IN123	PIC1BTAUD1IN122	PIC1BTAUD1IN121	PIC1BTAUD1IN120	PIC1BTAUD1IN103	PIC1BTAUD1IN102	PIC1BTAUD1IN101	PIC1BTAUD1IN100	PIC1BTAUD1IN83	PIC1BTAUD1IN82	PIC1BTAUD1IN81	PIC1BTAUD1IN80
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1BTAUD1IN63	PIC1BTAUD1IN62	PIC1BTAUD1IN61	PIC1BTAUD1IN60	PIC1BTAUD1IN43	PIC1BTAUD1IN42	PIC1BTAUD1IN41	PIC1BTAUD1IN40	PIC1BTAUD1IN23	PIC1BTAUD1IN22	PIC1BTAUD1IN21	PIC1BTAUD1IN20	PIC1BTAUD1IN03	PIC1BTAUD1IN02	PIC1BTAUD1IN01	PIC1BTAUD1IN00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.35 PIC1BTAUD1SEL Register Contents

Bit Position	Bit Name	Function
2m + 3	PIC1BTAUD1INm3	Selects the signal for output from the TAUD1TIN (m + 1) output pin. 00: TAUD1TIN (m + 1) is selected. 01: TAUD1TIN (m) is selected. 10: TAUD0TIN (m + 1) is selected. 11: TAUD0TIN (m) is selected.
2m + 2	PIC1BTAUD1INm2	
2m + 1	PIC1BTAUD1INm1	Selects the signal for output from the TAUD1TIN (m) output pin. 00: TAUD1TIN (m) is selected. 01: TAUD1TIN (m + 1) is selected. 10: TAUD0TIN (m) is selected. 11: TAUD0TIN (m + 1) is selected.
2m	PIC1BTAUD1INm0	

Note: m is an even channel number of TAUD1 (CHm_even).

24.2.2.24 PIC1BTAUD2SEL ─ TAUD2 Input Select Register*1

PIC1BTAUD2SEL is a 32-bit register that selects TAUDTIN input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B1_base> + 78_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PIC1BTAUD2IN143	PIC1BTAUD2IN142	PIC1BTAUD2IN141	PIC1BTAUD2IN140	PIC1BTAUD2IN123	PIC1BTAUD2IN122	PIC1BTAUD2IN121	PIC1BTAUD2IN120	PIC1BTAUD2IN103	PIC1BTAUD2IN102	PIC1BTAUD2IN101	PIC1BTAUD2IN100	PIC1BTAUD2IN83	PIC1BTAUD2IN82	PIC1BTAUD2IN81	PIC1BTAUD2IN80
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1BTAUD2IN63	PIC1BTAUD2IN62	PIC1BTAUD2IN61	PIC1BTAUD2IN60	PIC1BTAUD2IN43	PIC1BTAUD2IN42	PIC1BTAUD2IN41	PIC1BTAUD2IN40	PIC1BTAUD2IN23	PIC1BTAUD2IN22	PIC1BTAUD2IN21	PIC1BTAUD2IN20	PIC1BTAUD2IN03	PIC1BTAUD2IN02	PIC1BTAUD2IN01	PIC1BTAUD2IN00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.36 PIC1BTAUD2SEL Register Contents

Bit Position	Bit Name	Function
2m + 3	PIC1BTAUD2INm3	Selects the signal for output from the TAUD2TIN (m + 1) output pin. 00: TAUD2TIN (m + 1) is selected. 01: TAUD2TIN (m) is selected. 10: TAUD3TIN (m + 1) is selected. 11: TAUD3TIN (m) is selected. Settings other than above are prohibited.
2m + 2	PIC1BTAUD2INm2	
2m + 1	PIC1BTAUD2INm1	Selects the signal for output from the TAUD2TIN (m) output pin. 00: TAUD2TIN (m) is selected. 01: TAUD2TIN (m + 1) is selected. 10: TAUD3TIN (m) is selected. 11: TAUD3TIN (m + 1) is selected. Settings other than above are prohibited.
2m	PIC1BTAUD2INm0	

Note: m is an even channel number of TAUD2 (CHm_even).

Note 1. This register is not provided for the C1M-A1.

24.2.2.25 PIC1BTAUD3SEL ─ TAUD3 Input Select Register*1

PIC1BTAUD3SEL is a 32-bit register that selects TAUDTIN input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B1_base> + $7C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PIC1BTAUD3IN143	PIC1BTAUD3IN142	PIC1BTAUD3IN141	PIC1BTAUD3IN140	PIC1BTAUD3IN123	PIC1BTAUD3IN122	PIC1BTAUD3IN121	PIC1BTAUD3IN120	PIC1BTAUD3IN103	PIC1BTAUD3IN102	PIC1BTAUD3IN101	PIC1BTAUD3IN100	PIC1BTAUD3IN83	PIC1BTAUD3IN82	PIC1BTAUD3IN81	PIC1BTAUD3IN80
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1BTAUD3IN63	PIC1BTAUD3IN62	PIC1BTAUD3IN61	PIC1BTAUD3IN60	PIC1BTAUD3IN43	PIC1BTAUD3IN42	PIC1BTAUD3IN41	PIC1BTAUD3IN40	PIC1BTAUD3IN23	PIC1BTAUD3IN22	PIC1BTAUD3IN21	PIC1BTAUD3IN20	PIC1BTAUD3IN03	PIC1BTAUD3IN02	PIC1BTAUD3IN01	PIC1BTAUD3IN00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.37 PIC1BTAUD3SEL Register Contents

Bit Position	Bit Name	Function
2m + 3	PIC1BTAUD3INm3	Selects the signal for output from the TAUD3TIN (m + 1) output pin. 00: TAUD3TIN (m + 1) is selected. 01: TAUD3TIN (m) is selected. 10: TAUD2TIN (m + 1) is selected. 11: TAUD2TIN (m) is selected.
2m + 2	PIC1BTAUD3INm2	
2m + 1	PIC1BTAUD3INm1	Selects the signal for output from the TAUD3TIN (m) output pin. 00: TAUD3TIN (m) is selected. 01: TAUD3TIN (m + 1) is selected. 10: TAUD2TIN (m) is selected. 11: TAUD2TIN (m + 1) is selected.
2m	PIC1BTAUD3INm0	

Note: m is an even channel number of TAUD3 (CHm_even).

Note 1. This register is not provided for the C1M-A1.

24.2.2.26 PIC1BHIZCEN00 ─ Hi-Z Control Register 00

The PIC1BHIZCEN00 register selects Hi-Z control input signals of TAUD0.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 80_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1BHIZCEN005	—	—	—	—	PIC1BHIZCEN000
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R/W

Table 24.38 PIC1BHIZCEN00 Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When writing, write the value after reset.
5	PIC1BHIZCEN005	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BHIZCEN000	Enables or disables Hi-Z control by ESO0 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting U/V/W outputs or UB/VB/WB outputs of TAUD0.

Set TAPA0CTL0.TAPA0DCN = 0 and TAPA0CTL0.TAPA0DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

24.2.2.27 PIC1BHIZCEN01 ─ Hi-Z Control Register 01

The PIC1BHIZCEN01 register selects Hi-Z control input signals of TAUD1.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 84_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1BHIZCEN015	—	—	—	—	PIC1BHIZCEN010
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R/W

Table 24.39 PIC1BHIZCEN01 Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When writing, write the value after reset.
5	PIC1BHIZCEN015	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BHIZCEN010	Enables or disables Hi-Z control by ESO1 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting U/V/W outputs or UB/VB/WB outputs of TAUD1.

Set TAPA1CTL0.TAPA1DCN = 0 and TAPA1CTL0.TAPA1DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

24.2.2.28 PIC1BHIZCEN02 ─ Hi-Z Control Register 02

The PIC1BHIZCEN02 register selects Hi-Z control input signals of TSG30.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + 88_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1BHIZCEN025	—	PIC1BHIZCEN023	—	—	PIC1BHIZCEN020
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R/W	R	R	R/W

Table 24.40 PIC1BHIZCEN2 Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When writing, write the value after reset.
5	PIC1BHIZCEN025	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	PIC1BHIZCEN023	Enables or disables Hi-Z control by INTTSG30IER interrupt request signal. 0: Disabled 1: Enabled
2, 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BHIZCEN020	Enables or disables Hi-Z control by ESO3 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting TSG30 output.

Set TAPA3CTL0.TAPA3DCN = 0 and TAPA3CTL0.TAPA3DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

24.2.2.29 PIC1BHIZCEN03 ─ Hi-Z Control Register 03

The PIC1BHIZCEN03 register selects Hi-Z control input signals of TSG31.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + $8C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1BHIZCEN035	PIC1BHIZCEN034	—	—	—	PIC1BHIZCEN030
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W

Table 24.41 PIC1BHIZCEN3 Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When writing, write the value after reset.
5	PIC1BHIZCEN035	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4	PIC1BHIZCEN034	Enables or disables Hi-Z control by INTTSG31IER interrupt request signal. 0: Disabled 1: Enabled
3 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BHIZCEN030	Enables or disables Hi-Z control by ESO4 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting TSG31 output.

Set TAPA4CTL0.TAPA4DCN = 0 and TAPA4CTL0.TAPA4DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

24.2.2.30 PIC1BHIZCEN10 ─ Hi-Z Control Register 10*1

The PIC1BHIZCEN10 register selects Hi-Z control input signals of TAUD2.

Access: Readable/writable in 8-bit units.

Address: <PIC1B1_base> + 80_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1BHIZCEN 105	—	—	—	—	PIC1B1HIZCEN 100
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R/W

Table 24.42 PIC1BHIZCEN10 Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When writing, write the value after reset.
5	PIC1BHIZCEN 105	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BHIZCEN 100	Enables or disables Hi-Z control by ESO2 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting U/V/W outputs or UB/VB/WB outputs of TAUD2.

Set TAPA2CTL0.TAPA2DCN = 0 and TAPA2CTL0.TAPA2DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

Note 1. This register is not provided for the C1M-A1.

24.2.2.31 PIC1BHIZCEN12 ─ Hi-Z Control Register 12*1

The PIC1BHIZCEN12 register selects Hi-Z control input signals of TSG32.

Access: Readable/writable in 8-bit units.

Address: <PIC1B1_base> + 88_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1BHIZCEN 125	—	PIC1BHIZCEN 123	—	—	PIC1BHIZCEN 120
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R/W	R	R	R/W

Table 24.43 PIC1BHIZCEN12 Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When writing, write the value after reset.
5	PIC1BHIZCEN 125	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	PIC1BHIZCEN 123	Enables or disables Hi-Z control by INTTSG32IER interrupt request signal. 0: Disabled 1: Enabled
2, 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BHIZCEN 120	Enables or disables Hi-Z control by ESO5 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting TSG32 output.

Set TAPA5CTL0.TAPA5DCN = 0 and TAPA5CTL0.TAPA5DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

Note 1. This register is not provided for the C1M-A1.

24.2.2.32 PIC1BENCSEL400 ─ ENCATIN1 Input Select Register 400

The PIC1BENCSEL400 register is used for encoder capture trigger function.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + $B8_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	PIC1BENCSEL 4007	—	—	—	PIC1BENCSEL 4003	PIC1BENCSEL 4002	PIC1BENCSEL 4001	PIC1BENCSEL 4000
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Table 24.44 PIC1BENCSEL400 Register Contents

Bit Position	Bit Name	Function
7	PIC1BENCSEL 4007	Enables or disables output of INTTAUD0Im signal selected by PIC1BENCSEL400[3:0]. 0: Disabled 1: Enabled
6 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	PIC1BENCSEL 400[3:0]	Select TAUD0TINTm to be used as a capture trigger signal for ENCA0 and ENCA1. 0: INTTAUD0I0 is selected. 1: INTTAUD0I1 is selected. 2: INTTAUD0I2 is selected. 3: INTTAUD0I3 is selected. 4: INTTAUD0I4 is selected. 5: INTTAUD0I5 is selected. 6: INTTAUD0I6 is selected. 7: INTTAUD0I7 is selected. 8: INTTAUD0I8 is selected. 9: INTTAUD0I9 is selected. 10: INTTAUD0I10 is selected. 11: INTTAUD0I11 is selected. 12: INTTAUD0I12 is selected. 13: INTTAUD0I13 is selected. 14: INTTAUD0I14 is selected. 15: INTTAUD0I15 is selected.

24.2.2.33 PIC1BENCSEL410 ─ ENCATIN1 Input Select Register 410

The PIC1BENCSEL410 register is used for encoder capture trigger function.

Access: Readable/writable in 8-bit units.

Address: <PIC1B0_base> + BC_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	PIC1BENCSEL 4107	—	—	—	PIC1BENCSEL 4103	PIC1BENCSEL 4102	PIC1BENCSEL 4101	PIC1BENCSEL 4100
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Table 24.45 PIC1BENCSEL410 Register Contents

Bit Position	Bit Name	Function
7	PIC1BENCSEL 4107	Enables or disables output of INTTAUD1Im signal selected by PIC1BENCSEL410[3:0]. 0: Disabled 1: Enabled
6 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	PIC1BENCSEL 410[3:0]	Selects TAUD1TINTm to be used as a capture trigger signal for ENCA0 and ENCA1. 0: INTTAUD1I0 is selected. 1: INTTAUD1I1 is selected. 2: INTTAUD1I2 is selected. 3: INTTAUD1I3 is selected. 4: INTTAUD1I4 is selected. 5: INTTAUD1I5 is selected. 6: INTTAUD1I6 is selected. 7: INTTAUD1I7 is selected. 8: INTTAUD1I8 is selected. 9: INTTAUD1I9 is selected. 10: INTTAUD1I10 is selected. 11: INTTAUD1I11 is selected. 12: INTTAUD1I12 is selected. 13: INTTAUD1I13 is selected. 14: INTTAUD1I14 is selected. 15: INTTAUD1I15 is selected.

24.2.2.34 PIC1BREG200 ─ Timer Input/Output Control Register 200

The PIC1BREG200 register selects TAUD0 input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + $C0_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PIC1BREG20025	PIC1BREG20024	—	—	—	—	—	PIC1BREG20018	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PIC1BREG20011	PIC1BREG20010	PIC1BREG20009	PIC1BREG20008	—	—	—	—	PIC1BREG20003	PIC1BREG20002	PIC1BREG20001	PIC1BREG20000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 24.46 PIC1BREG200 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	Reserved	When read, the value after reset is read. When writing, write the value after reset.
25, 24	PIC1BREG20025, PIC1BREG20024	Select TAUD channel to use for TAPA0TSIM0 and TAPA0TUDCM0. 00: No channel selected. 01: TAUD0 channel 0 is selected. 10: TAUD0 channel 2 is selected. 11: TAUD0 channel 8 is selected.
23 to 19	Reserved	When read, the value after reset is read. When writing, write the value after reset.
18	PIC1BREG20018	Selects the signal for input as TAUD0TIN10, TAUD0TIN12, and TAUD0TIN14 signals of TAUD0. 1: TOUT from TAUD0 CH02 Settings other than above are prohibited.*1
17 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	PIC1BREG20011, PIC1BREG20010	Select the signal for input as TAUD0TIN6 and TAUD0TIN7 signals of TAUD0. 10: TS0PTE signal of TSG30. Settings other than above are prohibited.*1
9, 8	PIC1BREG20009, PIC1BREG20008	Select the signal for input as TAUD0TIN4 and TAUD0TIN5 signals of TAUD0. 10: TS0PTE signal of TSG30. Settings other than above are prohibited.*1
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	PIC1BREG20003	Selects the signal for input as TAUD0TIN7 signal of TAUD0. 0: TIN pin input 1: Input signal selected by PIC1BREG20011 and PIC1BREG20010 bits (TS0PTE signal).
2	PIC1BREG20002	Selects the signal for input as TAUD0TIN6 signal of TAUD0. 0: TIN pin input 1: Input signal selected by PIC1BREG20011 and PIC1BREG20010 bits (TS0PTE signal).

Table 24.46 PIC1BREG200 Register Contents (2/2)

Bit Position	Bit Name	Function
1	PIC1BREG20001	Selects the signal for input as TAUD0TIN5 signal of TAUD0. 0: TIN pin input 1: Input signal selected by PIC1BREG20009 and PIC1BREG20008 bits (TS0PTE signal).
0	PIC1BREG20000	Selects the signal for input as TAUD0TIN4 signal of TAUD0. 0: TIN pin input 1: Input signal selected by PIC1BREG20009 and PIC1BREG20008 bits (TS0PTE signal).

Note 1. Set any appropriate value because the value after reset is “setting prohibited”.

24.2.2.35 PIC1BREG210 ─ Timer Input/Output Control Register 210

The PIC1BREG210 register selects TAUD1 input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + $D4_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PIC1BREG21025	PIC1BREG21024	—	—	—	—	—	PIC1BREG21018	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PIC1BREG21011	PIC1BREG21010	PIC1BREG21009	PIC1BREG21008	—	—	—	—	PIC1BREG21003	PIC1BREG21002	PIC1BREG21001	PIC1BREG21000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 24.47 PIC1BREG210 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	Reserved	When read, the value after reset is read. When writing, write the value after reset.
25, 24	PIC1BREG21025, PIC1BREG21024	Select the TAUD channel to use for TAPA1TSIM0 and TAPA1TUDCM0. 00: No channel selected. 01: TAUD1 channel 0 is selected. 10: TAUD1 channel 2 is selected. 11: TAUD1 channel 8 is selected.
23 to 19	Reserved	When read, the value after reset is read. When writing, write the value after reset.
18	PIC1BREG21018	Selects the signal for input as TAUD1TIN10, TAUD1TIN12, and TAUD1TIN14 signals of TAUD1. 1: TOUT from TAUD1 CH02 Settings other than above are prohibited.*1
17 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	PIC1BREG21011, PIC1BREG21010	Select the signal for input as TAUD1TIN6 and TAUD1TIN7 signals of TAUD1. 10: TS0PTE signal of TSG31 Settings other than above are prohibited.*1
9, 8	PIC1BREG21009, PIC1BREG21008	Select the signal for input as TAUD1TIN4 and TAUD1TIN5 signals of TAUD1. 10: TS0PTE signal of TSG31 Settings other than above are prohibited.*1
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	PIC1BREG21003	Selects the signal for input as TAUD1TIN7 signal of TAUD1. 0: TIN pin input 1: Input signal selected by PIC1BREG21011 and PIC1BREG21010 bits (TS0PTE signal).
2	PIC1BREG21002	Selects the signal for input as TAUD1TIN6 signal of TAUD1. 0: TIN pin input 1: Input signal selected by PIC1BREG21011 and PIC1BREG21010 bits (TS0PTE signal).

Table 24.47 PIC1BREG210 Register Contents (2/2)

Bit Position	Bit Name	Function
1	PIC1BREG21001	Selects the signal for input as TAUD1TIN5 signal of TAUD1. 0: TIN pin input 1: Input signal selected by PIC1BREG21009 and PIC1BREG21008 bits (TS0PTE signal).
0	PIC1BREG21000	Selects the signal for input as TAUD1TIN4 signal of TAUD1. 0: TIN pin input 1: Input signal selected by PIC1BREG21009 and PIC1BREG21008 bits (TS0PTE signal).

Note 1. Set any appropriate value because the value after reset is “setting prohibited”.

24.2.2.36 PIC1BREG220 ─ Timer Input/Output Control Register*1

The PIC1BREG220 register selects TAUD2 input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B1_base> + $C0_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PIC1BREG22025	PIC1BREG22024	—	—	—	—	—	PIC1BREG22018	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PIC1BREG22011	PIC1BREG22010	PIC1BREG22009	PIC1BREG22008	—	—	—	—	PIC1BREG22003	PIC1BREG22002	PIC1BREG22001	PIC1BREG22000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 24.48 PIC1BREG220 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	Reserved	When read, the value after reset is read. When writing, write the value after reset.
25, 24	PIC1BREG22025, PIC1BREG22024	Select TAUD channel to use for TAPA2TSIM0 and TAPA2TUDCM0. 00: No channel selected. 01: TAUD2 channel 0 is selected. 10: TAUD2 channel 2 is selected. 11: TAUD2 channel 8 is selected.
23 to 19	Reserved	When read, the value after reset is read. When writing, write the value after reset.
18	PIC1BREG22018	Selects the signal for input as TAUD2TIN10, TAUD2TIN12, and TAUD2TIN14 signals of TAUD2. 1: TOUT from TAUD2 CH02 Settings other than above are prohibited.*2
17 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	PIC1BREG22011, PIC1BREG22010	Selects the signal for input as TAUD2TIN6 and TAUD2TIN7 signals of TAUD2. 10: TS0PTE signal of TSG32 Settings other than above are prohibited.*2
9, 8	PIC1BREG22009, PIC1BREG22008	Selects the signal for input as TAUD2TIN4 and TAUD2TIN5 signals of TAUD2. 10: TS0PTE signal of TSG32 Settings other than above are prohibited.*2
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	PIC1BREG22003	Selects the signal for input as TAUD2TIN7 signal of TAUD2. 0: TIN pin input 1: Input signal selected by PIC1BREG22011 and PIC1BREG22010 bits (TS0PTE signal).

Table 24.48 PIC1BREG220 Register Contents (2/2)

Bit Position	Bit Name	Function
2	PIC1BREG22002	Selects the signal for input as TAUD2TIN6 signal of TAUD2. 0: TIN pin input 1: Input signal selected by PIC1BREG22011 and PIC1BREG22010 bits (TS0PTE signal).
1	PIC1BREG22001	Selects the signal for input as TAUD2TIN5 signal of TAUD2. 0: TIN pin input 1: Input signal selected by PIC1BREG22009 and PIC1BREG22008 bits (TS0PTE signal).
0	PIC1BREG22000	Selects the signal for input as TAUD2TIN4 signal of TAUD2. 0: TIN pin input 1: Input signal selected by PIC1BREG22009 and PIC1BREG22008 bits (TS0PTE signal).

Note 1. This register is not provided for the C1M-A1.

Note 2. Set any appropriate value because the value after reset is “setting prohibited”.

24.2.2.37 PIC1BREG230 ─ Timer Input/Output Control Register 230*1

The PIC1BREG230 register selects TAUD3 input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B1_base> + $D4_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	PIC1BREG23018	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PIC1BREG23003	PIC1BREG23002	PIC1BREG23001	PIC1BREG23000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 24.49 PIC1BREG230 Register Contents

Bit Position	Bit Name	Function
31 to 19	Reserved	When read, the value after reset is read. When writing, write the value after reset.
18	PIC1BREG23018	Selects the signal for input as TAUD3TIN10, TAUD3TIN12, and TAUD3TIN14 signals of TAUD3. 1: TOUT from TAUD3 CH02 Settings other than above are prohibited.*2
17 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	PIC1BREG23003	Selects the signal for input as TAUD3TIN7 signal of TAUD3. 0: TIN pin input Settings other than above are prohibited.
2	PIC1BREG23002	Selects the signal for input as TAUD3TIN6 signal of TAUD3. 0: TIN pin input Settings other than above are prohibited.
1	PIC1BREG23001	Selects the signal for input as TAUD3TIN5 signal of TAUD3. 0: TIN pin input Settings other than above are prohibited.
0	PIC1BREG23000	Selects the signal for input as TAUD3TIN4 signal of TAUD3. 0: TIN pin input Settings other than above are prohibited.

Note 1. This register is not provided for the C1M-A1.

Note 2. Set any appropriate value because the value after reset is “setting prohibited”.

24.2.2.38 PIC1BREG2n1 ─ Timer Input/Output Control Register 2n1*1

The PICBREG2n1 register selects logical operation for the combination circuit PFN0xx.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + $C4_H$ (n = 0), <PIC1B0_base> + $D8_H$ (n = 1),
<PIC1B1_base> + $C4_H$ (n = 2), <PIC1B1_base> + $D8_H$ (n = 3)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PIC1BREG2n127	PIC1BREG2n126	PIC1BREG2n125	PIC1BREG2n124	PIC1BREG2n123	PIC1BREG2n122	PIC1BREG2n121	PIC1BREG2n120	PIC1BREG2n119	PIC1BREG2n118	PIC1BREG2n117	PIC1BREG2n116
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 24.50 PIC1BREG2n1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, the value after reset is read. When writing, write the value after reset.
27, 26	PIC1BREG2n127, PIC1BREG2n126	Select PFN045 WO2 output.*2 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*3
25, 24	PIC1BREG2n125, PIC1BREG2n124	Select PFN045 WO1 output.*2 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*3
23, 22	PIC1BREG2n123, PIC1BREG2n122	Select PFC023 VO2 output.*2 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*3
21, 20	PIC1BREG2n121, PIC1BREG2n120	Select PFC023 VO1 output.*2 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*3
19, 18	PIC1BREG2n119, PIC1BREG2n118	Select PFN001 UO2 output.*2 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*3
17, 16	PIC1BREG2n117, PIC1BREG2n116	Select PFN001 UO1 output.*2 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*3

Table 24.50 PIC1BREG2n1 Register Contents (2/2)

Bit Position	Bit Name	Function
15 to 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Note 1. TAUD2 (n = 2) and TAUD3 (n = 3) are not provided for the C1M-A1.

Note 2. The register value for some functions needs to be set depending on the value of TAUD. For the setting values, see **Section 24.2.3, Function**.

Note 3. Set any appropriate value because the value after reset is “setting prohibited”.

Block diagram of PFN001 is shown in the following figure.

PFN023 and PFN045 are operated under the same logic with different input signals and registers.

For connection of PFN0xx with peripheral circuits, see **Figure 24.15**.

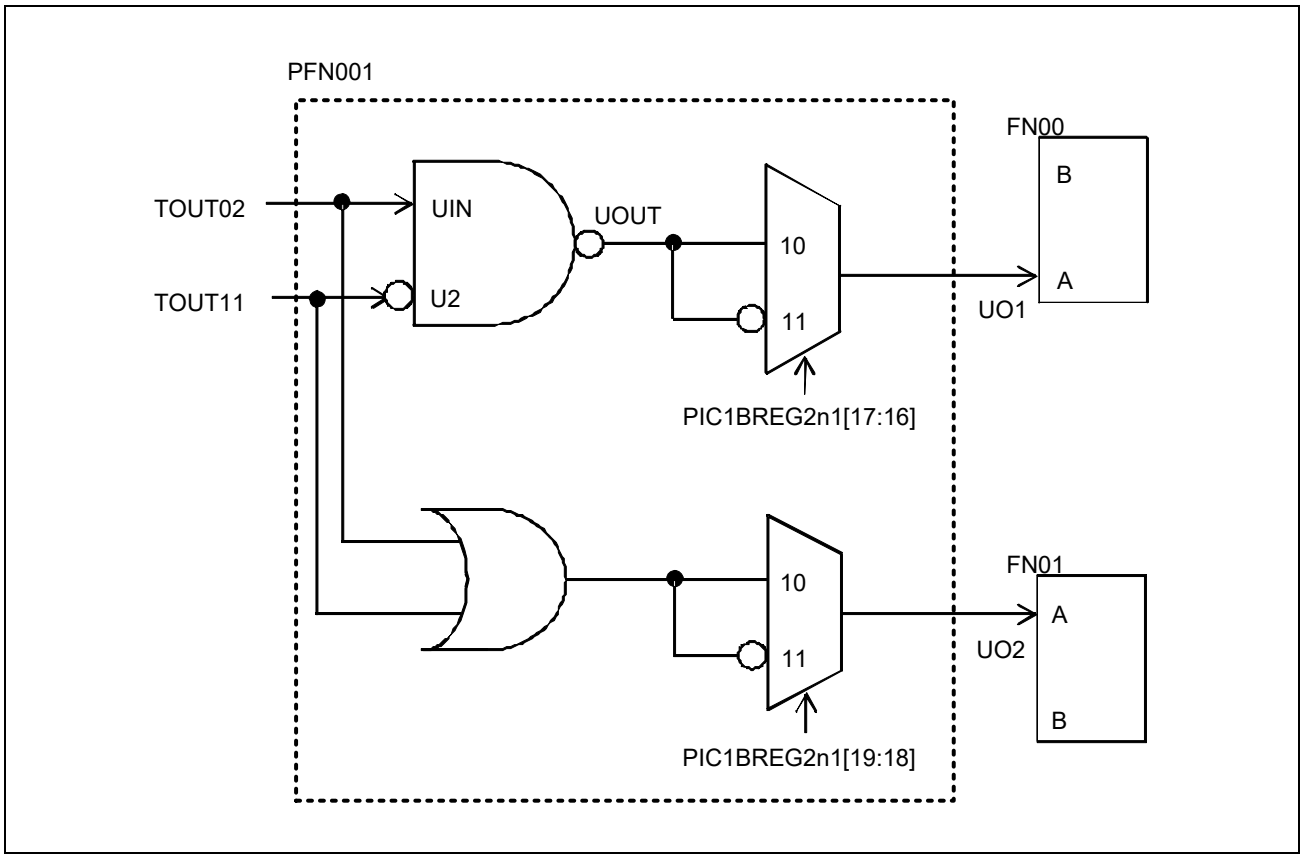


Figure 24.1 Block Diagram of PFN001

24.2.2.39 PIC1BREG2n2 ─ Timer Input/Output Control Register 2n2*1

The PIC1BREG2n2 register selects input signals of TAUDn CHm.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + $C8_H$ (n = 0), <PIC1B0_base> + DC_H (n = 1),
<PIC1B1_base> + $C8_H$ (n = 2), <PIC1B1_base> + DC_H (n = 3)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PIC1BREG2n227	PIC1BREG2n226	PIC1BREG2n225	PIC1BREG2n224	PIC1BREG2n223	PIC1BREG2n222	PIC1BREG2n221	PIC1BREG2n220	PIC1BREG2n219	PIC1BREG2n218	PIC1BREG2n217	PIC1BREG2n216
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PIC1BREG2n204	PIC1BREG2n203	PIC1BREG2n202	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Table 24.51 PIC1BREG2n2 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, the value after reset is read. When writing, write the value after reset.
27, 26	PIC1BREG2n227, PIC1BREG2n226	Selects TIN input signal of TAUDn CH15. 00: TIN pin input. 10: Signal selected by PIC1BREG2n204 bit (TOUT of TAUDn CH09). Settings other than above are prohibited.
25, 24	PIC1BREG2n225, PIC1BREG2n224	Selects TIN input signal of TAUDn CH14. 00: TIN pin input. 10: Signal selected by PIC1BREG2n018 bit (TOUT of TAUDn CH02). Settings other than above are prohibited.
23, 22	PIC1BREG2n223, PIC1BREG2n222	Selects TIN input signal of TAUDn CH13. 00: TIN pin input. 10: Signal selected by PIC1BREG2n203 bit (TOUT of TAUDn CH07). Settings other than above are prohibited.
21, 20	PIC1BREG2n221, PIC1BREG2n220	Select TIN input signal of TAUDn CH12. 00: TIN pin input. 10: Signal selected by PIC1BREG2n018 bit (TOUT of TAUDn CH02). Settings other than above are prohibited.
19, 18	PIC1BREG2n219, PIC1BREG2n218	Select TIN input signal of TAUDn CH11. 00: TIN pin input. 10: Signal selected by PIC1BREG2n202 bit (TOUT of TAUDn CH05). Settings other than above are prohibited.
17, 16	PIC1BREG2n217, PIC1BREG2n216	Select TIN input signal of TAUDn CH10. 00: TIN pin input. 10: Signal selected by PIC1BREG2n018 bit (TOUT of TAUDn CH02). Settings other than above are prohibited.
15 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Table 24.51 PIC1BREG2n2 Register Contents (2/2)

Bit Position	Bit Name	Function
4	PIC1BREG2n204	Selects the signal to be supplied to TIN of TAUDn CH15. 0: TOUT of TAUDnCH09. 1: Set or clear output by TAUDnINT08 and TAUDnINT09.
3	PIC1BREG2n203	Selects the signal to be supplied to TIN of TAUDn CH13. 0: TOUT of TAUDnCH07. 1: Set or clear output by TAUDnINT06 and TAUDnINT07.
2	PIC1BREG2n202	Selects the signal to be supplied to TIN of TAUDn CH11. 0: TOUT of TAUDnCH05. 1: Set or clear output by TAUDnINT04 and TAUDnINT05.
1, 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Note 1. TAUD2 (n = 2) and TAUD3 (n = 3) are not provided for the C1M-A1.

24.2.2.40 PIC1BREG2n3 ─ Timer Input/Output Control Register 2n3*1

The PIC1BREG2n3 register selects logical operation for the combination circuit FN0i.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + CC_H (n = 0), <PIC1B0_base> + $E0_H$ (n = 1),
<PIC1B1_base> + CC_H (n = 2), <PIC1B0_base> + $E0_H$ (n = 3)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PIC1BREG2n322	PIC1BREG2n321	PIC1BREG2n320	—	PIC1BREG2n318	PIC1BREG2n317	PIC1BREG2n316
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PIC1BREG2n314	PIC1BREG2n313	PIC1BREG2n312	—	PIC1BREG2n310	PIC1BREG2n309	PIC1BREG2n308	—	PIC1BREG2n306	PIC1BREG2n305	PIC1BREG2n304	—	PIC1BREG2n302	PIC1BREG2n301	PIC1BREG2n300
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 24.52 PIC1BREG2n3 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is read. When writing, write the value after reset.
22 to 20	PIC1BREG2n322, PIC1BREG2n321, PIC1BREG2n320	Select a logical operation to be performed on input signals A and B. 000: A 100: A and B 101: A or B Settings other than above are prohibited.
19	Reserved	When read, the value after reset is read. When writing, write the value after reset.
18 to 16	PIC1BREG2n318, PIC1BREG2n317, PIC1BREG2n316	Select a logical operation to be performed on input signals A and B. The register value for some functions needs to be set depending on the value of TAUD. 000: A 100: A and B 101: A or B Settings other than above are prohibited.
15	Reserved	When read, the value after reset is read. When writing, write the value after reset.
14 to 12	PIC1BREG2n314, PIC1BREG2n313, PIC1BREG2n312	Select a logical operation to be performed on input signals A and B. The register value for some functions needs to be set depending on the value of TAUD. 000: A 100: A and B 101: A or B Settings other than above are prohibited.
11	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Table 24.52 PIC1BREG2n3 Register Contents (2/2)

Bit Position	Bit Name	Function
10 to 8	PIC1BREG2n310, PIC1BREG2n309, PIC1BREG2n308	Select a logical operation to be performed on input signals A and B. The register value for some functions needs to be set depending on the value of TAUD. 000: A 100: A and B 101: A or B Settings other than above are prohibited.
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 4	PIC1BREG2n306, PIC1BREG2n305, PIC1BREG2n304	Select a logical operation to be performed on input signals A and B. The register value for some functions needs to be set depending on the value of TAUD. 000: A 100: A and B 101: A or B Settings other than above are prohibited.
3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2 to 0	PIC1BREG2n302, PIC1BREG2n301, PIC1BREG2n300	Select a logical operation to be performed on input signals A and B.*2 000: A 100: A and B 101: A or B Settings other than above are prohibited.

Note 1. TAUD2 (n = 2) and TAUD3 (n = 3) are not provided for the C1M-A1.

Note 2. The register value for some functions needs to be set depending on the value of TAUD. For the setting values, see **Section 24.2.3, Function**.

Block diagram of FN00 is shown in the following figure.

FN01 to FN05 are operated under the same logical with different input signals and registers.

For connection of FN0i with peripheral circuits, see **Figure 24.15**.

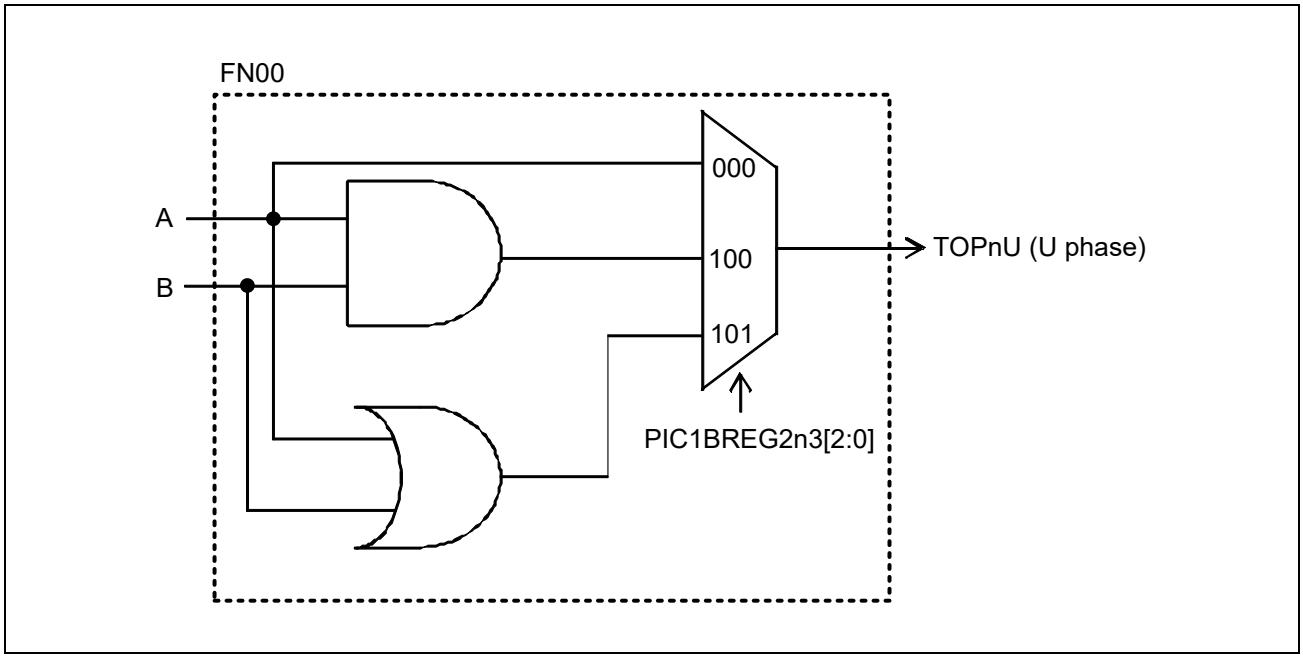


Figure 24.2 Block Diagram of FN00

24.2.2.41 PIC1BREG30 ─ Timer Input/Output Control Register 30

PIC1BREG30 selects ENCAn input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + $E8_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PIC1BREG3022	PIC1BREG3021	PIC1BREG3020	PIC1BREG3019	PIC1BREG3018	PIC1BREG3017	PIC1BREG3016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1BREG3015	PIC1BREG3014	PIC1BREG3013	PIC1BREG3012	PIC1BREG3011	PIC1BREG3010	PIC1BREG3009	PIC1BREG3008	PIC1BREG3007	PIC1BREG3006	PIC1BREG3005	PIC1BREG3004	PIC1BREG3003	PIC1BREG3002	PIC1BREG3001	PIC1BREG3000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.53 PIC1BREG30 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is read. When writing, write the value after reset.
22	PIC1BREG3022	Selects the input signals (to the ENCA0E0, ENCA0E1, ENCA0EC internal input pins) for the ENCA0 timer. 0: The signals selected by the PIC1BREG3000 bit (ENCA0E0 internal input), the PIC1BREG3001 bit (ENCA0E1 internal input), and the PIC1BREG3017 and PIC1BREG3016 bits (ENCA0EC internal input) 1: The signals selected by the PIC1BREG3020 and PIC1BREG3019 bits.
21	PIC1BREG3021	Selects the signals to be supplied to bits 12 to 14 of the PIC1BREG30. 0: ENCA1I1 (signal 1 of the ENCA1 external pin) 1: Signal selected with the PIC1BENCSEL4107 bit of the PIC1BENCSEL410 register.
20, 19	PIC1BREG3020, PIC1BREG3019	Selects the input signals (to the ENCA1E0, ENCA1E1, ENCA1EC internal input pins) for the ENCA1 timer. 00: ENCA1E0, ENCA1E1, ENCA1EC input pins (for the ENCA1 timer). 01: RDC1BOUT, RDC1AOUT, RDC1ZOUT input pins (for RDC3A1).*1 10: RDC0BOUT, RDC0AOUT, RDC0ZOUT input pins (for RDC3A0). Settings other than above are prohibited.
18	PIC1BREG3018	Selects the signal to be supplied to bits 2 to 4 of PIC1BREG30. 0: ENCA0I1 (signal 1 of the ENCA0 external pin) 1: The signal selected by the PIC1BENCSEL4007 bit of the PIC1BENCSEL400 register.
17, 16	PIC1BREG3017, PIC1BREG3016	Selects the input signals (to the ENCA0E0, ENCA0E1, ENCA0EC internal input pins) for the ENCA0 timer. 00: ENCA0E0, ENCA0E1, ENCA0EC input pins (for the ENCA0 timer). 01: RDC0BOUT, RDC0AOUT, RDC0ZOUT input pins (for RDC3A0). 10: RDC1BOUT, RDC1AOUT, RDC1ZOUT input pins (for RDC3A1).*1 Settings other than above are prohibited.

Table 24.53 PIC1BREG30 Register Contents (2/2)

Bit Position	Bit Name	Function
15 to 12	PIC1BREG30[15:12]	Select the signal for input as the ENCAT1TIN1 signal. 0: The signal selected by PIC1BREG3021. 1: The signal selected by PIC1BREG3018. 2: ADCC0TRG4 3: ADCC0TRG3 4: ADCC0TRG2 5: ADCC0TRG1 6: ADCC0TRG0 7: ADCC1TRG4 8: ADCC1TRG3 9: ADCC1TRG2 10: ADCC1TRG1 11: ADCC1TRG0 Settings other than above are prohibited.
11, 10	PIC1BREG3011, PIC1BREG3010	Select the ENCA1EC pin input of timer ENCA1. 00: The signal selected by the PIC1BREG3019 and PIC1BREG3020 bits. 10: The signal selected by the PIC1BREG3016 and PIC1BREG3017 bits. 11: ENCA0EQ1 signal (ENCA0 timer) Settings other than above are prohibited.
9, 8	PIC1BREG3009, PIC1BREG3008	Select the ENCA1E1 pin input of timer ENCA1. 00: Signal selected by the PIC1BREG3019 and PIC1BREG3020 bits. 01: The signal selected by the PIC1BREG3016 and PIC1BREG3017 bits. 10: TS1PUD signal of TSG31. Settings other than above are prohibited.
7, 6	PIC1BREG3007, PIC1BREG3006	Select the ENCA1E0 pin input of timer ENCA1. 00: The signal selected by the PIC1BREG3019 and PIC1BREG3020 bits. 01: The signal selected by the PIC1BREG3016 and PIC1BREG3017 bits. 10: TS1PEC signal of TSG31. Settings other than above are prohibited.
5 to 2	PIC1BREG30[05:02]	Select the signal for input as the ENCAT0TIN1 signal. 0: The signal selected by the PIC1BREG3018 bit. 1: The signal selected by the PIC1BREG3021 bit. 2: ADCC0TRG4 3: ADCC0TRG3 4: ADCC0TRG2 5: ADCC0TRG1 6: ADCC0TRG0 7: ADCC1TRG4 8: ADCC1TRG3 9: ADCC1TRG2 10: ADCC1TRG1 11: ADCC1TRG0 Settings other than above are prohibited.
1	PIC1BREG3001	Selects the signal to input to the ENCA0E1 internal signal. 0: Signal selected by PIC1BREG3017 and PIC1BREG3016. 1: TS0PUD signal of TSG30.
0	PIC1BREG3000	Selects the signal to input to the ENCA0E0 internal signal. 0: Signal selected by PIC1BREG3017 and PIC1BREG3016. 1: TS0PEC signal of TSG30.

Note 1. This setting is prohibited for the C1M-A1.

24.2.2.42 PIC1BREG31 ─ Timer Input/Output Control Register 31

The PIC1BREG31 register selects TAUD0, TAUD1, and TAUJ0 input signals.

Access: Readable/writable in 32-bit units.

Address: <PIC1B0_base> + EC_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PIC1BREG3122	PIC1BREG3121	PIC1BREG3120	PIC1BREG3119	PIC1BREG3118	PIC1BREG3117	PIC1BREG3116
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1BREG3115	—	PIC1BREG3113	PIC1BREG3112	PIC1BREG3111	PIC1BREG3110	PIC1BREG3109	PIC1BREG3108	PIC1BREG3107	PIC1BREG3106	—	PIC1BREG3104	PIC1BREG3103	—	PIC1BREG3101	PIC1BREG3100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W	R/W

Table 24.54 PIC1BREG31 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is read. When writing, write the value after reset.
22, 21	PIC1BREG3122, PIC1BREG3121	Selects TIN input signal of TAUD1 CH02. 00: TIN pin input. 01: DT output signal of ENCAT1EQ0 Settings other than above are prohibited.
20	PIC1BREG3120	Select TIN input signal of TAUD1 CH01. 0: TIN pin input. 1: Signal selected by PIC1BREG3115 to PIC1BREG3117.
19, 18	PIC1BREG3119, PIC1BREG3118	Selects TIN input signal of TAUD1 CH00. 00: Signal selected by PIC1BREG3115 to PIC1BREG3117. 10: DT output signal of ENCAT1EQ0 Settings other than above are prohibited.
17 to 15	PIC1BREG3117, PIC1BREG3116, PIC1BREG3115	Select TIN input signal of TAUD1 CH00 and CH01. 000: TIN pin input. 001: DT output signal of ENCAT1EQ1. Settings other than above are prohibited.
14	Reserved	When read, the value after reset is read. When writing, write the value after reset.
13, 12	PIC1BREG3113, PIC1BREG3112	Select TIN input signal of TAUD0 CH02. 00: TIN pin input. 01: DT output signal of ENCAT0EQ0. Settings other than above are prohibited.
11	PIC1BREG3111	Selects TIN input signal of TAUD0 CH01. 0: TIN pin input 1: The signal selected by PIC1BREG3106 to PIC1BREG3108.

Table 24.54 PIC1BREG31 Register Contents (2/2)

Bit Position	Bit Name	Function
10, 9	PIC1BREG3110, PIC1BREG3109	Select TIN input signal of TAUD0 CH00. 00: Signals selected by PIC1BREG3106 to PIC1BREG3108. 10: DT output signal of ENCAT0EQ0. Settings other than above are prohibited.
8 to 6	PIC1BREG3108, PIC1BREG3107, PIC1BREG3106	Select TIN input signal of TAUD0 CH00 and CH01. 000: TIN pin input. 001: DT output signal of ENCAT0EQ1. Settings other than above are prohibited.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	PIC1BREG3104	Select TIN input signal of TAUJ0 CH03. 0: TIN pin input. 1: DT output signal of ENCAT1IEC.
3	PIC1BREG3103	Select TIN input signal of TAUJ0 CH02. 0: TIN pin input. 1: DT output signal of ENCAT1IEC.
2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1	PIC1BREG3101	Select TIN input signal of TAUJ0 CH01. 0: TIN pin input. 1: DT output signal of ENCAT0IEC.
0	PIC1BREG3100	Select TIN input signal of TAUD0 CH00. 0: TIN pin input. 1: DT output signal of ENCAT0IEC.

24.2.2.43 PIC1BREG50 ─ Timer Input/Output Control Register 50

The PIC1BREG50 register selects TSG30 input signals.

Access: Readable/writable in 16-bit units.

Address: <PIC1B0_base> + $F8_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PIC1BREG5010	—	PIC1BREG5008	PIC1BREG5007	PIC1BREG5006	PIC1BREG5005	—	—	—	—	PIC1BREG5000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Table 24.55 PIC1BREG50 Register Contents

Bit Position	Bit Name	Function
15 to 11	Reserved	When read, the value after reset is read. When writing, write the value after reset.
10	PIC1BREG5010	Selects the signal for input as the TSG30TSTOPC0 signal of the TSG30 timer. 0: INTENCA1I1 input of the ENCA1 timer 1: Setting prohibited.
9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	PIC1BREG5008	Selects the signal for input as the TSG30TSTOPC0 signal of the TSG30 timer. 0: INTENCA0I1 input of the ENCA0 timer 1: Setting prohibited.
7	PIC1BREG5007	Selects the signal for input as the TS0OPCI1 signal of the TSG30 timer. 0: INTTAUD0I7 input of TAUD0 1: Setting prohibited.
6, 5	PIC1BREG5006, PIC1BREG5005	Select the signal for input as the TSG30TSTOPC0 (TS0OPCI0) signal of the TSG30 timer. 01: The signal selected by the PIC1BREG5008 bit 10: The signal selected by the PIC1BREG5010 bit 11: Select input of the INTTAUD0I5 signal of TAUD0. Settings other than above are prohibited.*1
4 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BREG5000	Switches the ENCA signal and external hall sensor signal. For the note on selection of the signals, see **Note 1** of **Section 24.2.2.21, PIC1BTSGHALLSEL ─ Hall Sensor Input Select Register**. 0: Select the pin input ENCA0E0, ENCA0E1, ENCA0EC. 1: Setting prohibited.

Note 1. Set any appropriate value because the value after reset is “setting prohibited”.

24.2.2.44 PIC1BREG51 ─ Timer Input/Output Control Register 51

The PIC1BREG51 register selects TSG31 input signals.

Access: Readable/writable in 16-bit units.

Address: <PIC1B0_base> + FC_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PIC1BREG5110	—	PIC1BREG5108	PIC1BREG5107	PIC1BREG5106	PIC1BREG5105	—	—	—	—	PIC1BREG5100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Table 24.56 PIC1BREG51 Register Contents

Bit Position	Bit Name	Function
15 to 11	Reserved	When read, the value after reset is read. When writing, write the value after reset.
10	PIC1BREG5110	Selects the signal for input as the TSG31TSTOPC0 signal from the TSG31 timer. 0: INTENCA1I1 input signal of the ENCA1 timer 1: Setting prohibited.
9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	PIC1BREG5108	Selects the signal for input as the TSG31TSTOPC0 signal from TSG31 timer. 0: INTENCA0I1 input signal of the ENCA0 timer 1: Setting prohibited.
7	PIC1BREG5107	Selects the signal for input as the TS1OPCI1 signal from TSG31 timer. 0: The INTTAUD1I7 signal input of the TAUD1 timer. 1: Setting prohibited.
6, 5	PIC1BREG5106, PIC1BREG5105	Select the signal for input as the TSG31TSTOPC0 (TS1OPCI0) signal from TSG31 timer. 01: The signal selected by the PIC1BREG5108 bit. 10: The signal selected by the PIC1BREG5110 bit. 11: Select input of the INTTAUD1I5 signal of TAUD1. Settings other than above are prohibited.*1
4 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	PIC1BREG5100	Switches the ENCA signal and external hall sensor signal. For the note on selection of the signals, see **Note 1** of **Section 24.2.2.21, PIC1BTSGHALLSEL ─ Hall Sensor Input Select Register**. 1: Select the pin input ENCA1E0, ENCA1E1, ENCA1EC. Settings other than above are prohibited.*1

Note 1. Set any appropriate value because the value after reset is “setting prohibited”.

24.2.3 Function

24.2.3.1 Simultaneous Start Trigger Function

(1) Overview

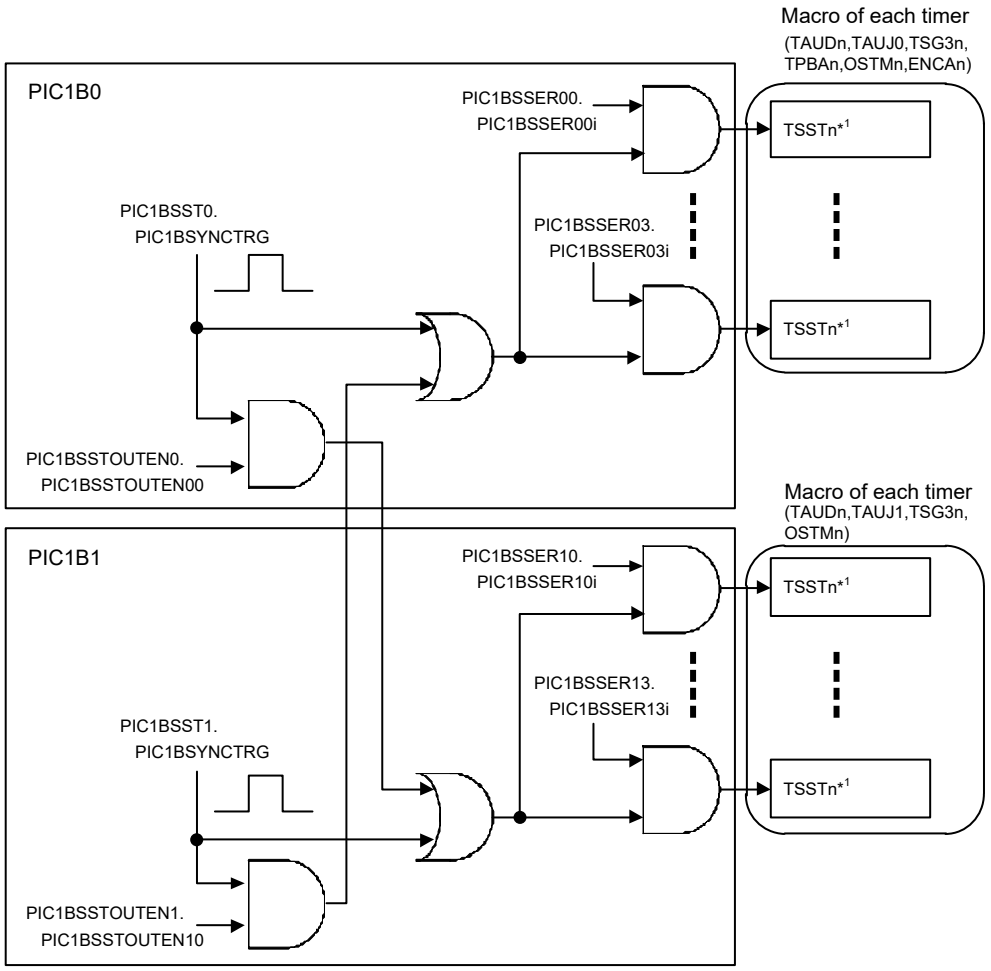
The function allows any combination of timers (TAUDn, TAUJn, TSG3n, TPBAn, OSTMn, and ENCAn) to be started simultaneously.

(2) Configuration

The timers which support simultaneous start trigger function are listed as follows.

- TAUDn
- TAUJn
- TSG3n
- TPBAn
- OSTMn
- ENCAn

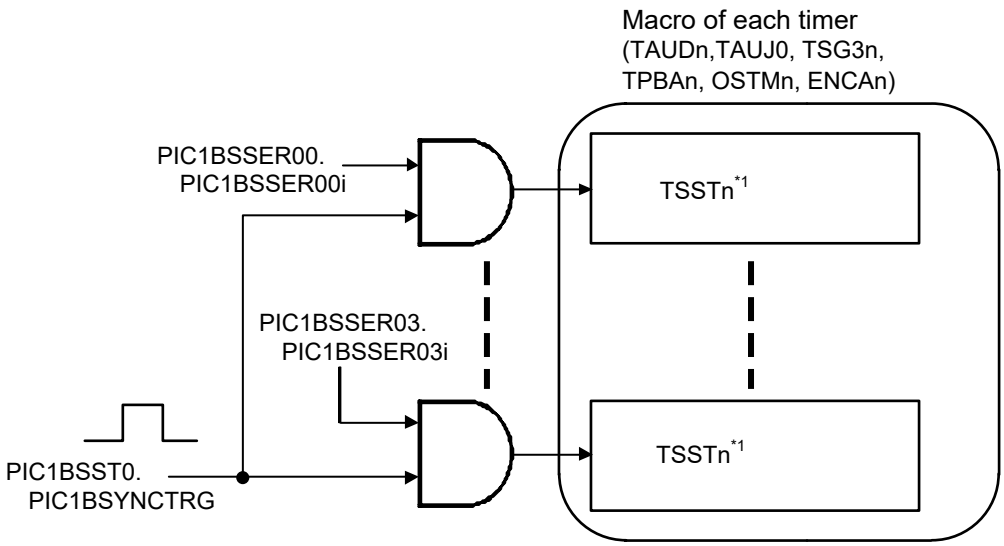
Note: The block diagram with two PIC1B units applies to the C1M-A2 and that with one PIC1B unit applies to the C1M-A1.



Note 1. Simultaneous start trigger signal for each timer

i: The applicable bit

Figure 24.3 Block Diagram of Simultaneous Start Trigger Function with Two PIC1B Units



Note 1. Simultaneous start trigger signal for each timer

i: The applicable bit

Figure 24.4 Block Diagram of Simultaneous Start Trigger Function with One PIC1B Unit

Setting the PIC1BSYNCTRG bits in the simultaneous start trigger control registers (PIC1BSSTk) to 1 after removing the trigger masks for the target timers activates input of the start triggers for each of the timers, and operation of the given timers starts.

In C1M-A2, the timers connected to the PIC1B0 unit can be simultaneously started by setting the PIC1BSSTOUTEN1 register of the PIC1B1 unit. Similarly, the timers connected to the PIC1B1 unit can be simultaneously started by setting the register of the PIC1B0 unit.

(3) Registers

The PIC1B registers set by this function is listed as follows. For setting value of the registers, see **Sections 24.2.2.2 to 24.2.2.12**.

- PIC1B registers to be set
  PIC1BSSTk
  PIC1BSSER00
  PIC1BSSER01
  PIC1BSSER02
  PIC1BSSER03
  PIC1BSSER10*[1]
  PIC1BSSER11*[1]
  PIC1BSSER12*[1]
  PIC1BSSER13*[1]
  PIC1BSSTOUTEN0*[1]
  PIC1BSSTOUTEN1*[1]

Note 1. This register is not provided for the C1M-A1.

(4) Function

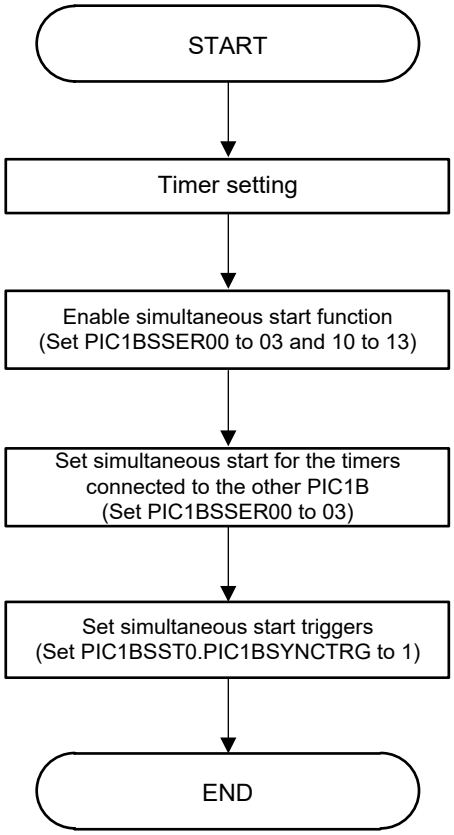
The function allows any combination of timers (TAUDn, TAUJn, TSG3n, TPBAn, OSTMn, and ENCAn) to be started simultaneously.

(5) Flow chart

The following figure shows the setting flow of this function.

Note: The flowchart with two PIC1B units applies to the C1M-A2 and that with one PIC1B unit applies to the C1M-A1.

For two PIC1B units



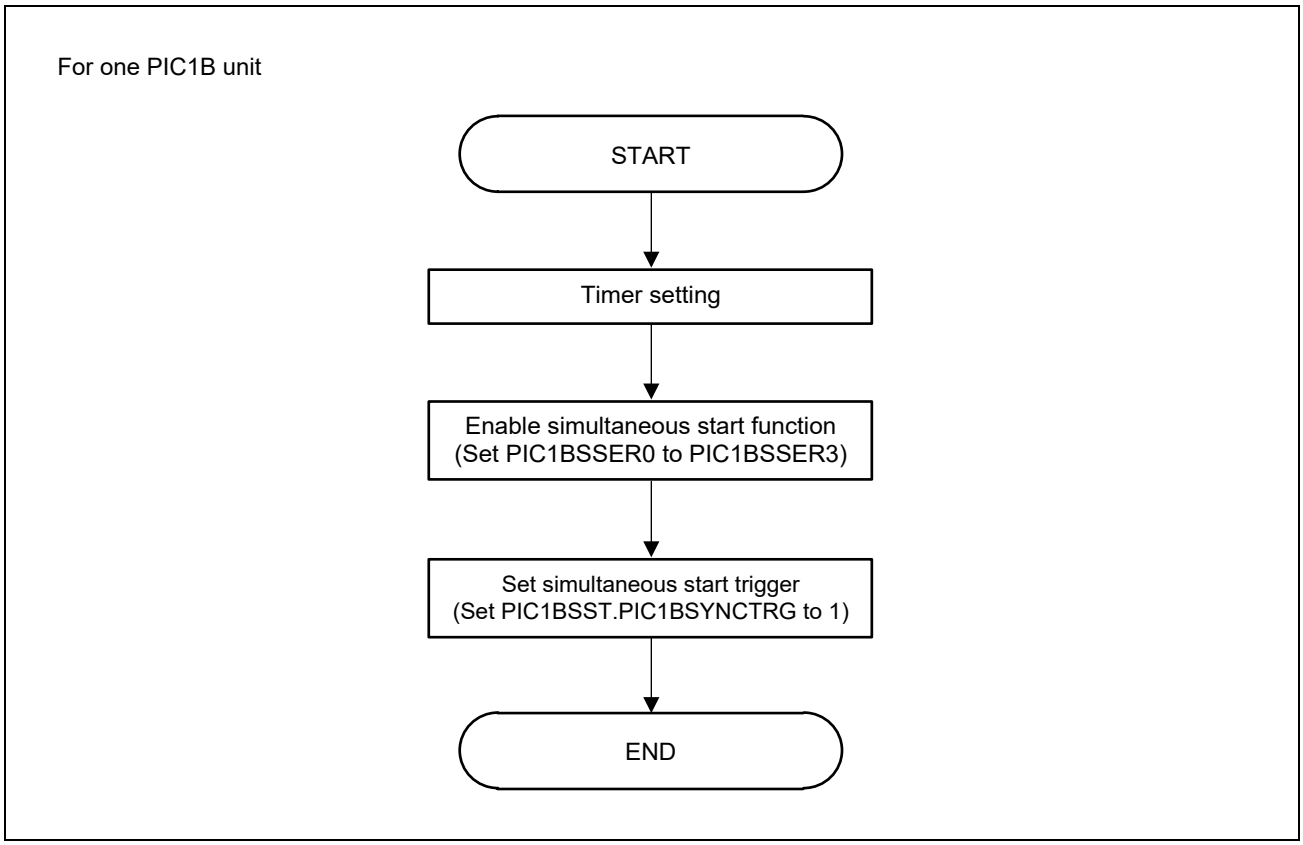


Figure 24.5 Setting Flow

Setting procedures are described as follows.

- Timer setting
  Set the timers to start simultaneously.
- Enable simultaneous start function
  Set 1 to the applicable bits of the PIC1BSSER00, PIC1BSSER01, PIC1BSSER02, and PIC1BSSER03 registers to enable simultaneous start of timers.
  In C1M-A2, enable the timers of the PIC1B1 unit by setting the applicable bits of the PIC1BSSER10, PIC1BSSER11, PIC1BSSER12, and PIC1BSSER13 registers.
- Set simultaneous start for the timers connected to the other PIC1B unit in C1M-A2
  In C1M-A2, set 1 to the PIC1BSSTOUTEN00 bit of the PIC1BSSTOUT0 register to enable simultaneous start of the timers connected to the PIC1B1 unit. Similarly, set 1 to the PIC1BSSTOUTEN10 bit of the PIC1BSSTOUT1 register to enable simultaneous start of the timers connected to the PIC1B0 unit.
- Setting simultaneous start triggers
  Setting the PIC1BSYNCTRG bits in simultaneous start trigger control registers (PIC1BSSTk) to 1 simultaneously starts the corresponding timers. In the C1M-A2, if the PIC1BSSTOUTk register of one PIC1B unit is set to enable simultaneous starting of the timers connected to the other PIC1B, setting of the PIC1BSSTk register of the other PIC1B is not required.

24.2.3.2 TSG Simultaneous Start Function (External Trigger)

(1) Overview

This function allows any combination of TSG3 timers to be started simultaneously by the trigger input from outside (TSGTSST).

(2) Configuration

The timer which supports this function is below.

- TSG3n

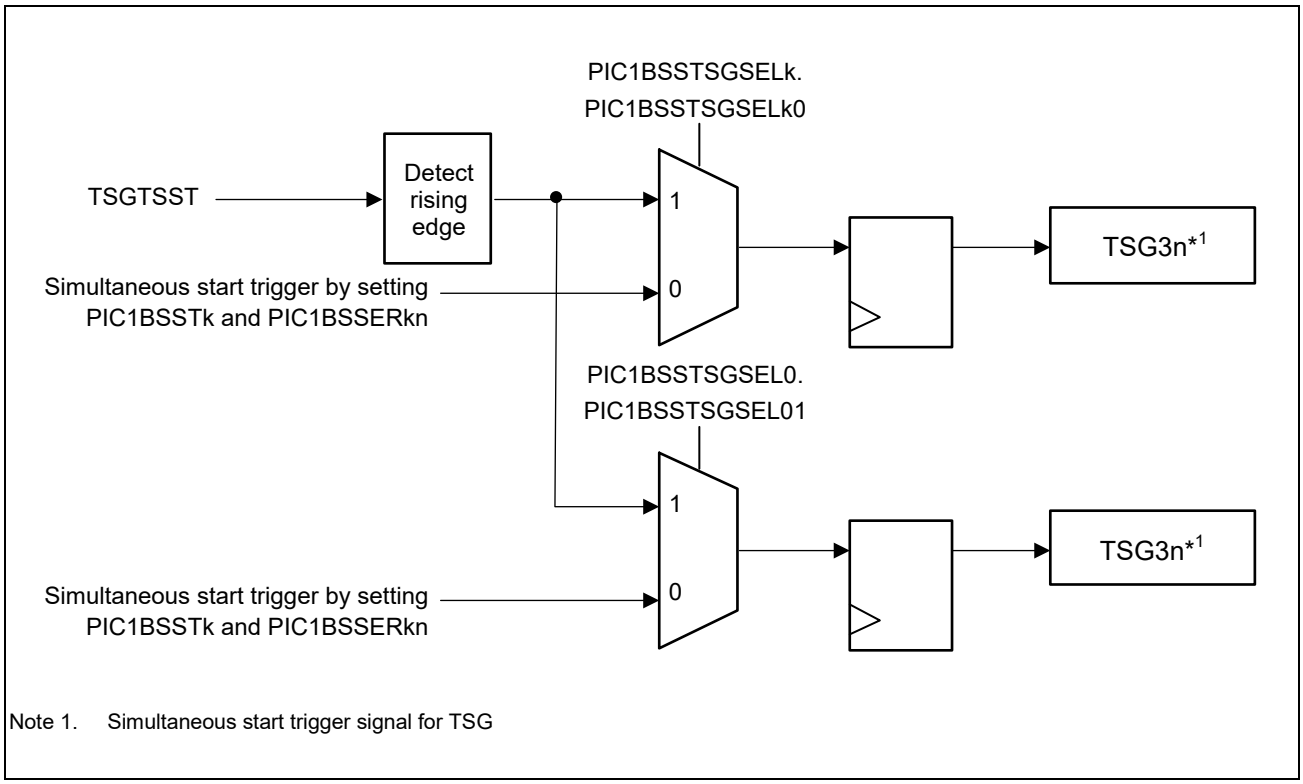


Figure 24.6 Block Diagram of TSG Simultaneous Start Function

(3) Registers

The PIC1B registers set by this function are listed as follows. For setting value of the registers, see **Sections 24.2.2.4 and 24.2.2.5**.

- PIC1B registers to be set
  PIC1BSSTSGSEL0
  PIC1BSSTSGSEL1*1

Note 1. This register is not provided for the C1M-A1.

(4) Function

Start the TSG3 timers are started by the trigger input from outside (TSGTSST).

(5) Flow chart

The following figure shows the setting flow of this function.

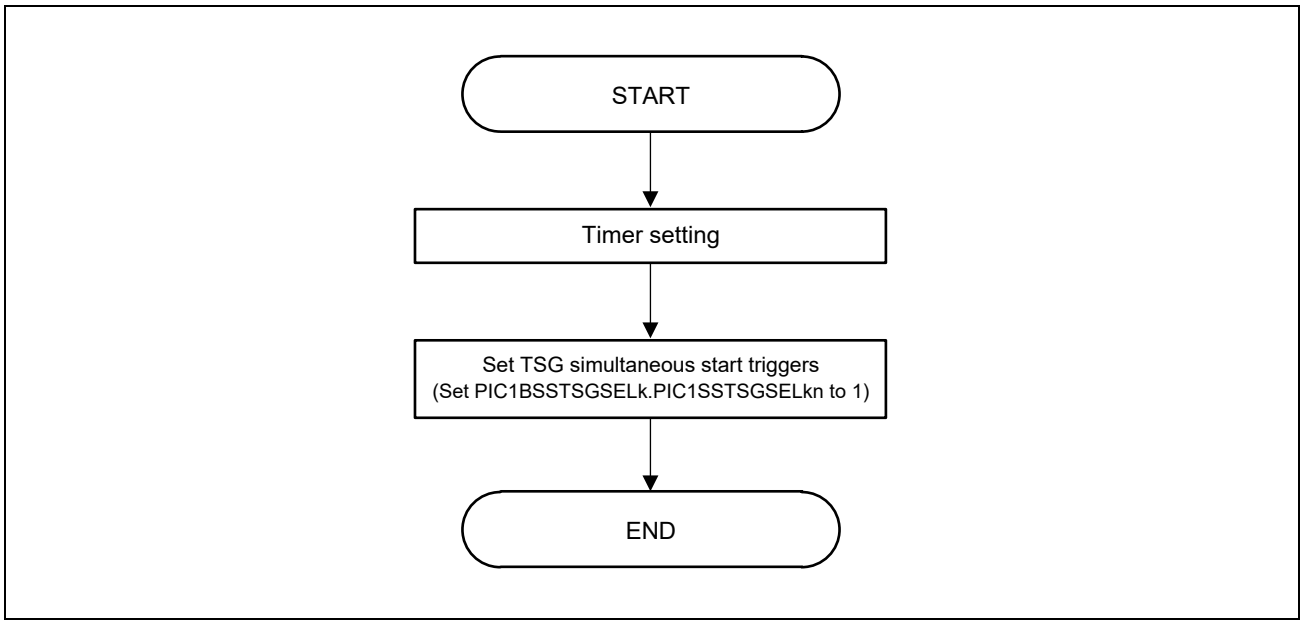


Figure 24.7 Setting Flow

Setting procedure is described as follows.

- Timer setting
  Set the TSG3 timers to start simultaneously.
- Set TSG simultaneous start trigger
  Set the PIC1BSSTSGSELkn bits to 1 in the selected TSG3 simultaneously start trigger select registers k (PIC1BSSTSGSELk). Input of a trigger on the external pin (TSGTSST) then starts the selected TSG3 timers.

24.2.3.3 PWM Output Function with Dead Time

(1) Overview

This function generates and outputs PWM waveforms with the dead time from one phase to three phases using TAUDn.

The PWM output function of TAUD sets only the clear timing in a period under the duty ratio specification. On the other hand, this function can specify the set timing in addition to the clear timing to output more flexible PWM waveforms with the dead time.

The following table lists the number of channels used.

Table 24.57 Number of TAUDn Channels for Use in PWM Output with Dead Time

PWM Output with Dead Time	Number of TAUDn Channels
One-phase PWM output (U phase/UB phase)	5 channels (master channel: 1, slave channel: 4)
Two-phase PWM output (U phase/UB phase, V phase/VB phase)	9 channels (master channel: 1, slave channel: 8)
Three-phase PWM output (U phase/UB phase, V phase/VB phase, W phase/WB phase)	13 channels (master channel: 1, slave channel: 12)

Note: Above are examples of combination.

Usages of each channel of TAUDn are listed in the following table. CH2 is used as the master channel.

Table 24.58 Usage of TAUDn Channels

TAUDn Channel	U phase/ UB phase	V phase/ VB phase	W phase/ WB phase	Usage
CH0	—	—	—	Not used.
CH1	—	—	—	Not used.
CH2	√	√	√	Carrier period (common to each phase)
CH3	—	—	—	Not used.
CH4	√	—	—	Duty (U phase setting)
CH5	√	—	—	Duty (U phase clearing)
CH6	—	√	—	Duty (V phase setting)
CH7	—	√	—	Duty (V phase clearing)
CH8	—	—	√	Duty (W phase setting)
CH9	—	—	√	Duty (W phase clearing)
CH10	√	—	—	U phase output (TOUT10)
CH11	√	—	—	UB phase output (TOUT11)
CH12	—	√	—	V phase output (TOUT12)
CH13	—	√	—	VB phase output (TOUT13)
CH14	—	—	√	W phase output (TOUT14)
CH15	—	—	√	WB phase output (TOUT15)

Note: √: Used; — : Not used

(2) Configuration

The PWM output function with the dead time is realized by using the PWM output function/one-phase output PWM function of TAUDn and PIC1B in combination. The following figure shows the block diagram of the PWM output function with the dead time.

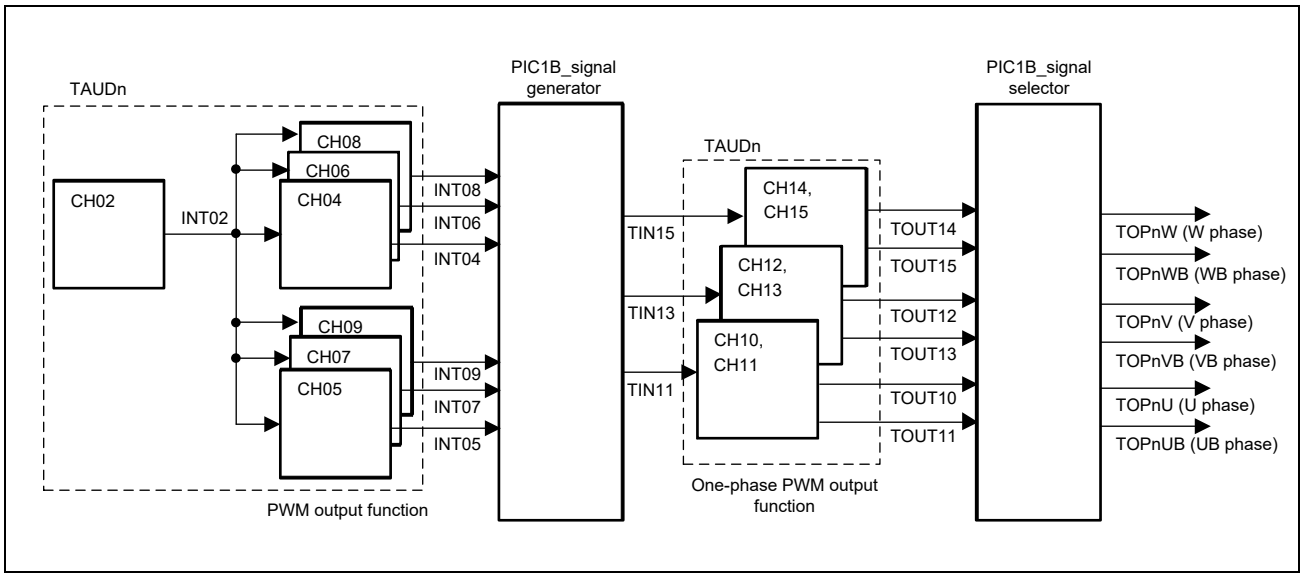


Figure 24.8 Block Diagram of PWM Output Function with Dead Time

The configuration of this function is described as follows using the PWM output of U phase/UB phase as an example.

- [TAUDn] PWM output function
  CH02, CH04, and CH05 are used in combination. Setting the period, U phase set value, and U phase clear value to CDR02, CDR04, and CDR05, respectively generates the set and clear signals (INT04 and INT05).
- [PIC1B_signal generator] RS flip-flop circuit (RSn2)
  Selecting the INT04 and INT05 inputs allows TIN11 (PWM signal) to be generated.
- [TAUDn] One-phase PWM output function
  CH10 and CH11 are used in combination. Setting the dead time value to CDR11, and inserting dead time into the PWM signal to be input to TIN11 allows TOUT10 (U phase PWM signal) and TOUT11 (UB phase PWM signal) to be output.
- [PIC1B_signal selector]
  TOUT10 and TOUT11 inputs are selected and output to TOPnU and TOPnUB pins, respectively.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

(3) Registers

The block diagram of PIC1B is shown in the following figure.

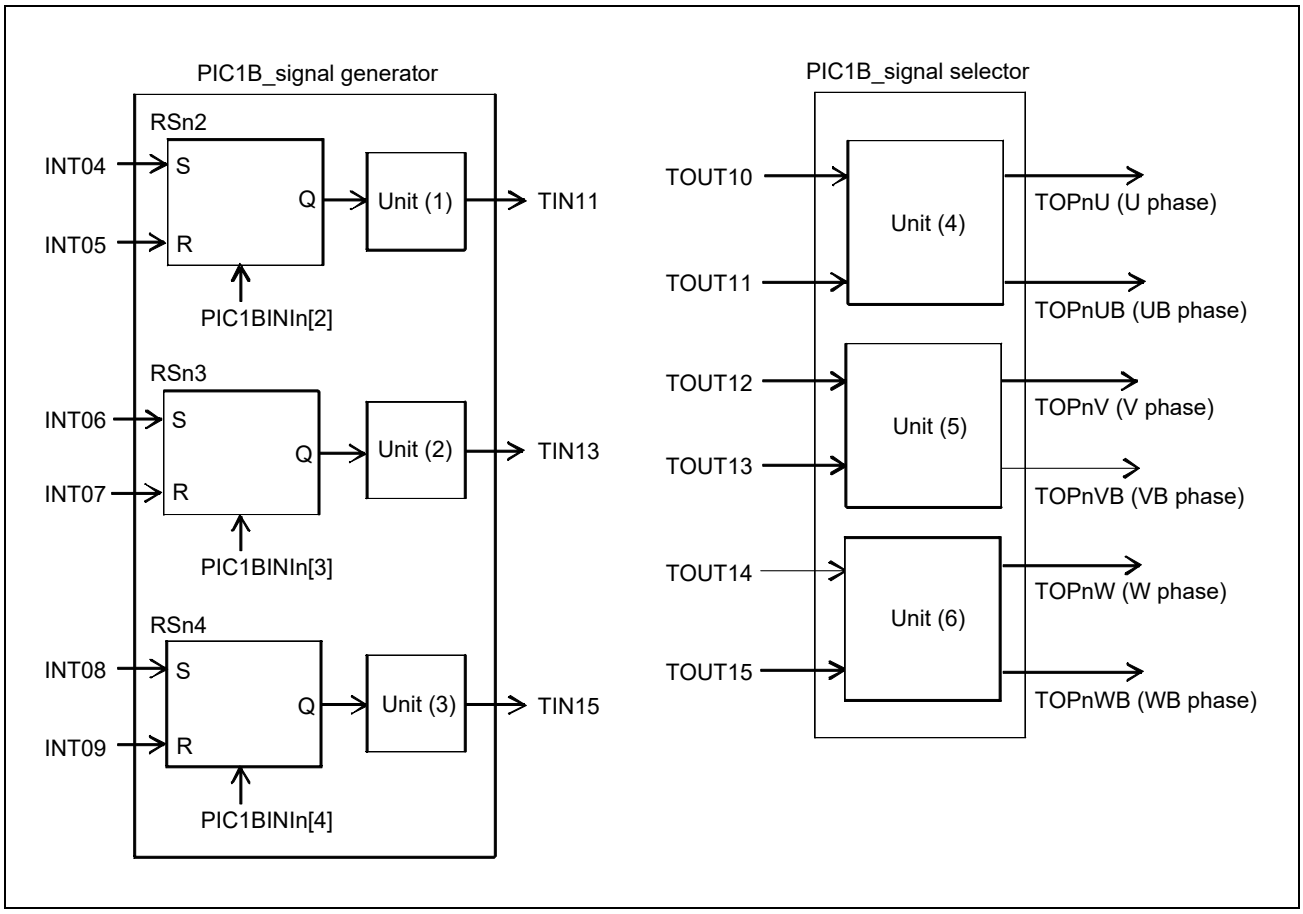


Figure 24.9 Block Diagram of PIC1B

The values of PIC1B registers used in this function are as follows.

U phase/UB phase

The values to output the signal Q from RSn2 as TIN11. (**Figure 24.9**, unit (1))

PIC1BREG2n2[19:18] = 10_B

PIC1BREG2n2[2] = 1_B

PIC1BTAUDnSEL[23:22] = 00_B

The values to output TOUT10 and TOUT11 as TOPnU and TOPnUB, respectively (**Figure 24.9**, unit (4))

PIC1BREG2n1[19:16] = 0000_B

PIC1BREG2n3[2:0] = 000_B

PIC1BREG2n3[6:4] = 000_B

V phase/VB phase

The values to output the signal Q from RSn3 as TIN13. (**Figure 24.9**, unit (2))

PIC1BREG2n2[23:22] = 10_B

PIC1BREG2n2[3] = 1_B

PIC1BTAUDnSEL[27:26] = 00_B

The values to output TOUT12 and TOUT13 as TOPnV and TOPnVB, respectively (**Figure 24.9**, unit (5))

PIC1BREG2n1[23:20] = 0000_B

PIC1BREG2n3[10:8] = 000_B

PIC1BREG2n3[14:12] = 000_B

W phase/WB phase

The values to output the signal Q from RSn4 as TIN15. (**Figure 24.9**, unit (3))

PIC1BREG2n2[27:26] = 10_B

PIC1BREG2n2[4] = 1_B

PIC1BTAUDnSEL[31:30] = 00_B

The values to output TOUT14 and TOUT15 as TOPnW and TOPnWB, respectively (**Figure 24.9**, unit (6))

PIC1BREG2n1[27:24] = 0000_B

PIC1BREG2n3[18:16] = 000_B

PIC1BREG2n3[22:20] = 000_B

Enables initialization of RSn2 to RSn4

The values to enable initialization of RSn2 to RSn4

PIC1BINIn0[4] = 1_B (initialized)

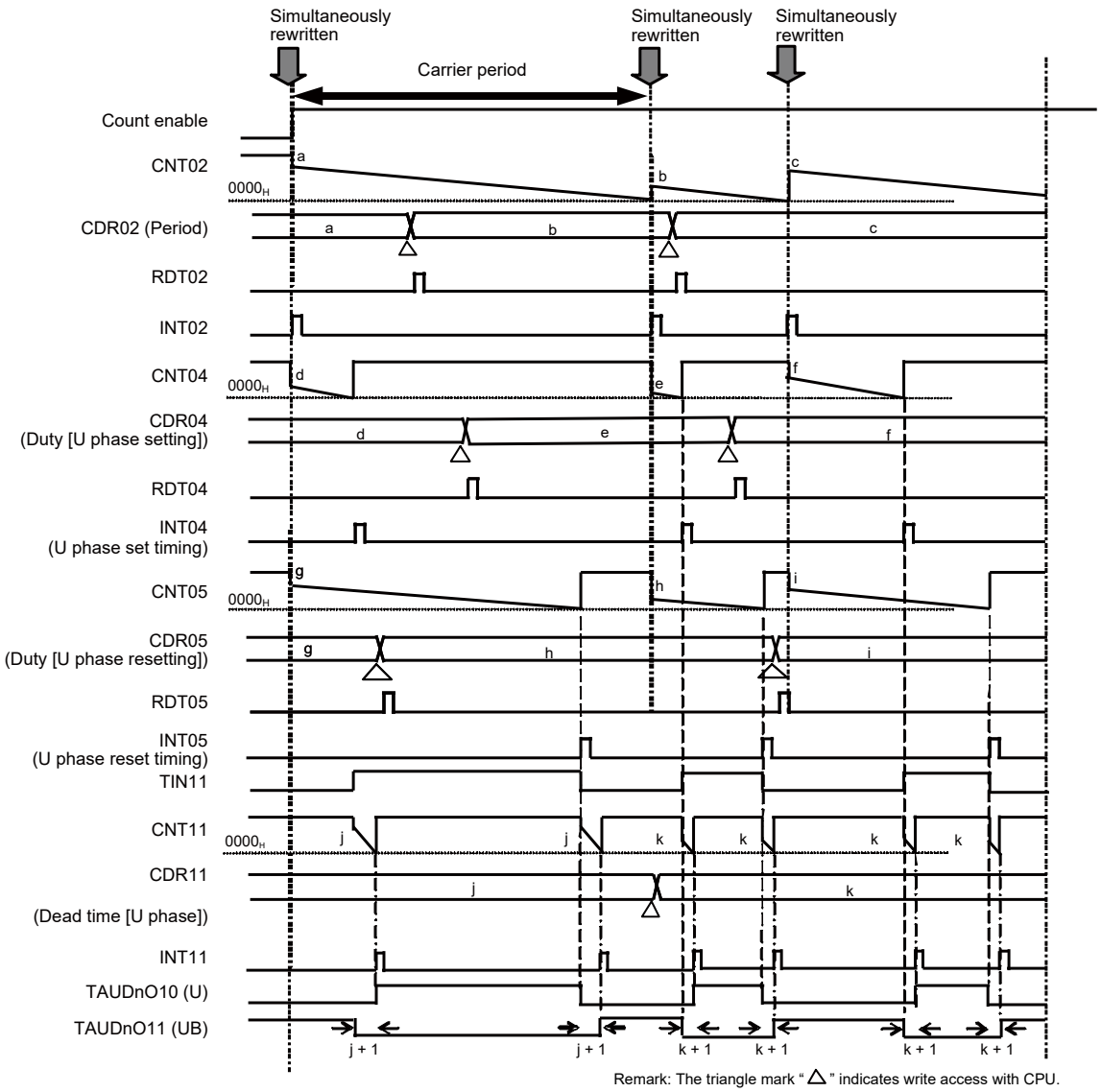
PIC1BINIn0[3] = 1_B (initialized)

PIC1BINIn0[2] = 1_B (initialized)

(4) Function

Detail of the function is described using the one-phase PWM output (U phase/UB phase) with dead time as an example.

The following figure shows the timing diagram.



Note: For the detail of TAUD function including operation timing, see **Section 18, Timer Array Unit D (TAUD)**.

Figure 24.10 Timing Diagram of One-Phase PWM Output with Dead Time (U phase/UB phase)

(1) Using the simultaneous start trigger function, the timers to be used are started simultaneously.*1

(2) For CH04 and CH05, CH02 underflow allows the set values in CDR04 and CDR05 to be reloaded to CNT04 and CNT05, respectively.

(3) CH04 underflow allows the INT04 to be generated and TIN11 to become high level. CH05 underflow allows the INT05 to be generated and TIN11 to become low level, and the PWM waveform to be generated.

(4) Both edges of TIN11 allows the set values to be reloaded to CNT11.*2

(5) CH11 underflow allows the INT11to be generated and TAUDnO10 to become high level. CH05 underflow allows the INT05 to be generated and TAUDnO10 to become low level. The PWM waveform of U phase is generated and output to TOPnU.

(6) The rising edge of TIN11 allows TAUDnO11 to become low level. CH11 underflow allows the INT11to be generated and TAUDnO11 to become high level. The PWM waveform of UB phase is generated and output to TOPnUB.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

Note 1. Select the count clock signal of the same clock for TAUDn.

Note 2. Set the effective edge to be detected by TIN11 of TAUDn as both edges (rising edge and falling edge) for this function.

Detail of the function when setting the longer clear timing than the carrier period is described using V phase/VB phase as an example.

The following figure shows the timing diagram.

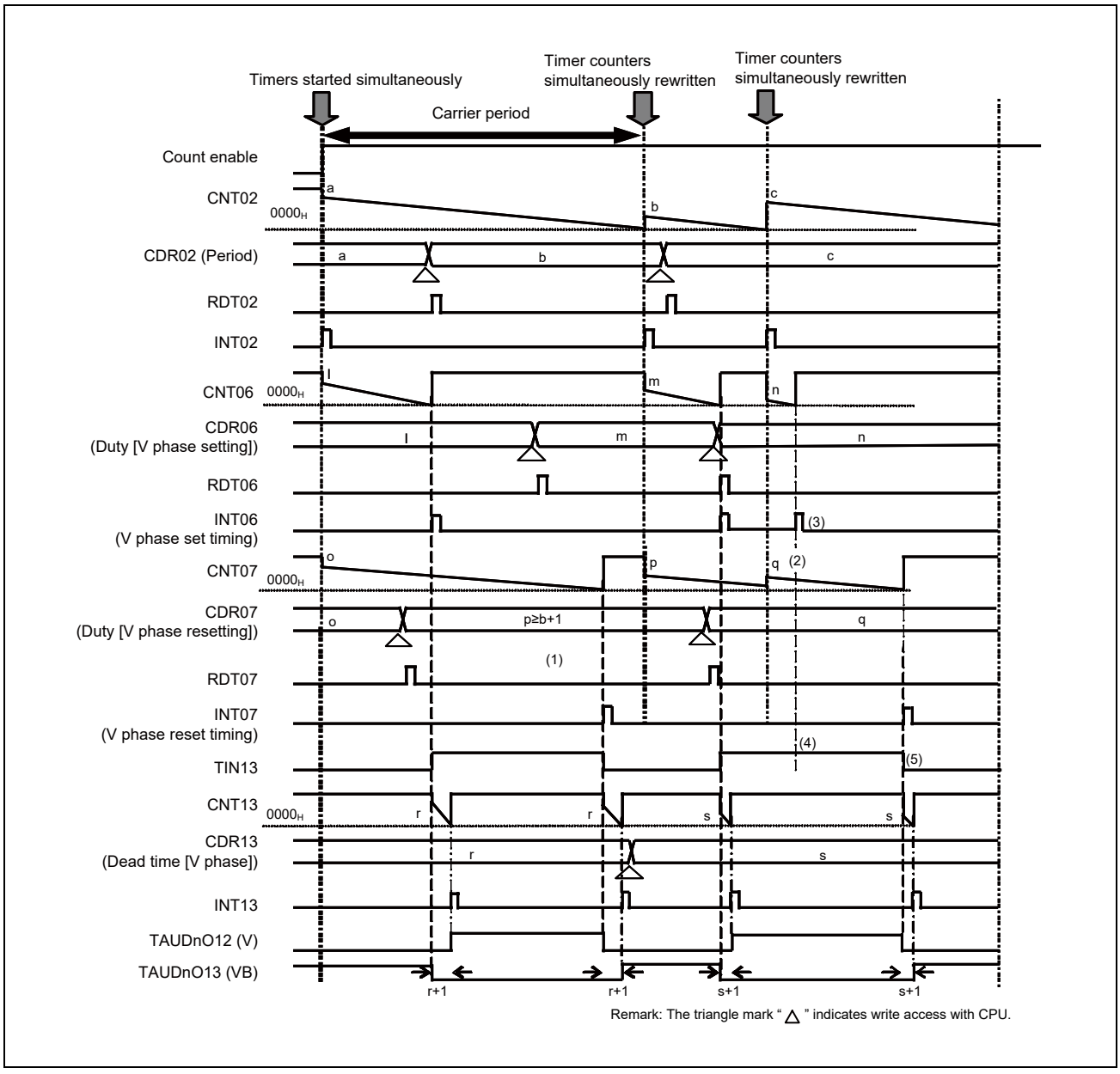


Figure 24.11 Clear Timing Value > Carrier Period Value (V Phase/VB Phase)

Setting the longer clear time than the carrier period allows the waveform output to be extended over the carrier period.

An operation example of one-phase PWM output (V phase/VB phase) is shown as follows. The operation flow from timer operation start to one-phase PWM output by one-phase PWM output function, refer to the description for one-phase PWM output with dead time (U phase/UB phase).

When the value set in CH07 is longer than the value set in CH02 (**Figure 24.11** (1)), underflow of the carrier period timer is generated before generation of V phase clearing timing signal (INT07), and the value is reloaded (**Figure 24.11** (2)).
This causes generation of the V phase set timing signal (INT06) to precede, thus preventing generation of the V phase clear timing signal (INT07) (**Figure 24.11** (3)).
Then, the PWM waveform is not influenced because V phase set timing signal is ignored in the PIC circuit (**Figure 24.11** (4)). Therefore, the PWM waveform is output extended over the carrier period (**Figure 24.11** (5)).

The following figure shows the timing diagram of three-phase PWM output with the dead time.

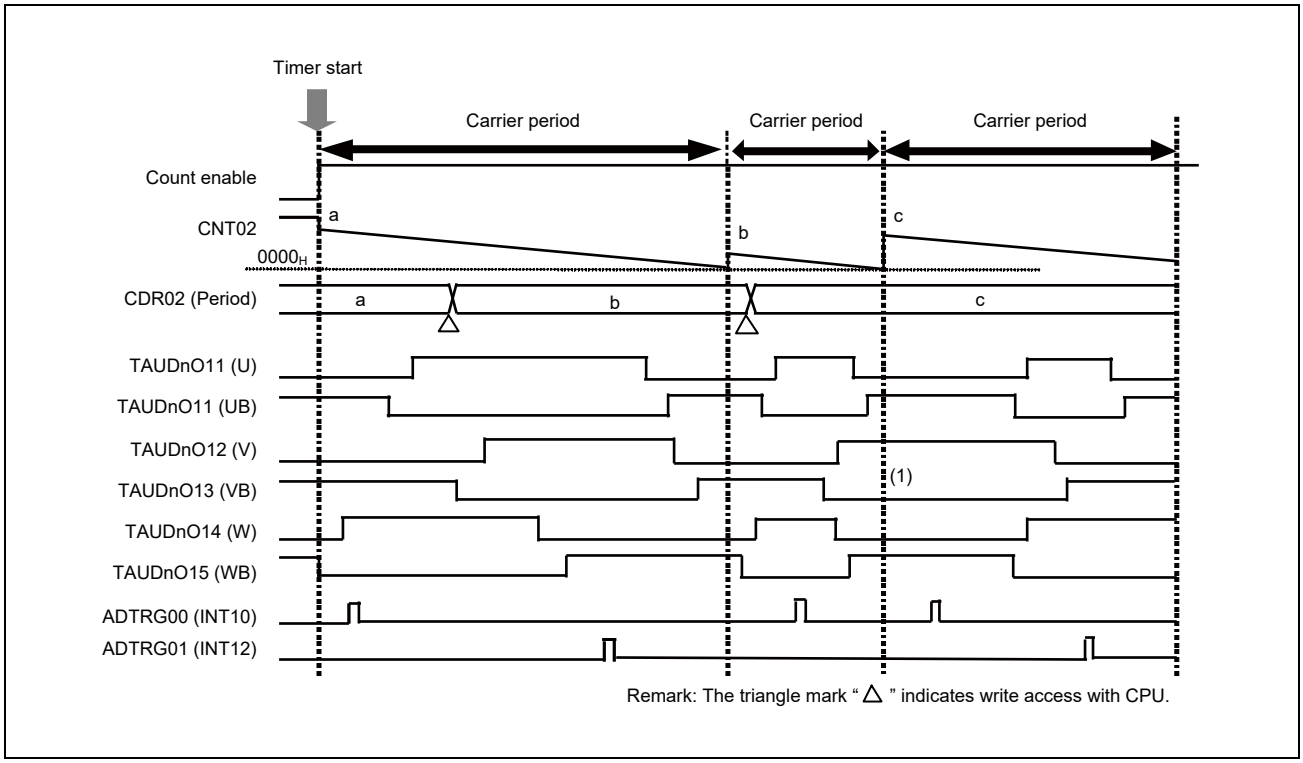
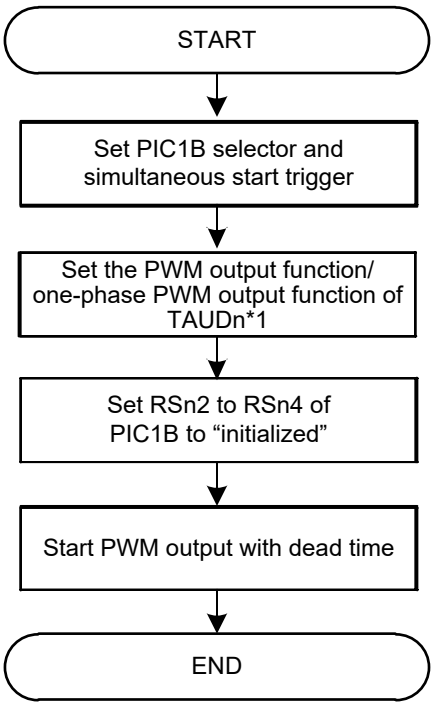


Figure 24.12 Timing Diagram of Three-Phase PWM Output with Dead Time

(5) Flow chart

The following flow chart shows the PMW output function with the dead time.



Note 1. For the setting of PWM output function of TAUDn, see **Section 18.4.12.1, PWM Output Function**. For the setting of one-phase PWM output function, see **Section 18.4.12.11, One-Phase PWM Output Function**.

Figure 24.13 Setting Flow

24.2.3.4 High Accuracy Triangle Wave PWM Output Function with Dead Time

(1) Overview

This function generates triangle wave PWM output with dead time from one-phase to three-phases. Compared to the triangle wave PWM output function with dead time of TAUDn, this function enables a control of the variable dead time range, where duty cycle is close to 100% and 0%.

The TAUDn channels used in this function are listed in the following table.

Table 24.59 Number of TAUDn Channels for Use in Triangle Wave PWM Output with Dead Time

High Accuracy Triangle Wave PWM Output with Dead Time	Number of TAUDn Channels
One-phase PWM output (U phase/UB phase)	5 channels (master channel: 1, slave channel: 4)
Two-phase PWM output (U phase/UB phase, V phase/VB phase)	9 channels (master channel: 1, slave channel: 8)
Three-phase PWM output (U phase/UB phase, V phase/VB phase, W phase/WB phase)	13 channels (master channel: 1, slave channel: 12)

Note: Above are examples of combination.

Usages of each channel of TAUDn are listed in the following table. CH2 is used as the master channel of CH3 to 9.

CHm is used as the master channel of CHm+1 (m = 10, 12, 14).

Table 24.60 Usage of TAUDn Channels

TAUDn Channel	U phase/ UB phase	V phase/ VB phase	W phase/ WB phase	Usage
CH0	—	—	—	Not used.
CH1	—	—	—	Not used.
CH2	√	√	√	Carrier period (common to each phase)
CH3	—	—	—	Not used.
CH4	√	—	—	Triangle PWM output with dead time (U phase/UB phase)
CH5	√	—	—	
CH6	—	√	—	Triangle PWM output with dead time (V phase/VB phase)
CH7	—	√	—	
CH8	—	—	√	Triangle PWM output with dead time (W phase/WB phase)
CH9	—	—	√	
CH10	√	—	—	Reduced dead time pulse (U phase/UB phase)
CH11	√	—	—	
CH12	—	√	—	Reduced dead time pulse (V phase/VB phase)
CH13	—	√	—	
CH14	—	—	√	Reduced dead time pulse (W phase/WB phase)
CH15	—	—	√	

Note: √: Used; — : Not used

(2) Configuration

The high accuracy triangle wave PWM output function with the dead time is realized by using the triangle wave PWM output function/one-shot pulse output function of TAUDn and PIC1B in combination. The following figure shows the block diagram of the high accuracy triangle wave PWM output function with the dead time.

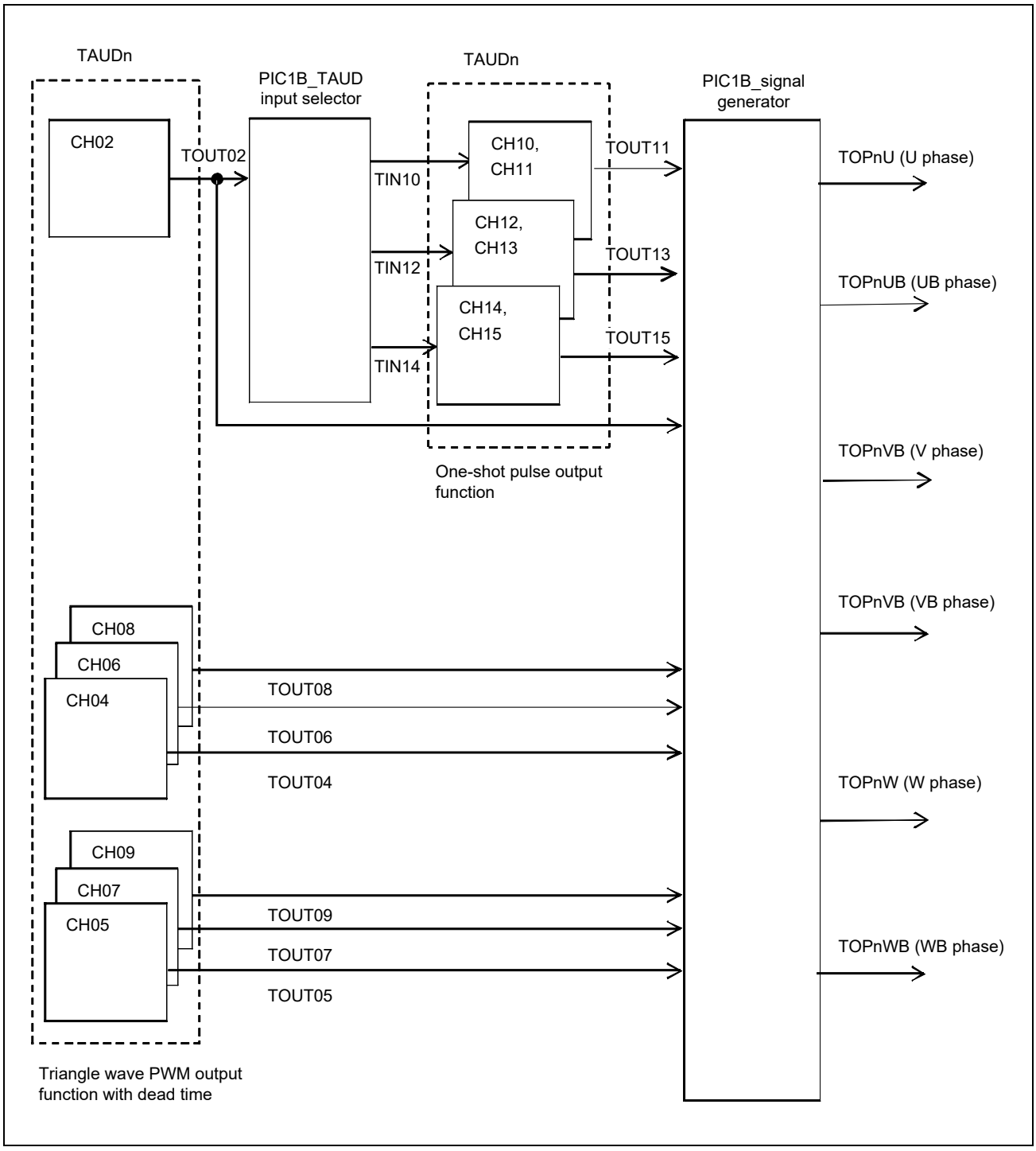


Figure 24.14 Block Diagram of High Accuracy Triangle Wave Three-Phase PWM Output with Dead Time

The configuration of this function is described as follows using the PWM output of U phase/UB phase as an example.

- [PIC1B_TAUD input selector]
  TOUT02 is selected and output to TIN10.
- [TAUDn] One-shot pulse output function
  CH10 and CH11 are used in combination. Setting the delay value and the pulse width to CDR10 and CDR11, respectively allows the one-shot pulse output signal (TOUT11) to be generated.
- [TAUDn] Triangle wave PWM output function with dead time
  CH02, CH04, and CH05 are used in combination. Setting the period, duty, and dead time to CDR02, CDR04, and CDR05, respectively generates the triangle wave PWM output signal with dead time (TOUT04 and TOUT05).
- [PIC1B_signal generator]
  The reduced dead time pulses (UO1 and UO2) are generated at PFN001 from the one-shot pulse output signal. UO1 and UO2 are synthesized with TOUT04 and TOUT05 at FN00 and FN01, respectively, added dead time variable range pulse, and TOPnU (U phase PWM signal) and TOPnUB (UB phase PWM signal) are generated.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

(3) Registers

The following figure shows the block diagram of PIC1B.

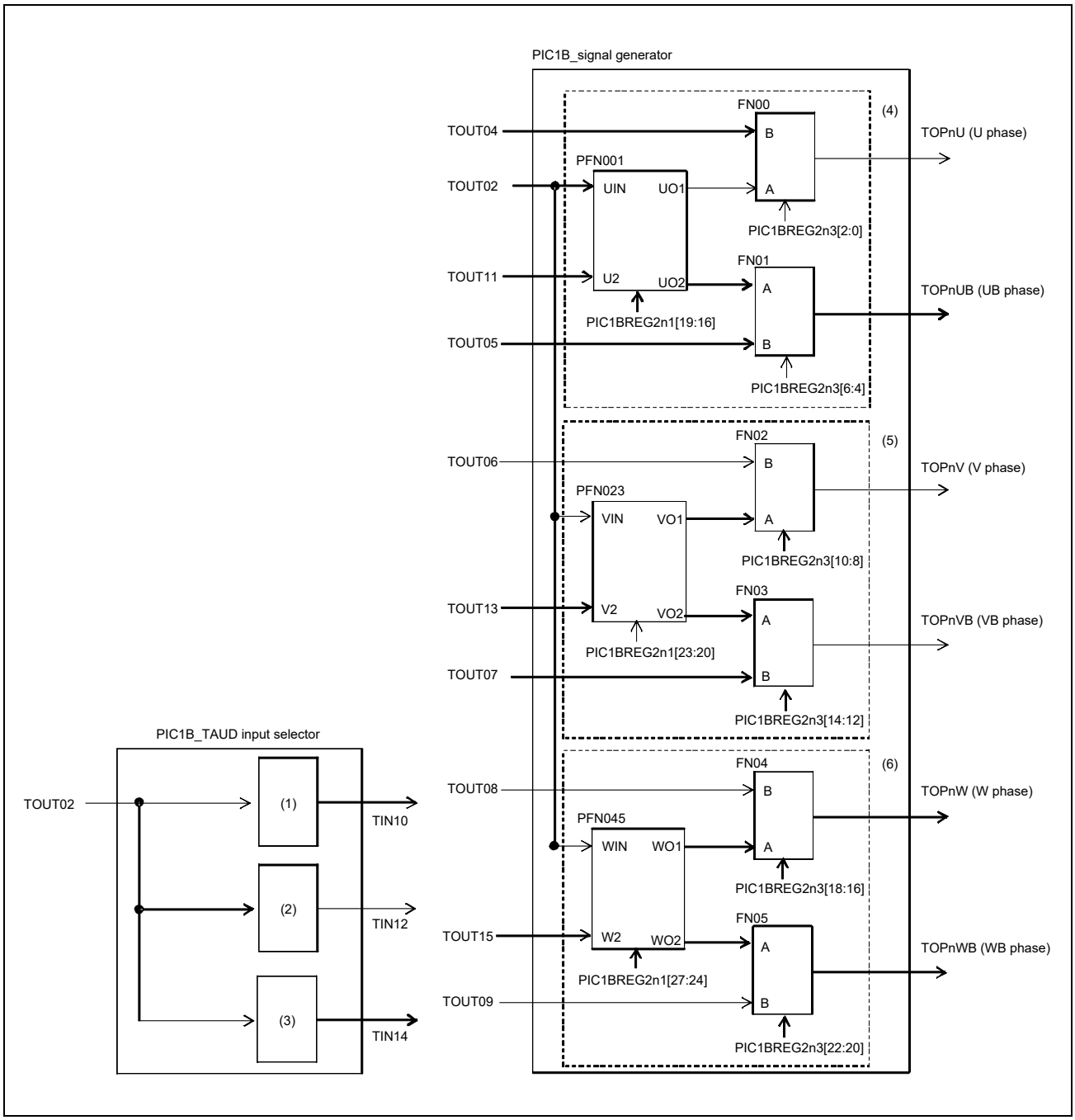


Figure 24.15 Block Diagram of PIC1B

The values of PIC1B registers used in this function are as follows.

Unit (1): PIC1B_TAUD input selector (U phase/UB phase)

The values to output TOUT02 as TIN10 (common to active high/low)

PIC1BREG2n0[18] = 1_B

PIC1BREG2n2[17:16] = 10_B

PIC1BTAUDnSEL[21:20] = 00_B

Unit (2): PIC1B_TAUD input selector (V phase/VB phase)

The values to output TOUT02 as TIN12 (common to active high/low)

PIC1BREG2n0[18] = 1_B

PIC1BREG2n2[21:20] = 10_B

PIC1BTAUDnSEL[25:24] = 00_B

Unit (3): PIC1B_TAUD input selector (W phase/WB phase)

The values to output TOUT02 as TIN14 (common to active high/low)

PIC1BREG2n0[18] = 1_B

PIC1BREG2n2[25:24] = 10_B

PIC1BTAUDnSEL[29:28] = 00_B

Unit (4): PIC1B_signal generator (U phase/UB phase)

The values to output one-phase PWM (active high/low) from TAUDnO10 and TAUDnO11

PIC1BREG2n1[19:16] = 1010_B (active high), 1111_B (active low)

PIC1BREG2n3[06:04] = 100_B (active high), 101_B (active low)

PIC1BREG2n3[02:00] = 100_B (active high), 101_B (active low)

Unit (5): PIC1B_signal generator (V phase/VB phase)

The values to output one-phase PWM (active high/low) from TAUDnO12 and TAUDnO13

PIC1BREG2n1[23:20] = 1010_B (active high), 1111_B (active low)

PIC1BREG2n3[14:12] = 100_B (active high), 101_B (active low)

PIC1BREG2n3[10:08] = 100_B (active high), 101_B (active low)

Unit (6): PIC1B_signal generator (W phase/WB phase)

The values to output one-phase PWM (active high/low) from TAUDnO14 and TAUDnO15

PIC1BREG2n1[27:24] = 1010_B (active high), 1111_B (active low)

PIC1BREG2n3[22:20] = 100_B (active high), 101_B (active low)

PIC1BREG2n3[18:16] = 100_B (active high), 101_B (active low)

(4) Function

Detail of the function is described using U phase/UB phase as an example. The function with V phase/VB phase and W phase/WB phase are operated under the same logic as that of U phase/UB phase with different input signals and register settings.

- U phase combination circuit (PFN001)
  This circuit generates reduced dead time pulses*1 (FN00A and FN01A) that are used to insert the dead time pulse generated by the one-shot pulse output function into the triangle wave PWM generated by the triangle wave PWM output function with the dead time. For the block diagram, see **Figure 24.1, Block Diagram of PFN001**.

Note 1. Reduced dead time pulses are the pseudo pulses that are inserted into the PWM output provided by the triangle wave PWM output function with dead time of TAUDn and that are modeled on the dead time pulses that are generated in the range where duty cycle is close to 100% or 0% in PWM output in HT-PWM mode of TSG3n.

- Logical operation circuits (FN0i (i = 0, 1))
  This circuit synthesizes the triangle wave PWM output (TOUT04 and TOUT05) from the triangle wave PWM output function with the dead time and the outputs from the combinational circuit PFN001 (UO0 and UO1) to generate PWM with reduced dead time pulses inserted. The synthesizing logic of this circuit is selected with PIC1BREG2n3k (k = 00 to 02, 04 to 06). For the block diagram, see **Figure 24.2, Block Diagram of FN00**.

Detail of the function is described using high accuracy triangle wave PWM output function (U phase/UB phase) as an example.

The following figure shows the timing diagram when U phase duty cycle = 0% and UB phase duty cycle = 100% at active high.

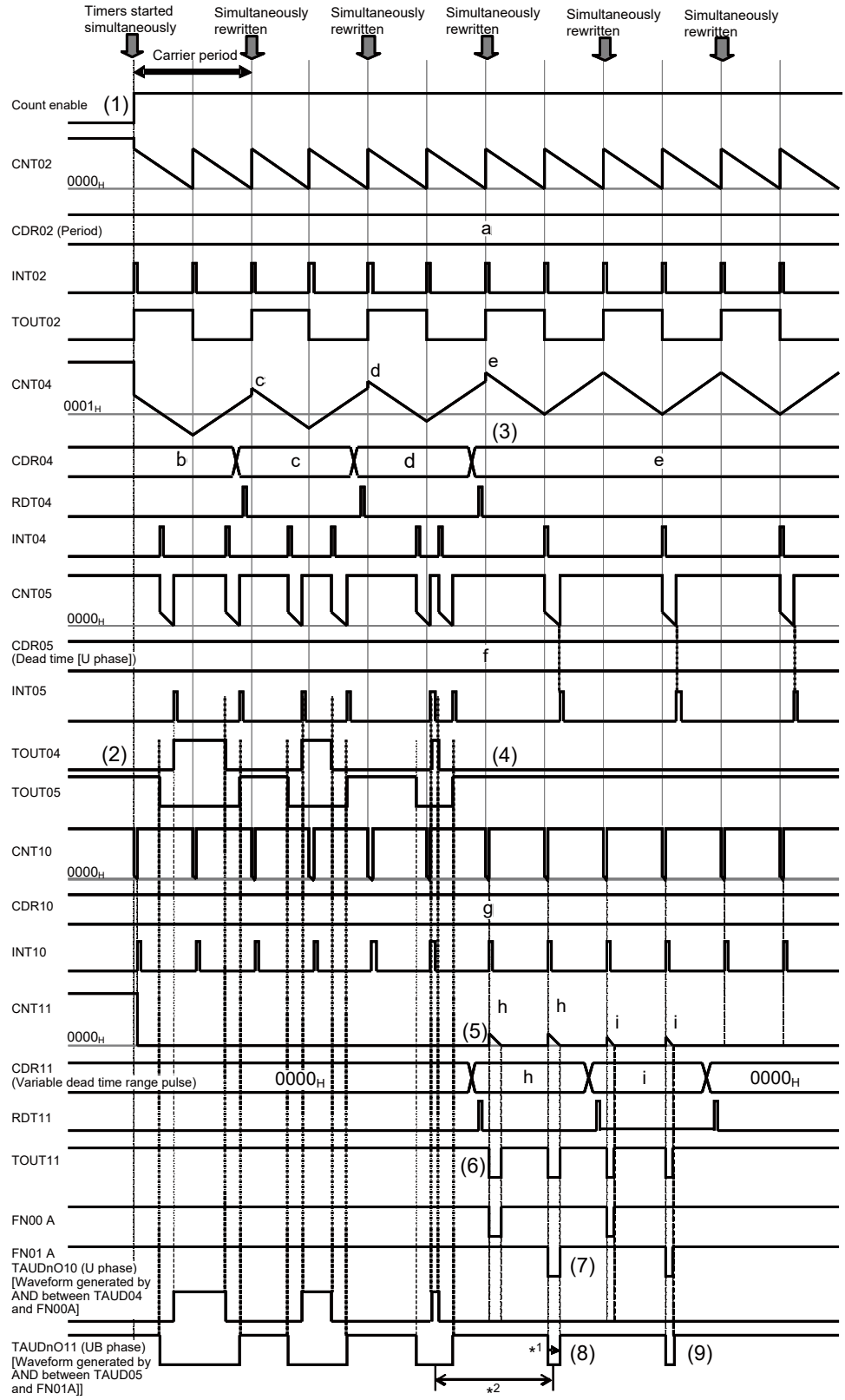


Figure 24.16 Timing Diagram of High Accuracy Triangle Wave PWM (U phase duty cycle = 0%, UB phase duty cycle = 100%) Output with Dead Time (Active High)

(1) With the simultaneous timer start function, the timers to be used are started simultaneously.

(2) TOUT04 and TOUT05 are generated by the triangle wave PWM output function with dead time.

(3) A 0% duty cycle value is set in CDR04 for U phase output.

(4) The setting described in step (3) sets the TOUT04 output to the inactive level and the TOUT05 output to the active level.

(5) To generate the reduced dead time pulse, a value for the reduced dead time pulse width is set in CDR11 when a 0% duty cycle value for U phase output is set in step (3).

(6) CH10 count starts with the effective edge of TOUT02 and INT10 is generated when the counter underflows. CH11 count starts by generation of INT10 and the reduced dead time pulse (TOUT11) of the width set in CDR11 is output.

(7) The reduced dead time pulse (UO1 and UO2) are generated from TOUT02 and TOUT11 in PFN001.

(8) UO1 and UO2 are synthesized with TOUT04 and TOUT05 in FN00 and FN01, and are output as TOPnU (U phase PWM signal) and TOPnUB (UB phase PWM signal).

CAUTION

Since the reduced dead time pulses are sawtooth waves, they expand and contract on one side, unlike the triangle wave pulses, which expand and contract on both sides. Since one-side expansion and contraction applies to the reduced dead time pulses, a one-phase PWM output period in the reduced dead time range is longer by half the width of an inserted reduced dead time pulse.

The following figure shows the timing diagram when U phase duty cycle = 100% and UB phase duty cycle = 0% at active high.

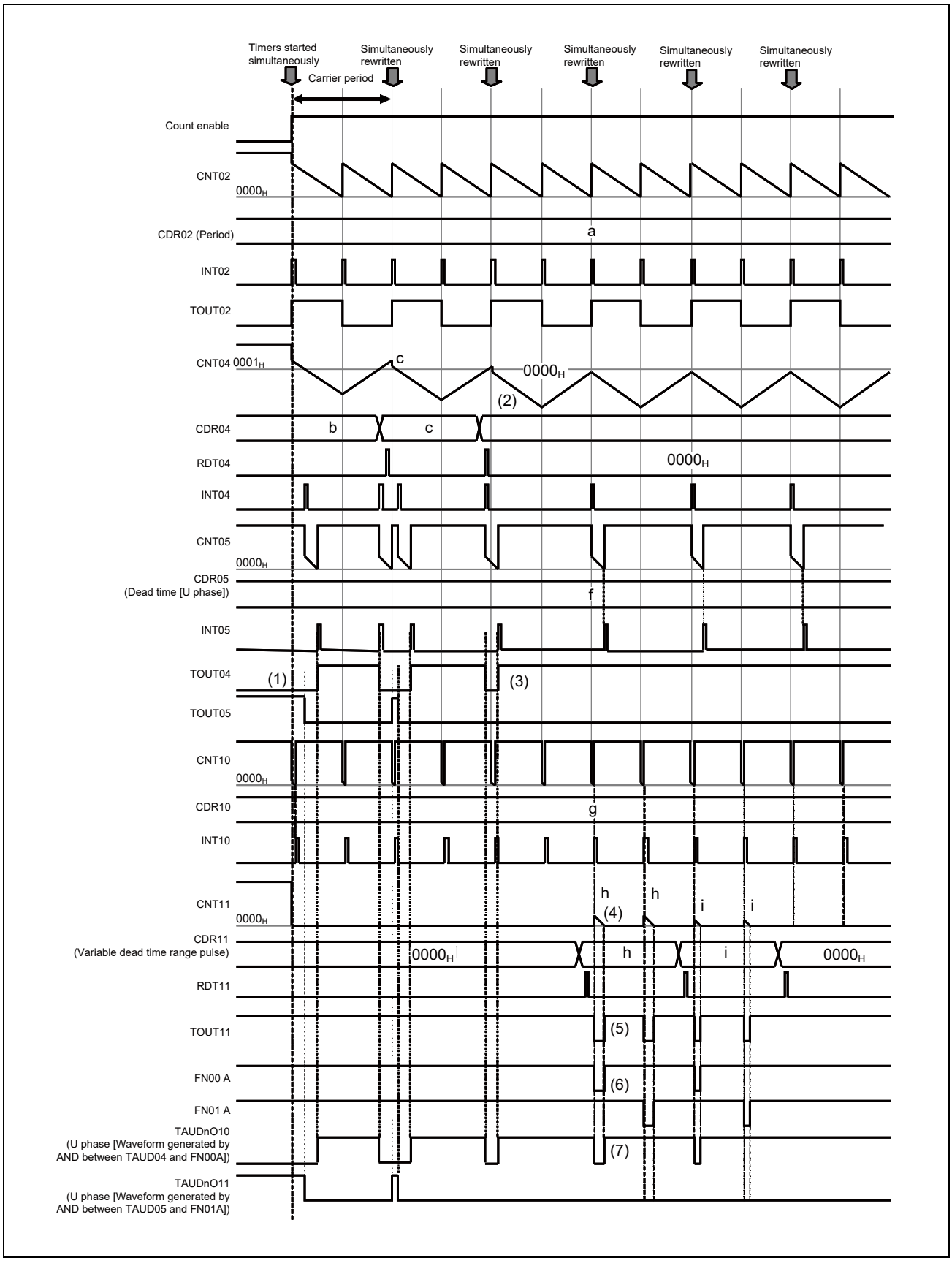


Figure 24.17 Timing Diagram of High Accuracy Triangle Wave PWM (U phase duty cycle = 100%, UB phase duty cycle = 0%) Output with Dead Time (Active High)

(1) With the simultaneous timer start function, the timers to be used are started simultaneously.

(2) TOUT04 and TOUT05 are generated by the triangle wave PWM output function with dead time.

(3) A 100% duty cycle value (CDR04 = 0000_H) is set in CDR04 for U phase output.

(4) TOUT04 outputs the active level and TOUT05 outputs the inactive level.

(5) A value for the reduced dead time pulse width is set in CDR11 one period after setting 100% duty cycle for U phase output.

(6) CH10 count starts with the effective edge of TOUT02 and INT10 is generated when the counter underflows. CH11 count starts by generation of INT10 and the reduced dead time pulse (TOUT11) of the width set in CDR11 is output.

(7) The reduced dead time pulse (UO1 and UO2) are generated from TOUT02 and TOUT11 in PFN001.

(8) UO1 and UO2 are synthesized with TOUT04 and TOU05 in FN00 and FN01, and are output as TOPnU (U phase PWM signal) and TOPnUB (UB phase PWM signal).

CAUTION

As shown in **Figure 24.18**, if a value is set for the variable dead time range pulse width in CDR11 at the same time as a 100% duty cycle value is set for U phase output in CDR04, the variable dead time range pulse affects the last PWM output from TOUT04 (indicated by **Figure 24.18**, (1)) by the amount of time indicated by **Figure 24.18**, (2), which is due to the functional specification. To eliminate this influence, CDR11 must be set one period after setting CDR04.

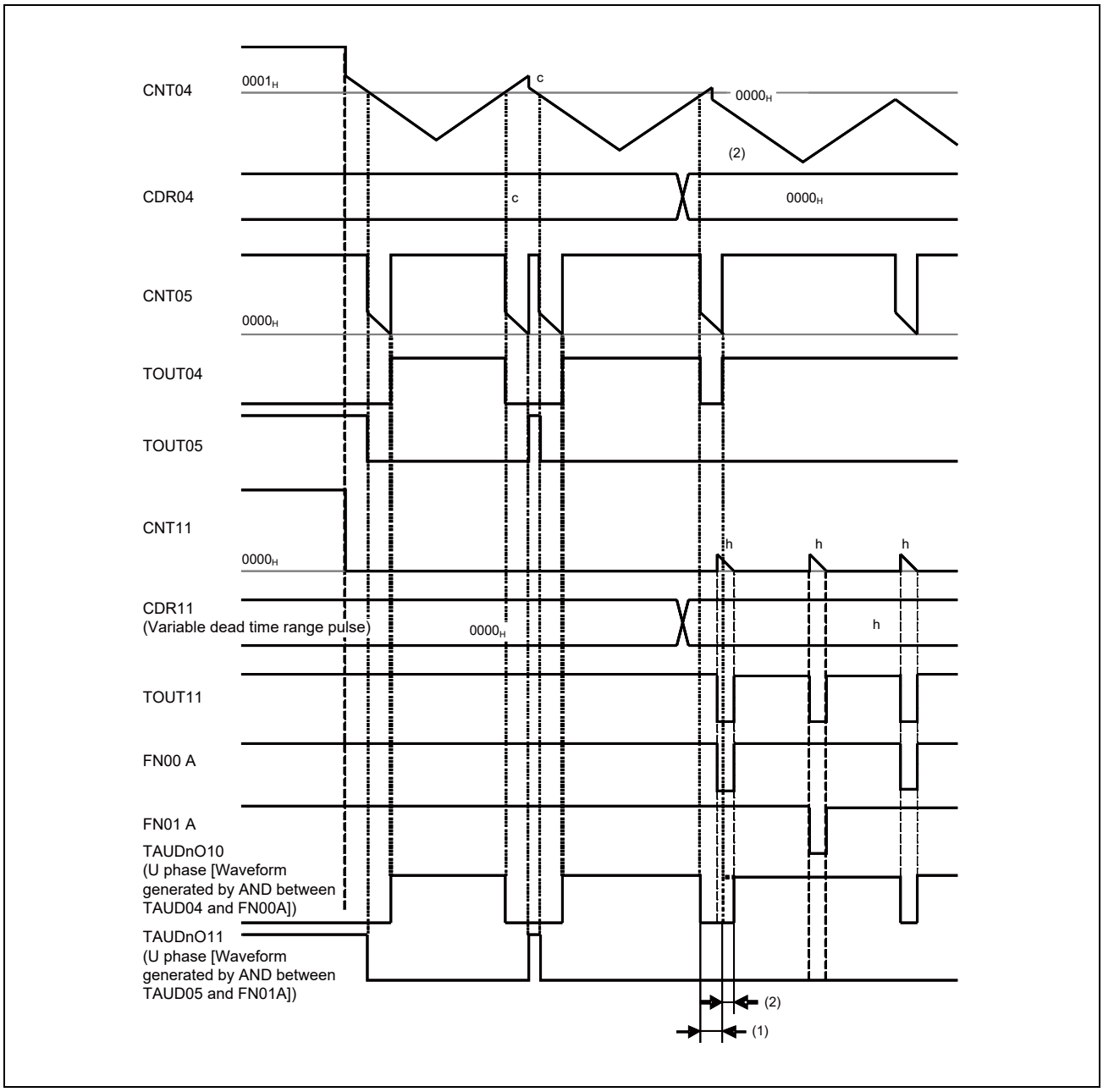


Figure 24.18 Timing Diagram of an Example of Reduced Dead Time Pulse Giving Influence on Triangle Wave PWM Output with Dead Time

The following figure shows the timing diagram when U phase duty cycle = 100% and UB phase duty cycle = 0% at active low.

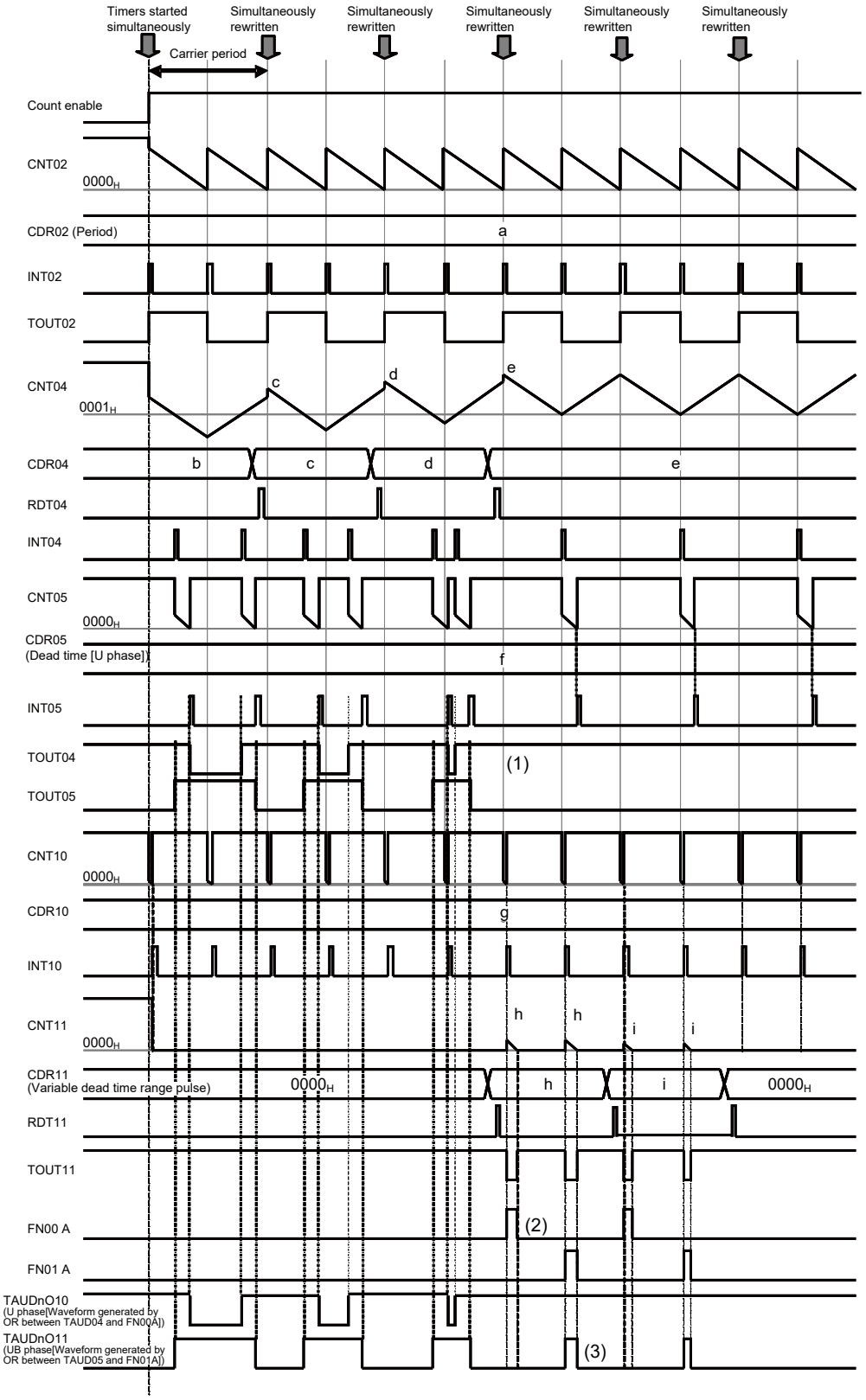


Figure 24.19 Timing Diagram of High Accuracy Triangle Wave PWM (U phase duty cycle = 100%, UB phase duty cycle = 0%) Output with Dead Time (Active Low)

The operation flow from timer operation start to triangle wave PWM output with dead time is same as **Figure 24.19, Timing Diagram of High Accuracy Triangle Wave PWM (U phase duty cycle = 100%, UB phase duty cycle = 0%) Output with Dead Time (Active Low)**, with the difference of the PWM output signal from TOUT04 and TOUT05 are active low.

CAUTION

Set each CDR for the one-shot pulse output function so that the following condition is satisfied.

CDR05 ≥ (CDR10 + CDR11)

If the condition above is not satisfied, the output waveform may be influenced. To minimize the influence, satisfy the condition above, and also fix the value of CDR11 to 0000_H until the reduced dead time pulse is required.

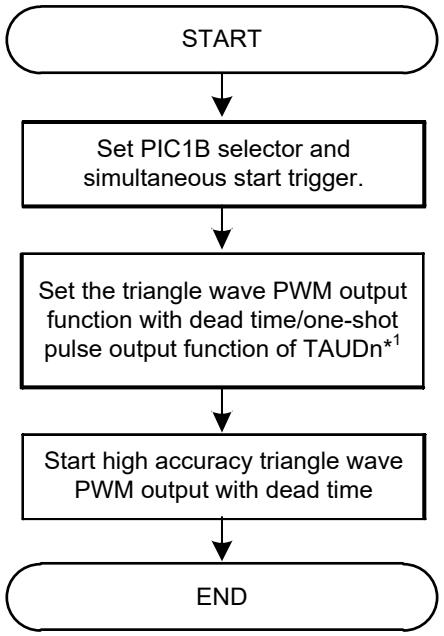
Set the both edges to be detected of TIN10 (TOUT02) as effective, and set TAUDnTOL11 to 1 (active low).

Select the count clock signal (CK0 to 3) of the same clock for TAUDn.

After high accuracy triangle wave PWM output with dead time is started, do not set the value of variable dead time pulse width at the same time as setting a 100% duty cycle value for U phase, V phase, and W phase.

(5) Flow chart

The flow chart of this function is shown in the following figure.



Note 1. For the setting of triangle wave PWM output function with dead time of TAUDn, see **Section 18.4.12.8, Triangle PWM Output Function with Dead Time**, for the setting of one-shot pulse output function, see **Section 18.4.12.2, One-Shot Pulse Output Function**.

Figure 24.20 Setting Flow

24.2.3.5 Delay Pulse Output Function with Dead Time

(1) Overview

This function allows generation of PWM output with the dead time, that is, delayed as specified from the period timing using TAUDn.

With this function, PWM output can be reset in the next period unlike with the function described in **24.2.3.3, PWM Output Function with Dead Time**.

The following table lists the number of channels used.

Table 24.61 Number of TAUDn Channels for Use in Delay Pulse Output with Dead Time

PWM Output with Dead Time	Number of TAUDn Channels
One-phase PWM output (U phase/UB phase)	5 channels (master channel: 1, slave channel: 4)
Two-phase PWM output (U phase/UB phase, V phase/VB phase)	9 channels (master channel: 1, slave channel: 8)
Three-phase PWM output (U phase/UB phase, V phase/VB phase, W phase/WB phase)	13 channels (master channel: 1, slave channel: 12)

Note: Above are examples of combination.

Usages of each channel of TAUDn are listed in the following table. CH2 is used as the master channel of CH3 to 9.

Table 24.62 Usage of TAUDn Channels

TAUDn Channel	U phase/ UB phase	V phase/ VB phase	W phase/ WB phase	Usage
CH0	—	—	—	Not used.
CH1	—	—	—	Not used.
CH2	√	√	√	Carrier period (common to each phase)
CH3	√	√	√	Reserved
CH4	√	—	—	Delay pulse input (U phase/UB phase)
CH5	√	—	—	
CH6	—	√	—	Delay pulse input (V phase/VB phase)
CH7	—	√	—	
CH8	—	—	√	Delay pulse input (W phase/WB phase)
CH9	—	—	√	
CH10	√	—	—	U phase output (TOUT10)
CH11	√	—	—	UB phase output (TOUT11)
CH12	—	√	—	V phase output (TOUT12)
CH13	—	√	—	VB phase output (TOUT13)
CH14	—	—	√	W phase output (TOUT14)
CH15	—	—	√	WB phase output (TOUT15)

Note: √: Used; — : Not used

(2) Configuration

The delay pulse output function with the dead time is realized by using the delay pulse output function/one-phase PWM output function of TAUDn and PIC1B in combination. The following figure shows the block diagram of the delay pulse output function with the dead time.

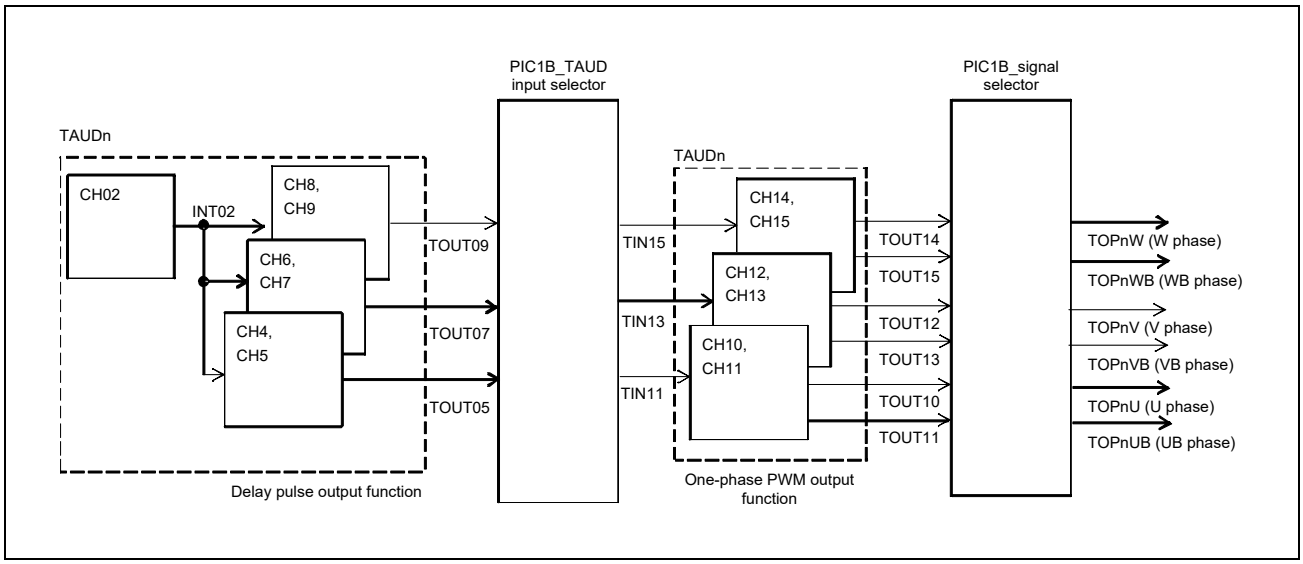


Figure 24.21 Block Diagram of Delay Pulse Output Function with Dead Time

The configuration of this function is described as follows using the PWM output of U phase/UB phase as an example.

- [TAUDn] Delay pulse output function
  CH02, CH04, and CH05 are used in combination. Setting the period, delay value, and the pulse width to CDR02, CDR04, and CDR05, respectively generates the delay pulse output signal (TOUT05).
- [PIC1B_TAUD input selector]
  TOUT05 is selected for output as TIN11.
- [TAUDn] One-phase PWM output function
  CH10 and CH11 are used in combination. Setting the dead time value to CDR11, and inserting dead time into the PWM signal to be input to TIN11 allows TOUT10 (U phase PWM signal) and TOUT11 (UB phase PWM signal) to be output.
- [PIC1B_signal selector]
  TOUT10 and TOUT11 inputs are selected and output to TOPnU and TOPnUB pins, respectively.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

(3) Registers

Block diagram of PIC1B is shown in the following figure.

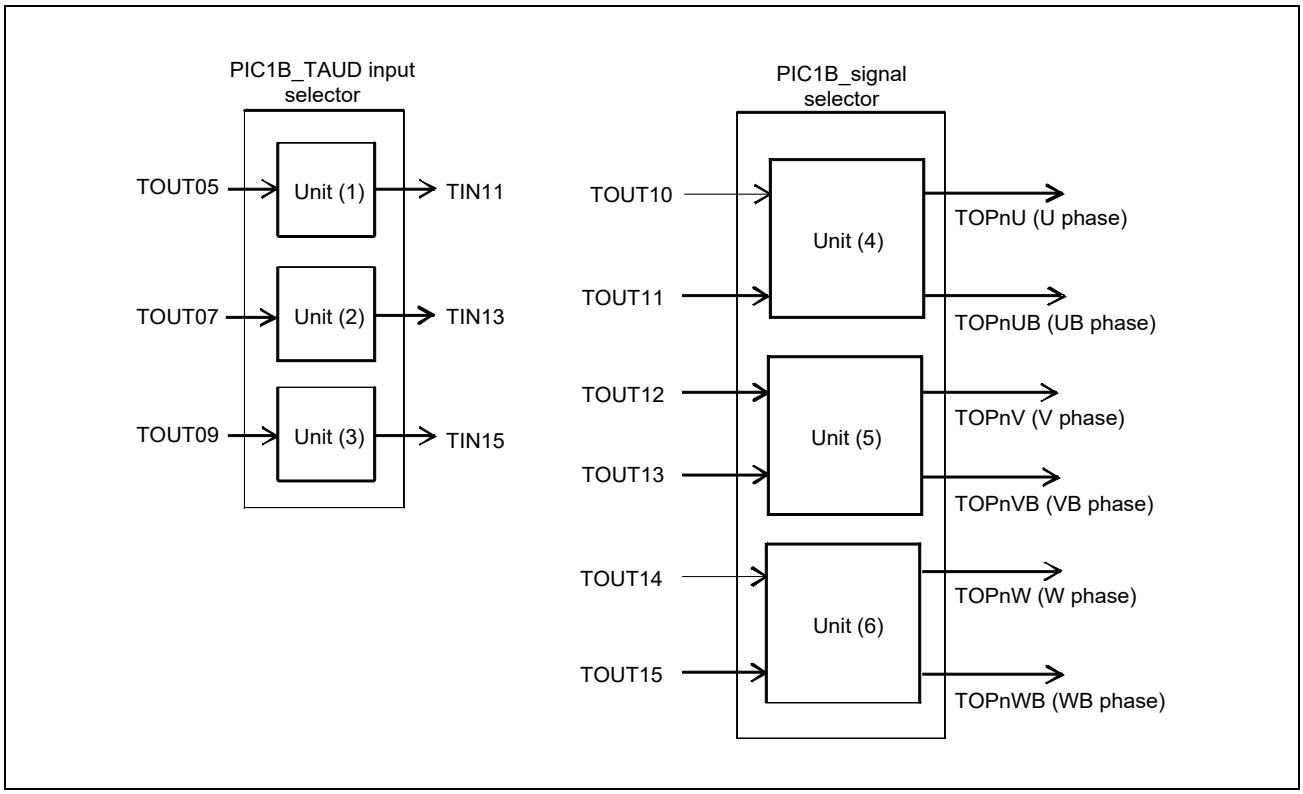


Figure 24.22 Block Diagram of PIC1B

The values of PIC1B registers used in this function are as follows.

U phase/UB phase

The values to output TOUT05 as TIN11 (**Figure 24.22**, unit (1))

PIC1BREG2n2[19:18] = 10_B

PIC1BREG2n2[2] = 0_B

PIC1BTAUDnSEL[23:22] = 00_B

The values to output the TOUT10 and TOUT11 as TOPnU and TOPnUB, respectively (**Figure 24.22**, unit (4))

PIC1BREG2n1[19:16] = 0000_B

PIC1BREG2n3[2:0] = 000_B

PIC1BREG2n3[6:4] = 000_B

V phase/VB phase

The values to output TOUT07 as TIN13 (**Figure 24.22**, unit (2))

PIC1BREG2n2[23:22] = 10_B

PIC1BREG2n2[3] = 0_B

PIC1BTAUDnSEL[27:26] = 00_B

The values to output TOUT12 and TOUT13 as TOPnV and TOPnVB, respectively (**Figure 24.22**, unit (5))

PIC1BREG2n1[23:20] = 0000_B

PIC1BREG2n3[10:8] = 000_B

PIC1BREG2n3[14:12] = 000_B

W phase/WB phase

The values to output TOUT09 as TIN15 (**Figure 24.22**, unit (3))

PIC1BREG2n2[27:26] = 10_B

PIC1BREG2n2[4] = 0_B

PIC1BTAUDnSEL[31:30] = 00_B

The values to output TOUT14 and TOUT15 as TOPnW and TOPnWB, respectively (**Figure 24.22**, unit (6))

PIC1BREG2n1[27:24] = 0000_B

PIC1BREG2n3[18:16] = 000_B

PIC1BREG2n3[22:20] = 000_B

(4) Function

Detail of the function is described using the delay pulse output (U phase/UB phase) as an example.

The following figure shows the timing diagram.

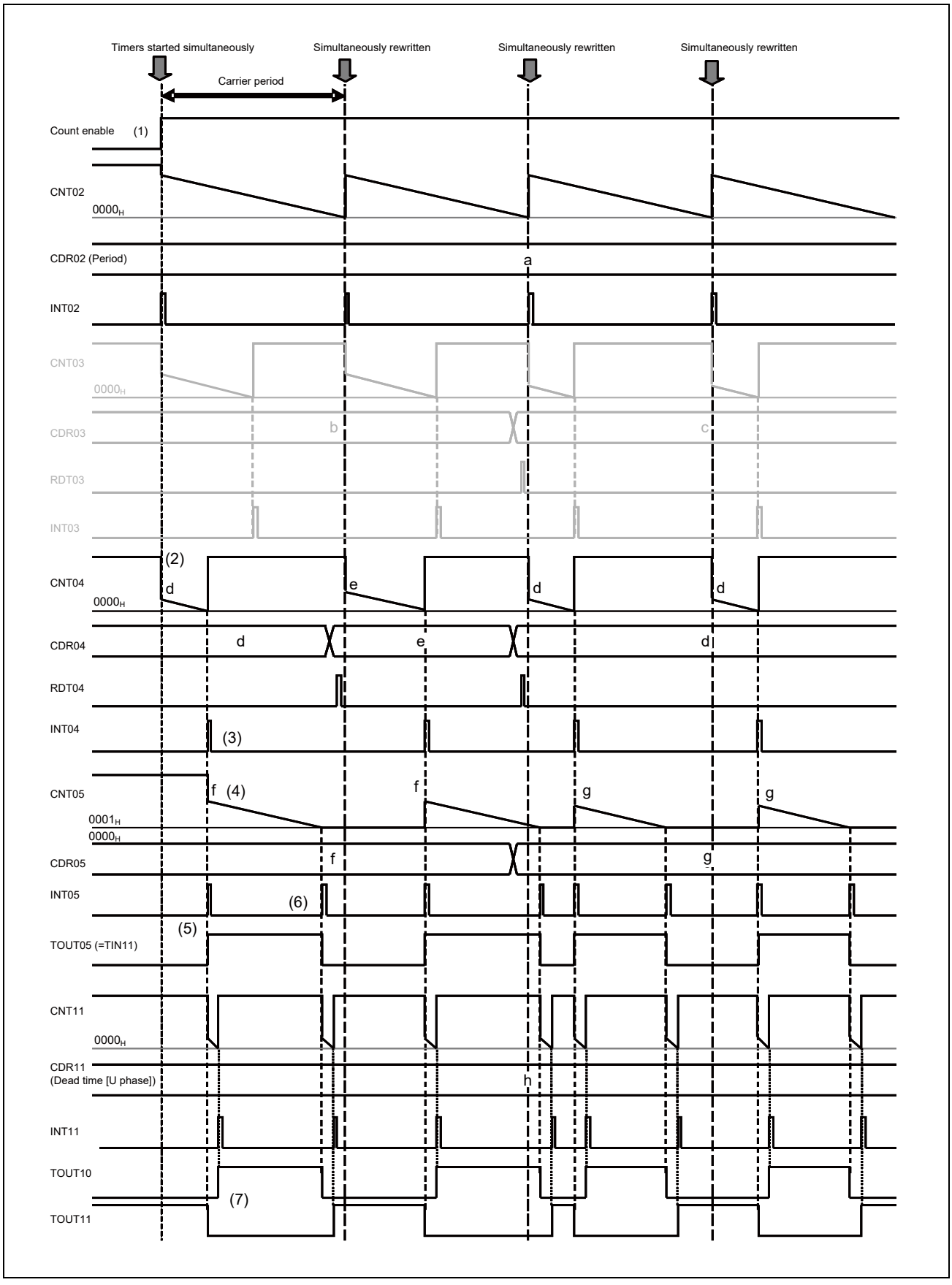


Figure 24.23 Timing Diagram of Delay Pulse Output with Dead Time (U phase/UB phase)

(1) Using the simultaneous start trigger function, the timers to be used are started simultaneously.

(2) For CH04, CH02 underflow allows the set value in CDR04 to be reloaded to CNT04.

(3) CH04 underflow allows the delay timing signal (INT04) to be generated.

(4) INT04 generation allows the set value to be reloaded from CDR05 to CNT05, thus starting the count operation of CH05.

(5) Here, INT05 is generated and the TOUT05 output level changes to the active level.

(6) Upon CH05 underflow, INT05 is generated again and the TOUT05 output level changes to the inactive level. TOUT05 is supplied to TIN11.

(7) The U phase PWM signal (TOUT10) and UB phase PWM signal (TOUT11) with the dead time is generated and output according to the TIN11 edges detected, and output to TOPnU and TOPnUB.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

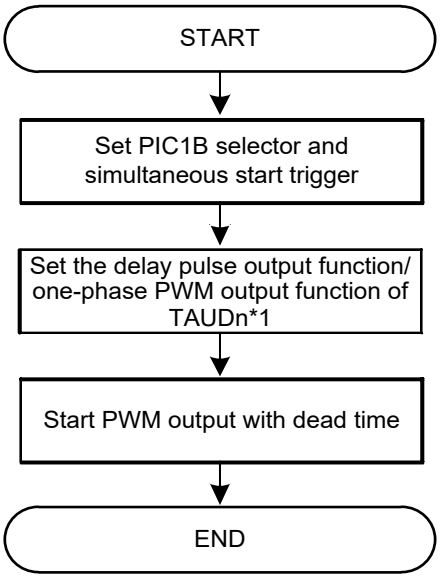
CAUTION

Do not set the delay value which extends over carrier period.

Select the count clock signal of the same clock for TAUDn.

(5) Flow chart

The following flow chart shows the PMW output function with the dead time.



Note 1. For the setting of delay pulse output function of TAUDn, see **Section 18.4.12.4, Delay Pulse Output Function**, for the setting of one-phase PWM output function, see **Section 18.4.12.11, One-Phase PWM Output Function**.

Figure 24.24 Setting Flow

24.2.3.6 Trigger and Pulse Width Measurement Function

(1) Overview

This function allows measurement of trigger periods by inputting the trigger signal output from ENCAn to TAUJ0 and TAUDn.

The following table lists the ENCAn interrupt trigger signals to be measured by each timer and channel.

Table 24.63 Combinations of Measurement Timers and Trigger Signals to be Measured

Timer	Channel	Interrupt Trigger Signal to be Measured
TAUJ0	CH0	ENCAT0IEC
	CH1	ENCAT0IEC
	CH2	ENCAT1IEC
	CH3	ENCAT1IEC
TAUD0	CH0	ENCAT0EQ0 or ENCAT0EQ1
	CH1	ENCAT0EQ1
	CH2	ENCAT0EQ0
TAUD1	CH0	ENCAT1EQ0 or ENCAT1EQ1
	CH1	ENCAT1EQ1
	CH2	ENCAT1EQ0

(2) Configuration

The trigger and pulse width measurement function is realized by using the TAUJ0 and TINm input pulse of TAUDn, and PIC1B in combination. The following figure shows the block diagram of the trigger and pulse width measurement function.

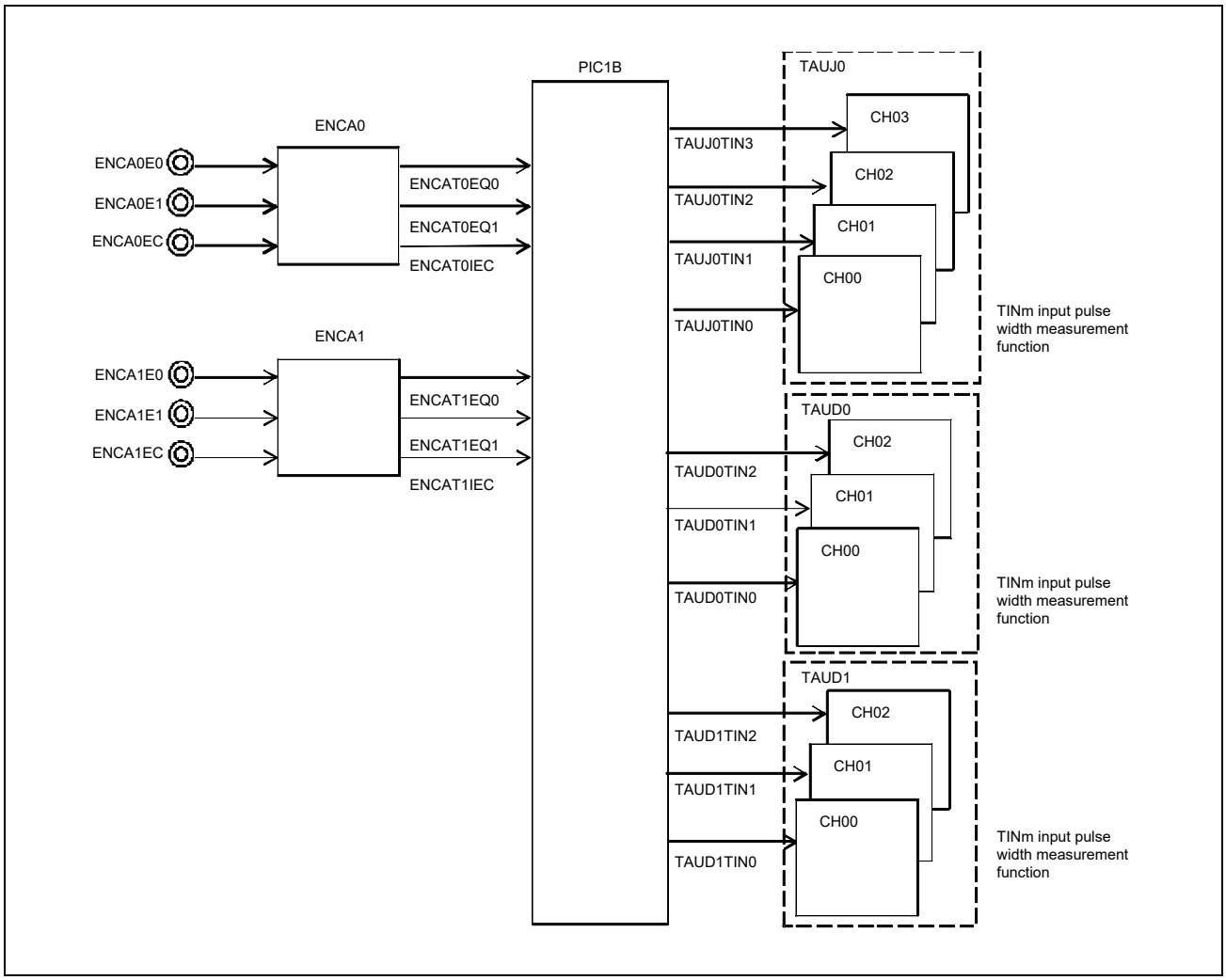


Figure 24.25 Block Diagram of Trigger and Pulse Width Measurement Function

The configuration of this function is described as follows using TAUJ0 CH0 as an example.

- [ENCA0]
  ENCAT0IEC interrupt trigger signal is generated each time ENCA0 timer counter is cleared by input from ENCA0EC pin.
- [PIC1B] Latch and toggle output (DT) circuit
  ENCA0IEC interrupt trigger signal selected by the DT circuit is converted into a level-sensitive toggle signal and output to TAUJ0TIN0.
- [TAUJ0] TINm input pulse width measurement function
  TAUJ0 CH0 is used. TAUJ0CNT0 is captured each time input signal is toggled. The counter is cleared and restarted.

Similar configuration is applied for trigger and pulse width measurement of TAUD0 and TAUD1.

(3) Registers

Block diagram of PIC1B is shown in the following figure.

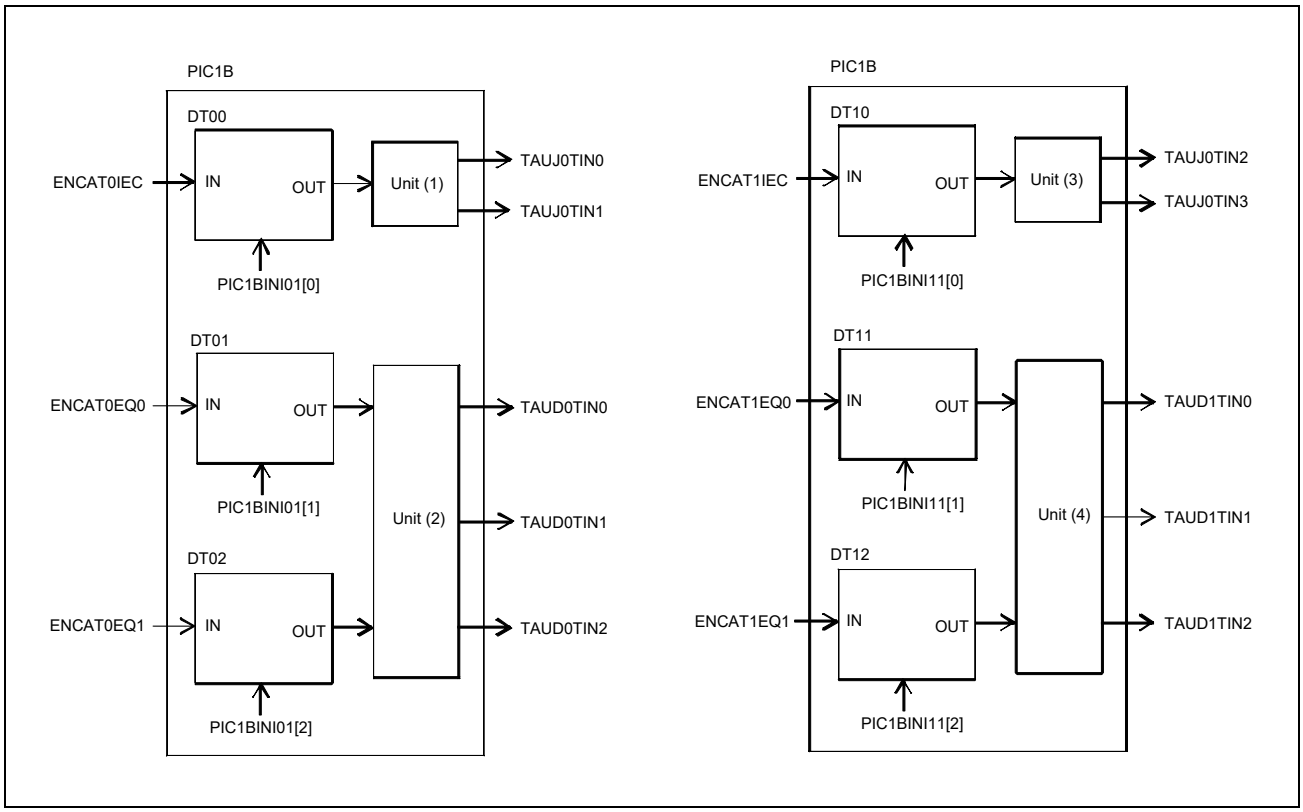


Figure 24.26 Block Diagram of PIC1B

The values of PIC1B registers used in this function are as follows.

ENCA0

- ENCAT0EC trigger and pulse width measurement
  The register values to select the timer which performs ENCAT0EC trigger and pulse width measurement. (**Figure 24.26**, unit (1))

Table 24.64 Setting Values

Register Value			
PIC1BREG31			
1	0	TAUJ0.TIN00	TAUJ0.TIN01
0	0	Not selected	
0	1	ENCAT0IEC	—
1	0	—	ENCAT0IEC
1	1	ENCAT0IEC	ENCAT0IEC

Note: Write 0 (value after reset) to PIC1BREG30[22,17:16].

- ENCAT0EQ0 and ENCAT0EQ1 trigger and pulse width measurement
  The register values to select the timer which performs ENCAT0EQ0 and ENCAT0EQ1 trigger and pulse width measurement. (**Figure 24.26**, unit (2))

Table 24.65 Setting Values

Register Value										
PIC1BREG31										
13	12	11	10	9	8	7	6	TAUD0.TIN00	TAUD0.TIN01	TAUD0.TIN02
0	0	0	0	0	0	0	0	Not selected		
0	0	1	1	0	0	0	1	ENCAT0EQ0	ENCAT0EQ1	—
0	1	0	0	0	0	0	1	ENCAT0EQ1	—	ENCAT0EQ0
0	1	1	0	0	0	0	1	ENCAT0EQ1	ENCAT0EQ1	ENCAT0EQ0
0	1	1	1	0	0	0	1	ENCAT0EQ0	ENCAT0EQ1	ENCAT0EQ0

Note: Do not set the values other than the settings listed above for this function.
Write 0 (value after reset) to PIC1BTAUD0SEL[5:0] and PIC1BREG30[22, 17:16, 1:0].

- Enables initialization of the DT02 to DT00 circuits
  The register values to enable initialization of the DT02 to DT00 circuits.
  PIC1BINI01[2:0] = 111_B (initialized)

ENCA1

- ENCAT1EC trigger and pulse width measurement
  The register values to select the timer which performs ENCAT1EC trigger and pulse width measurement. (**Figure 24.26**, unit (3))

Table 24.66 Setting Values

Register Value			
PIC1BREG31			
4	3	TAUJ0.TIN02	TAUJ0.TIN03
0	0	Not selected	
0	1	ENCAT1IEC	—
1	0	—	ENCAT1IEC
1	1	ENCAT1IEC	ENCAT1IEC

Note: Write 0 (value after reset) to PIC1BREG30[20:19,11:10].

- ENCAT1EQ0 and ENCAT1EQ1 trigger and pulse width measurement
  The register values to select the timer which performs ENCAT1EQ0 and ENCAT1EQ1 trigger and pulse width measurement. (**Figure 24.26**, unit (4))

Table 24.67 Setting Values

Register Value										
PIC1BREG31										
22	21	20	19	18	17	16	15	TAUD1.TIN00	TAUD1.TIN01	TAUD1.TIN02
0	0	0	0	0	0	0	0	Not selected		
0	0	1	1	0	0	0	1	ENCAT1EQ0	ENCAT1EQ1	—
0	1	0	0	0	0	0	1	ENCAT1EQ1	—	ENCAT1EQ0
0	1	1	0	0	0	0	1	ENCAT1EQ1	ENCAT1EQ1	ENCAT1EQ0
0	1	1	1	0	0	0	1	ENCAT1EQ0	ENCAT1EQ1	ENCAT1EQ0

Note: Do not set the values other than the settings listed above for this function.
Write 0 (value after reset) to PIC1BTAUD1SEL[5:0] and PIC1BREG30[20:19, 9:6].

- Enable initialization of DT12 to DT10 circuits
  The register values to enable initialization of the DT12 to DT10 circuits.
  PIC1BINI11 [2:0] = 111_B (initialized)

(4) Function

Detail of the function is described here.

The following figure shows the timing diagram.

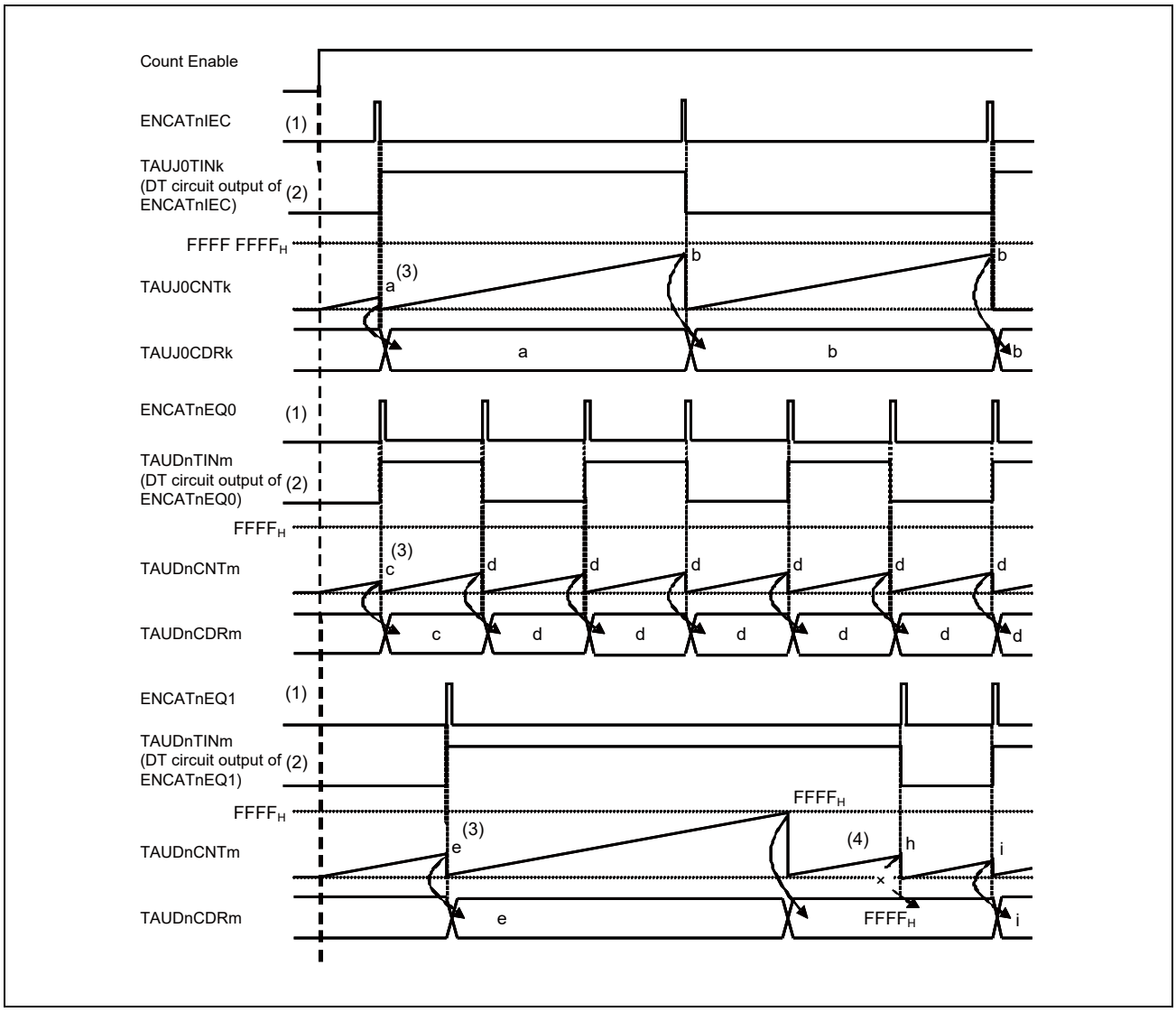


Figure 24.27 Timing Diagram of Trigger and Pulse Width Measurement

(1) Using the simultaneous start trigger function, the timers to be used are started simultaneously.

(2) The interrupt trigger signal output from ENCAn is converted to the level-sensitive toggle signal by the DT circuit and is output to TINm of TAUJ0 and TINm of TAUDn.

(3) The CNTm value is captured into CDRm on the TINm toggle timing and cleared.

(4) When an overflow occurs, the greatest count value ($FFFF_H$ for TAUDn and FFFF $FFFF_H$ for TAUJ0) is captured and the counter is cleared at the same time. The count value is not captured on the first trigger after the overflow. (When TAUDnCMORm.TAUDnCOS[1] = 1_B)

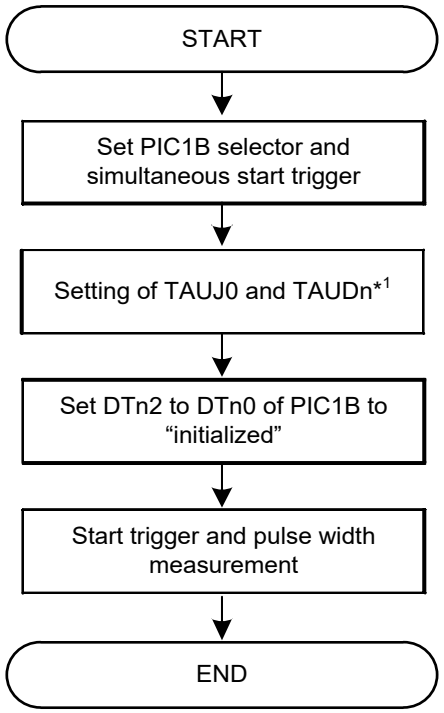
CAUTION

Operation at an overflow varies depending on the setting of TAUJ and TAUD. For the detail of the TAUJ setting, see **Section 19.4.9.3, TAUJTTINm Input Pulse Interval Measurement Function**, for the detail of the TAUD setting, see **Section 18.4.9.7, TAUDTTINm Input Pulse Interval Measurement Function**. In this function, set the effective edge to be detected by TAUJ0 and TINm of TAUDn as both (rising edge and falling edge).

(5) Flow chart

The following figure shows the setting flow of this function.

This flow chart can be set at both during ENCAn operation and while waiting for simultaneous start trigger.



Note 1. For the setting of TAUDn, see **Section 18.4.9.7, TAUDTTINm Input Pulse Interval Measurement Function**, for the setting of TAUJ0, see **Section 19.4.9.3, TAUJTTINm Input Pulse Interval Measurement Function**.

Figure 24.28 Setting Flow

The ENCAn registers to use this function are as follows.

ENCAnCTL[15:0] = xx00_0000_x00x_xxxx$_B$

ENCAnIOC0[7:0] = 0000_0000$_B$

ENCAnIOC1[7:0] = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 23, Encoder Timer (ENCA)**.

24.2.3.7 Encoder Capture Trigger Select Function

(1) Overview

The function selects any of the ADCCnTRGm (ADCCn conversion start trigger m), TAUDnTINTm (TAUDn-CHm interrupt request signal), and ENCAnI1 (signal 1 of the ENCAn external pin input).

(2) Configuration

The encoder capture trigger select function is realized by using ADCCnTRGm (ADCCn conversion start trigger m), TAUDnTINTm (TAUDn CHm interrupt request signal), ENCAnI1 (ENCAn external pin input 1 signal), and PIC1B.

Block diagram of this function is shown in the following figure.

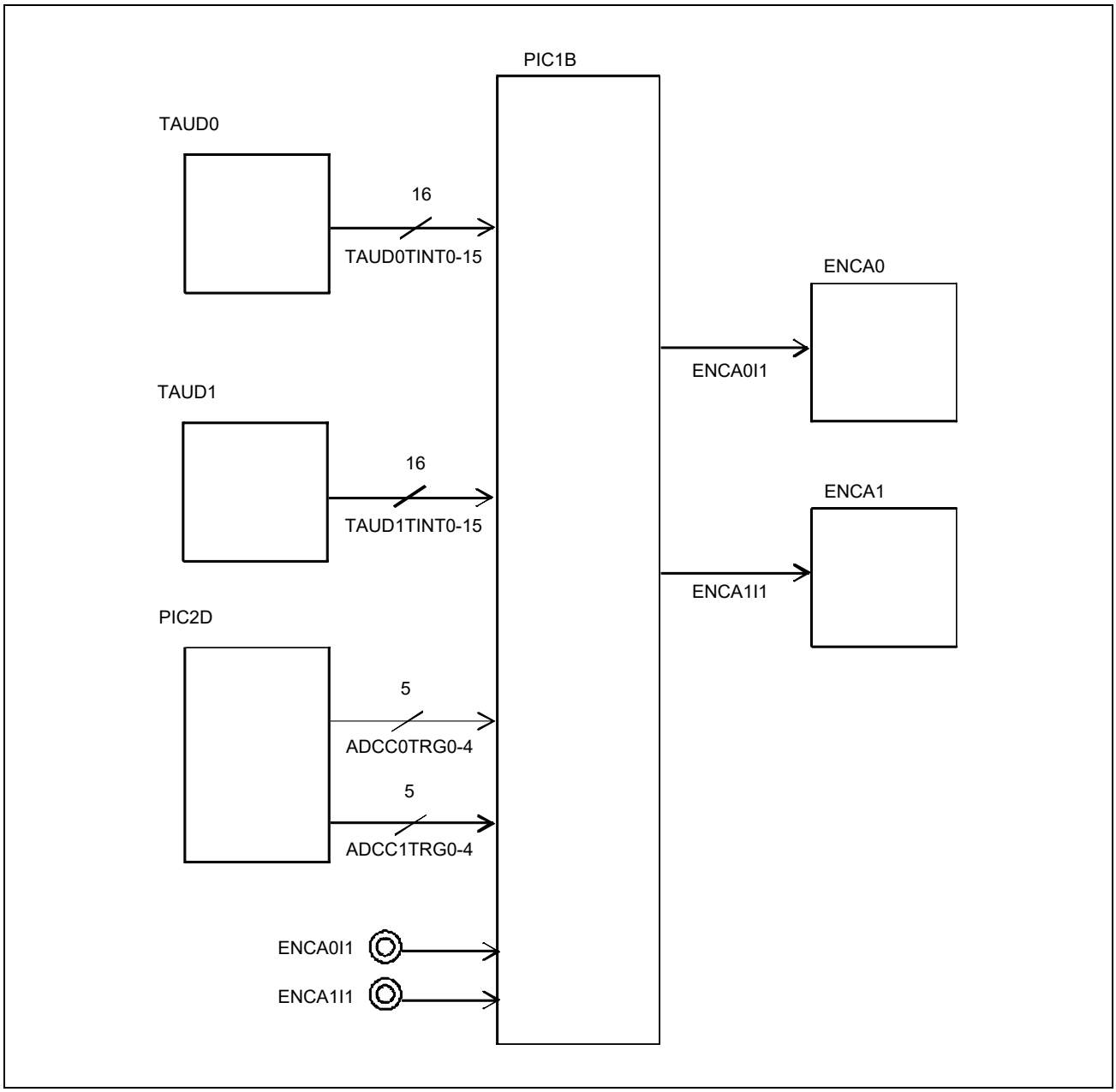


Figure 24.29 Block Diagram of Encoder Capture Trigger Select Function

An example of selecting CH0 of TAUD0 as a capture trigger input of ENCA0 is described as follows.

PIC1BENCSEL400[7] = 1_B

PIC1BENCSEL400[3:0] = 0000_B

PIC1BREG30[18] = 1_B

PIC1BREG30[5:2] = 0000_B

(3) Registers

Block diagram of PIC1B is shown in the following figure.

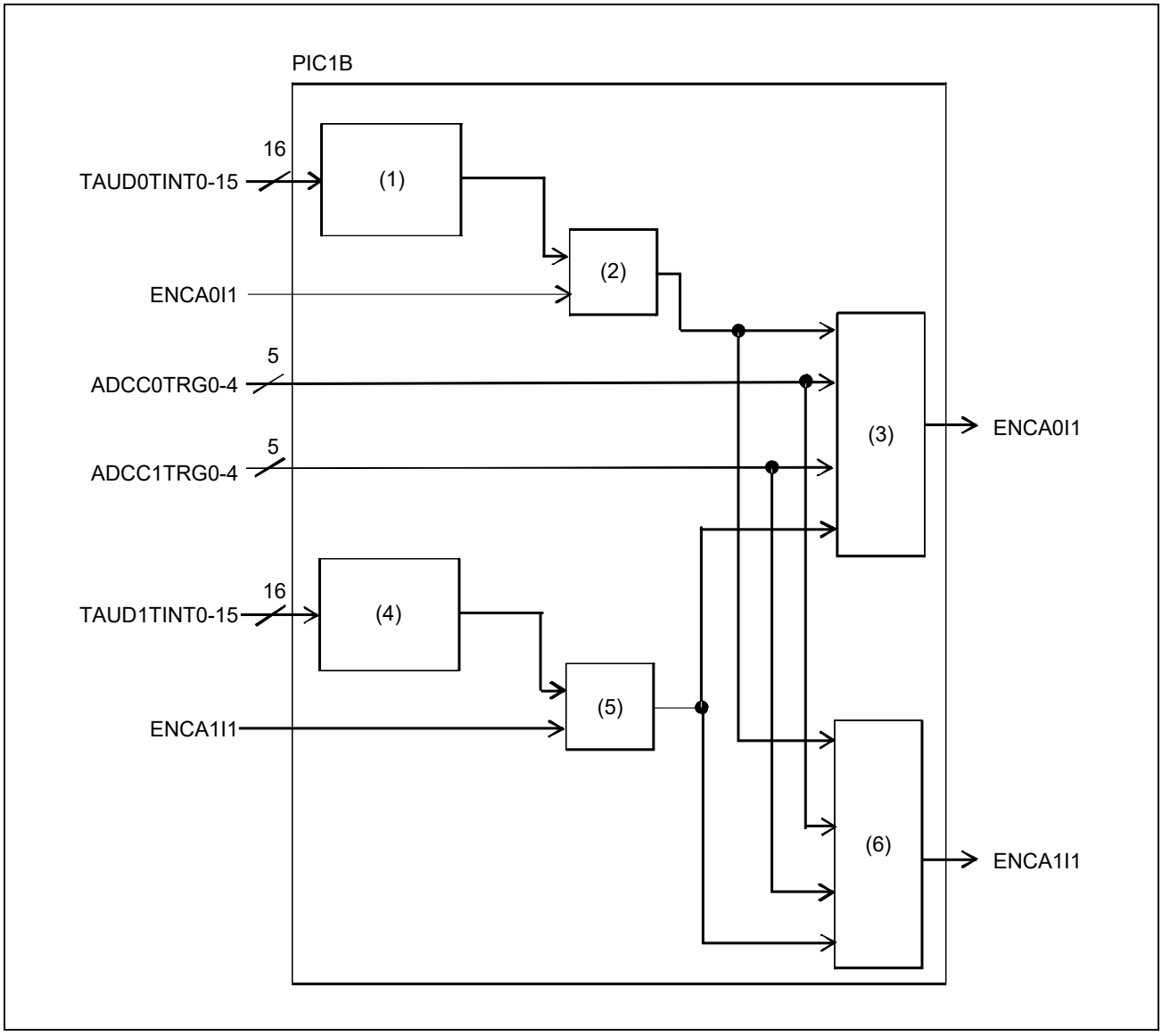


Figure 24.30 Block Diagram of PIC1B

The values of PIC1B registers used in this function are as follows.

ENCA0

- (1) TAUD0TINTm selection
  The register values to select TAUD0TINTm. Set 1 to the PIC1BENCSEL400[7] to select TAUD0TINTm.

Table 24.68 Setting Values

Register Value					
PIC1BENCSEL400					
7	3	2	1	0	Output
1	0	0	0	0	INTTAUD0I0
1	0	0	0	1	INTTAUD0I1
1	0	0	1	0	INTTAUD0I2
1	0	0	1	1	INTTAUD0I3
1	0	1	0	0	INTTAUD0I4
1	0	1	0	1	INTTAUD0I5
1	0	1	1	0	INTTAUD0I6
1	0	1	1	1	INTTAUD0I7
1	1	0	0	0	INTTAUD0I8
1	1	0	0	1	INTTAUD0I9
1	1	0	1	0	INTTAUD0I10
1	1	0	1	1	INTTAUD0I11
1	1	1	0	0	INTTAUD0I12
1	1	1	0	1	INTTAUD0I13
1	1	1	1	0	INTTAUD0I14
1	1	1	1	1	INTTAUD0I15

- (2) TAUD0TINTm and ENCA0I1 pins selection
  The register values to select either the output (1) or ENCA0I1.

Table 24.69 Setting Values

Register Value	
PIC1BREG30	
18	Output
1	Output (1)
0	ENCA0I1

- (3) ENCA0I1 selection

  The register values to select any of the output (2), output (5), ADCC0TRG0 to 4, and ADCC1TRG0 to 4.

Table 24.70 Setting Values

Register Value				
PIC1BREG30				
5	4	3	2	ENCA0I1
0	0	0	0	Output (2)
0	0	0	1	Output (5)
0	0	1	0	ADCC0TRG4
0	0	1	1	ADCC0TRG3
0	1	0	0	ADCC0TRG2
0	1	0	1	ADCC0TRG1
0	1	1	0	ADCC0TRG0
0	1	1	1	ADCC1TRG4
1	0	0	0	ADCC1TRG3
1	0	0	1	ADCC1TRG2
1	0	1	0	ADCC1TRG1
1	0	1	1	ADCC1TRG0

Note: Do not set the values other than the settings listed above for this function.

ENCA1

- (4) TAUD1TINTm selection

  The register values to select TAUD1TINTm. Set 1 to the PIC1BENCSEL410[7] to select TAUD1TINTm.

Table 24.71 Setting Values

Register Value					
PIC1BENCSEL410					
7	3	2	1	0	Output
1	0	0	0	0	INTTAUD1I0
1	0	0	0	1	INTTAUD1I1
1	0	0	1	0	INTTAUD1I2
1	0	0	1	1	INTTAUD1I3
1	0	1	0	0	INTTAUD1I4
1	0	1	0	1	INTTAUD1I5
1	0	1	1	0	INTTAUD1I6
1	0	1	1	1	INTTAUD1I7
1	1	0	0	0	INTTAUD1I8
1	1	0	0	1	INTTAUD1I9
1	1	0	1	0	INTTAUD1I10
1	1	0	1	1	INTTAUD1I11
1	1	1	0	0	INTTAUD1I12
1	1	1	0	1	INTTAUD1I13
1	1	1	1	0	INTTAUD1I14
1	1	1	1	1	INTTAUD1I15

- (5) TAUD1TINTm and ENCA1I1 pins selection
  The register values to select either the output (4) or ENCA1I1.

Table 24.72 Setting Values

Register Value	
PIC1BREG30	
21	Output
1	Output (4)
0	ENCA1I1

- (6) ENCA1I1 selection
  The register values to select any of the output (2), output (5), ADCC0TRG0 to 4, and ADCC1TRG0 to 4.

Table 24.73 Setting Values

Register Value				
PIC1BREG30				
15	14	13	12	ENCA1I1
0	0	0	0	Output (5)
0	0	0	1	Output (2)
0	0	1	0	ADCC0TRG4
0	0	1	1	ADCC0TRG3
0	1	0	0	ADCC0TRG2
0	1	0	1	ADCC0TRG1
0	1	1	0	ADCC0TRG0
0	1	1	1	ADCC1TRG4
1	0	0	0	ADCC1TRG3
1	0	0	1	ADCC1TRG2
1	0	1	0	ADCC1TRG1
1	0	1	1	ADCC1TRG0

Note: Do not set the values other than the settings listed above for this function.

(4) Function

Detail of the function is described with an example of selecting TAUDnTINTm as a capture trigger signal.

The following figure shows the timing diagram.

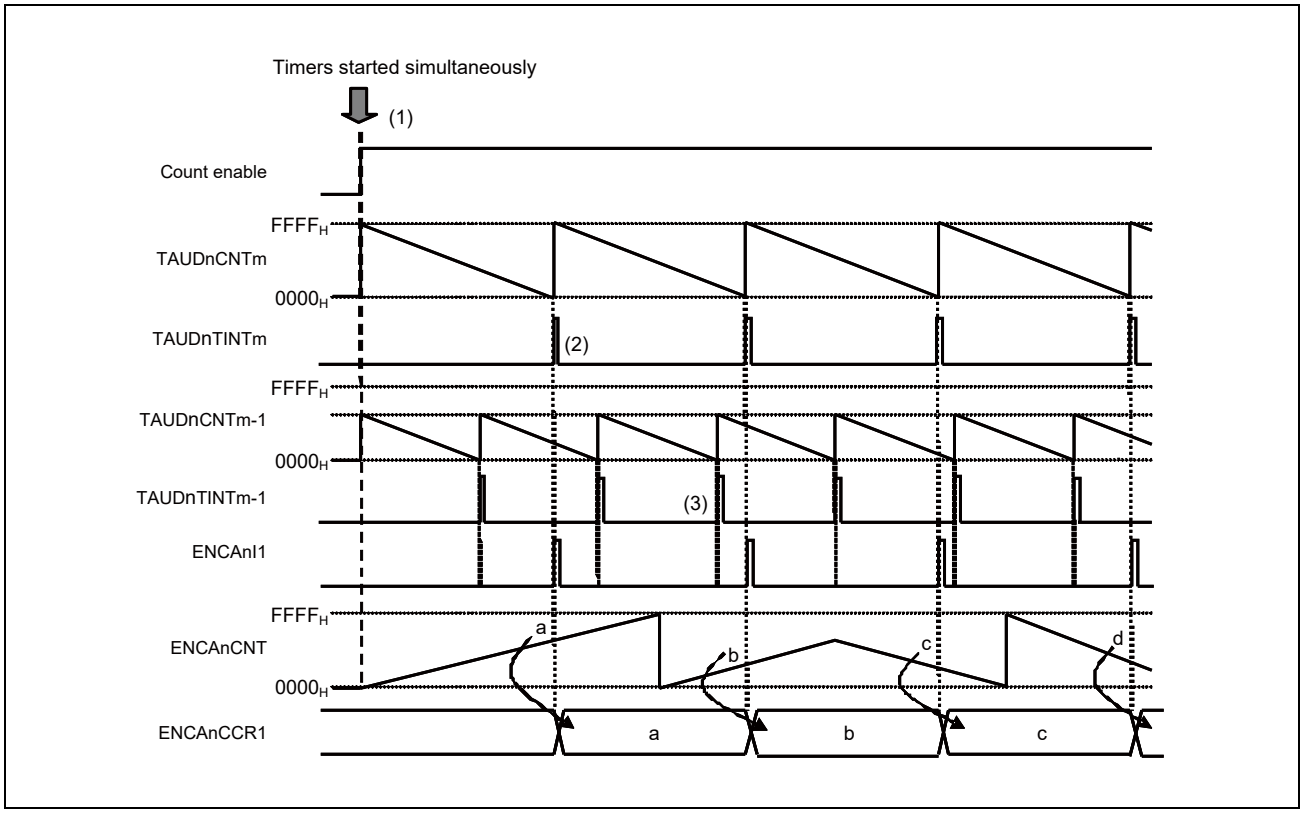


Figure 24.31 Timing Diagram of Encoder Capture Trigger Select Function (TAUDnTINTm)

(1) Using the simultaneous start trigger function, the timers to be used are started simultaneously.

On generation of an effective edge of TAUDnTINTm, ENCAn captures ENCAnCNT.

Do not select ENCAn interrupt trigger signal (INTENCATnI1) as the ADCCn trigger described in **Section 24.3.3.1, ADCC Trigger Select Function**, the correct operation cannot be performed because the following loop occurs: ADCCnTRG1 generation → ENCAn capture operation → INTENCAnI1 generation by capture operation → ADCCnTRG1 generation.

The following figure shows a timing chart of the loop paths of PIC1B, PIC2D, and ENCAn.

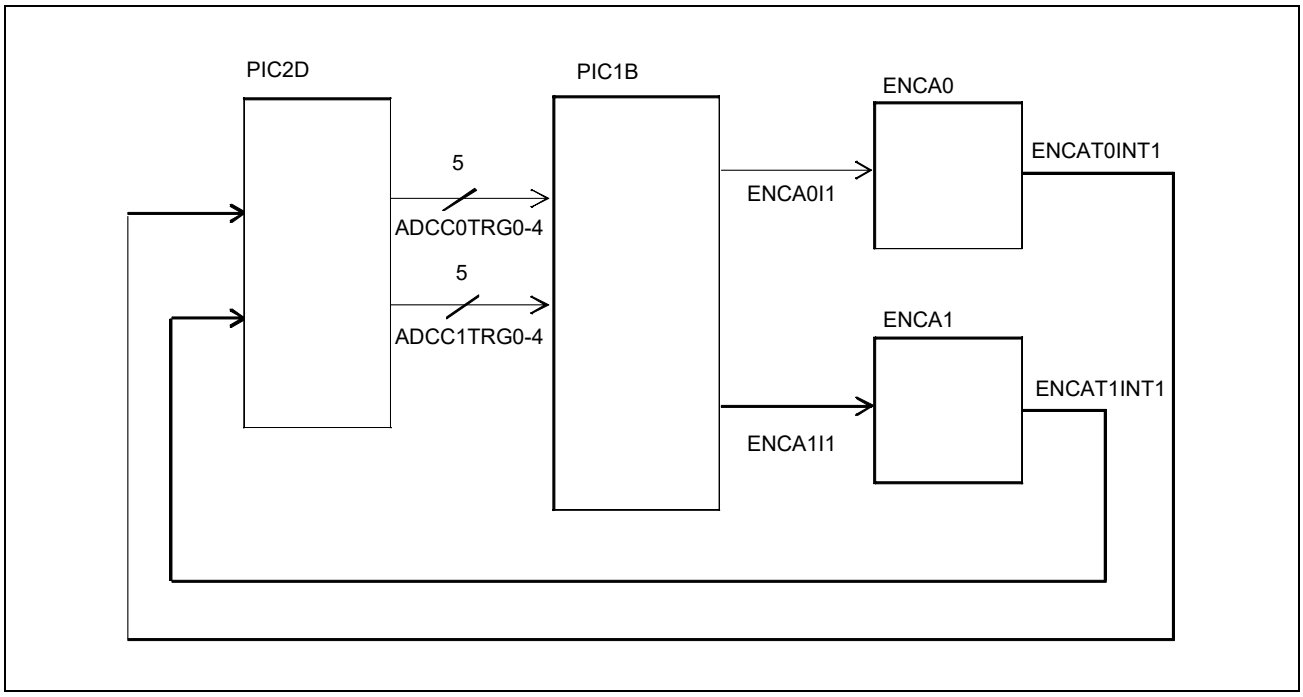


Figure 24.32 Timing Chart of Loop Paths of PIC1B, PIC2D, and ENCAn

(5) Flow chart

Select the encoder capture trigger before starting the encoder timer.

ENCAn settings to use this function are described as follows.

ENCAnCTL[15:0] = 0000_001x_0 00x_xxxx$_B$

ENCAnIOC0[7:0] = 0000_01xx$_B$

ENCAnIOC1[7:0] = (set any value)

“x” can be set arbitrarily. For the registers specifications, see **Section 23, Encoder Timer (ENCA)**.

24.2.3.8 Two-Phase Encoder Control Function (Control Method 1)

(1) Overview

This function allows switching of the output patterns of the motor control function (TSG3n) using the two-phase encoder control function (ENCAn).

(2) Configuration

Switching of output pattern in 120-DC mode by encoder result is realized by using ENCAn and TSG3n, and PIC1B in combination. The following figure describes the block diagram of two-phase encoder control function (control method 1).

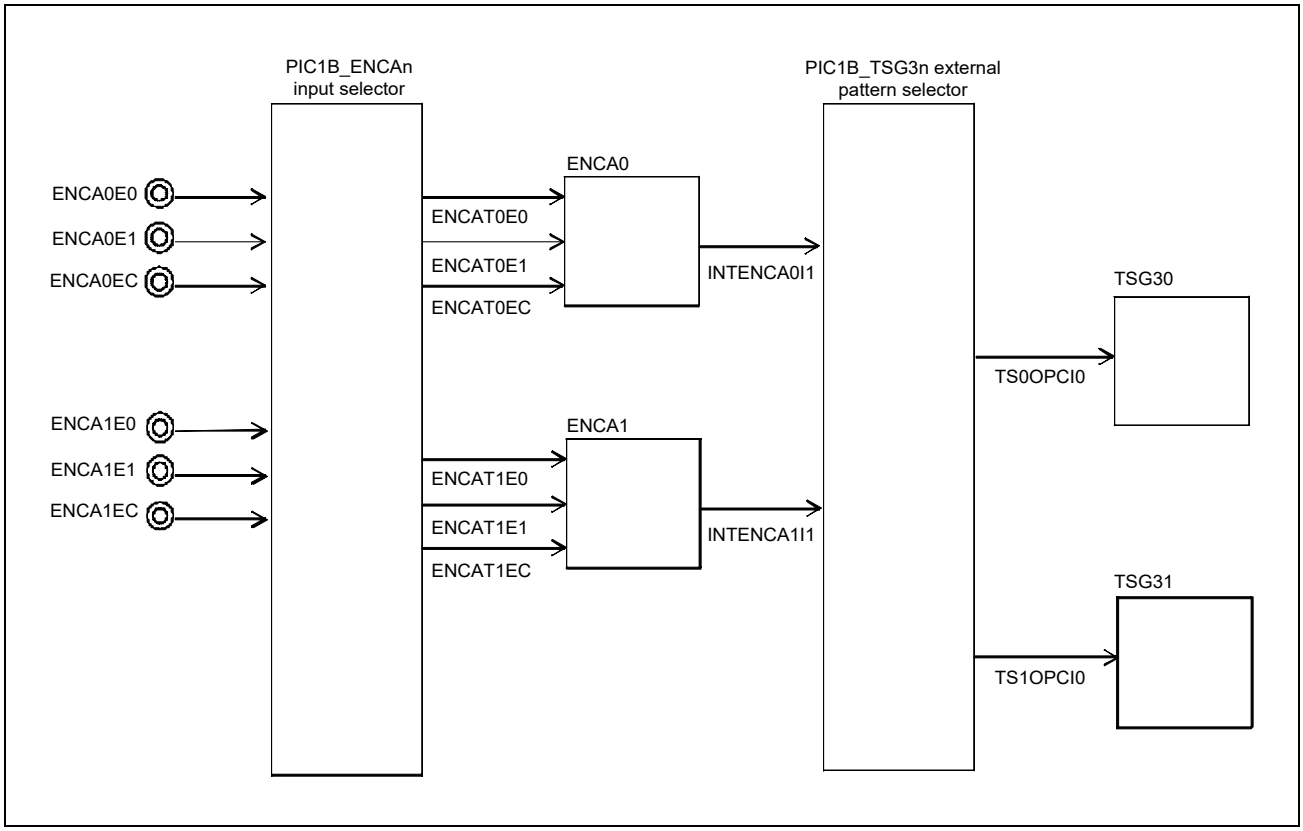


Figure 24.33 Block Diagram of Two-Phase Encoder Control Function (Control Method 1)

The configuration of two-phase encoder control function (control method 1) is described as follows.

- [PIC1B_ENCAn input selector]
  ENCAnE0, ENCAnE1, and ENCAnEC pin inputs are selected and output as ENCATnE0, ENCATnE1, and ENCATnEC.
- [ENCAn]
  INTENCAnI1 is output by two-phase encoder processing.
- [PIC1B_TSG3n external pattern selector]
  INTENCAnI1 is selected and output to TS0OPCI0 or TS1OPCI0.
- [TSG3n]
  Output pattern in 120-DC mode is switched by TSG3nOPCI0.

(3) Registers

The values of PIC1B registers used in this function are as follows.

PIC1B_ENCAn input selector

The register values to output ENCAn pin inputs (ENCAnE0, ENCAnE1, and ENCAnEC) as ENCATnE0, ENCATnE1, and ENCATnEC.

PIC1BREG30[22] = 0_B

PIC1BREG30[20:19] = 00_B

PIC1BREG30[17:16] = 00_B

PIC1BREG30[11:6] = 000000_B

PIC1BREG30[1:0] = 00_B

PIC1B_TSG3n external pattern selection

The register values to select the interrupt request signal to be input as TSG30 external pattern.

Table 24.74 Setting Values

Register Value				
PIC1BREG50				
10	8	6	5	TS0OPCI0
X	0	0	1	INTENCA0I1
0	X	1	0	INTENCA1I1

Note: Do not set the values other than the settings listed above for this function.

The register values to select the interrupt request signal to be input as TSG31 external pattern.

Table 24.75 Setting Values

Register Value				
PIC1BREG51				
10	8	6	5	TS1OPCI0
X	0	0	1	INTENCA0I1
0	X	1	0	INTENCA1I1

Note: Do not set the values other than the settings listed above for this function.

(4) Function

Detail of the function is described using the two-phase encoder control function (method 1) at up count (normal rotation) as an example.

The following figure shows the timing diagram.

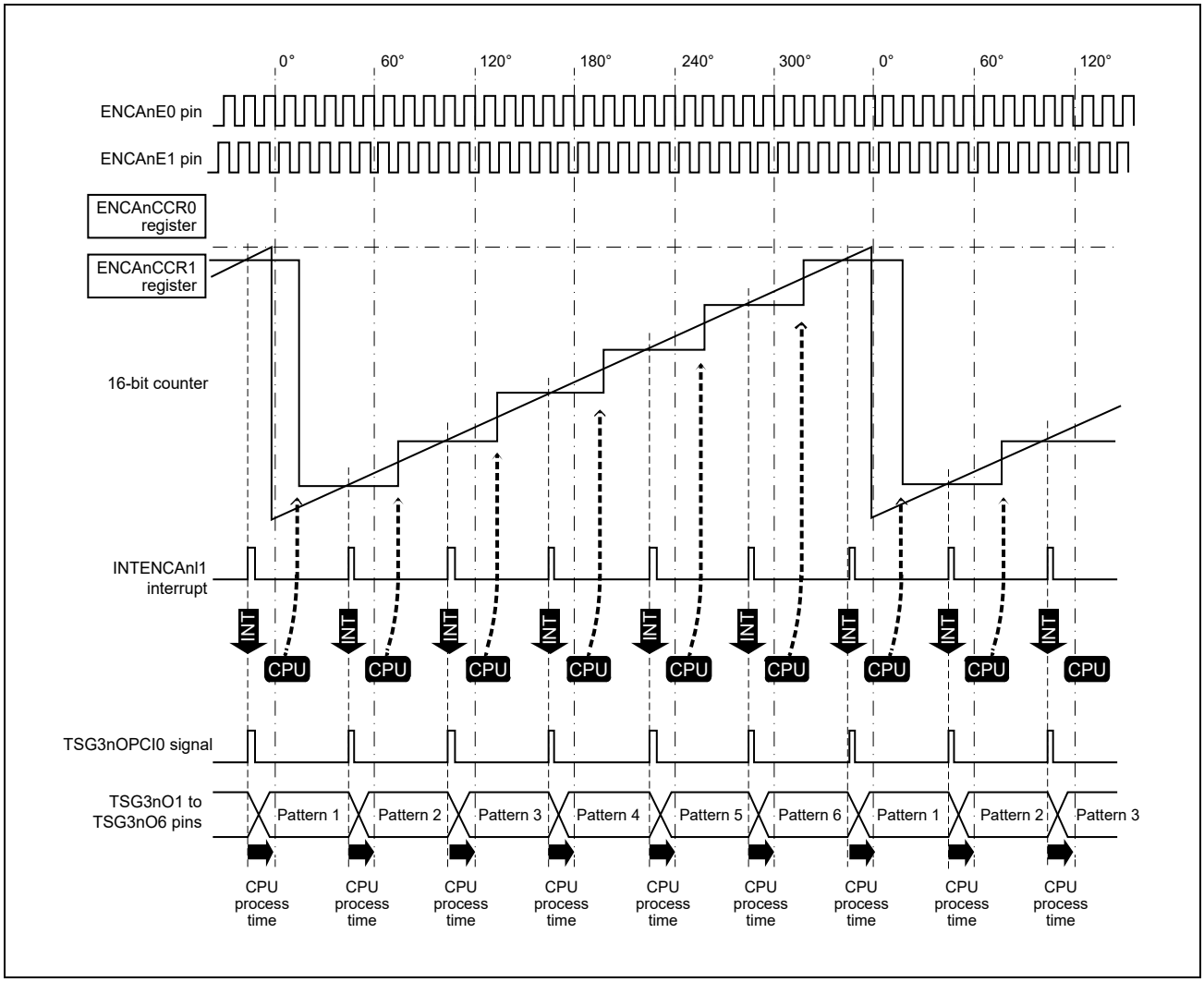


Figure 24.34 Timing Diagram of Two-Phase Encoder Control Function (Control Method 1) at Up Count (Normal Rotation)

(1) A match of the value of the encoder counter and ENCAnCCR1 allows INTENCAnI1 to be generated, and the pattern is output from the TSG3nO1 to TSG3nO6 pins.

(2) CPU calculates the next timing to switch output patterns by an interrupt processing and set the value to ENCAnCCR1.

(3) With a match of the value of the encoder counter and ENCAnCCR0 the encoder counter is cleared.

CAUTION

It is necessary to set ENCAnCCR1 at each pattern switch (each INTENCAnI1 interrupt). It is necessary to match the initial output pattern of timer TSG3n to the set value of ENCAnCCR1 before start.

Switching between normal and reverse rotations of output patterns should be set with the TSG3nPSC bit of the TSG3nOPT0 register.

The following figure shows the timing diagram.

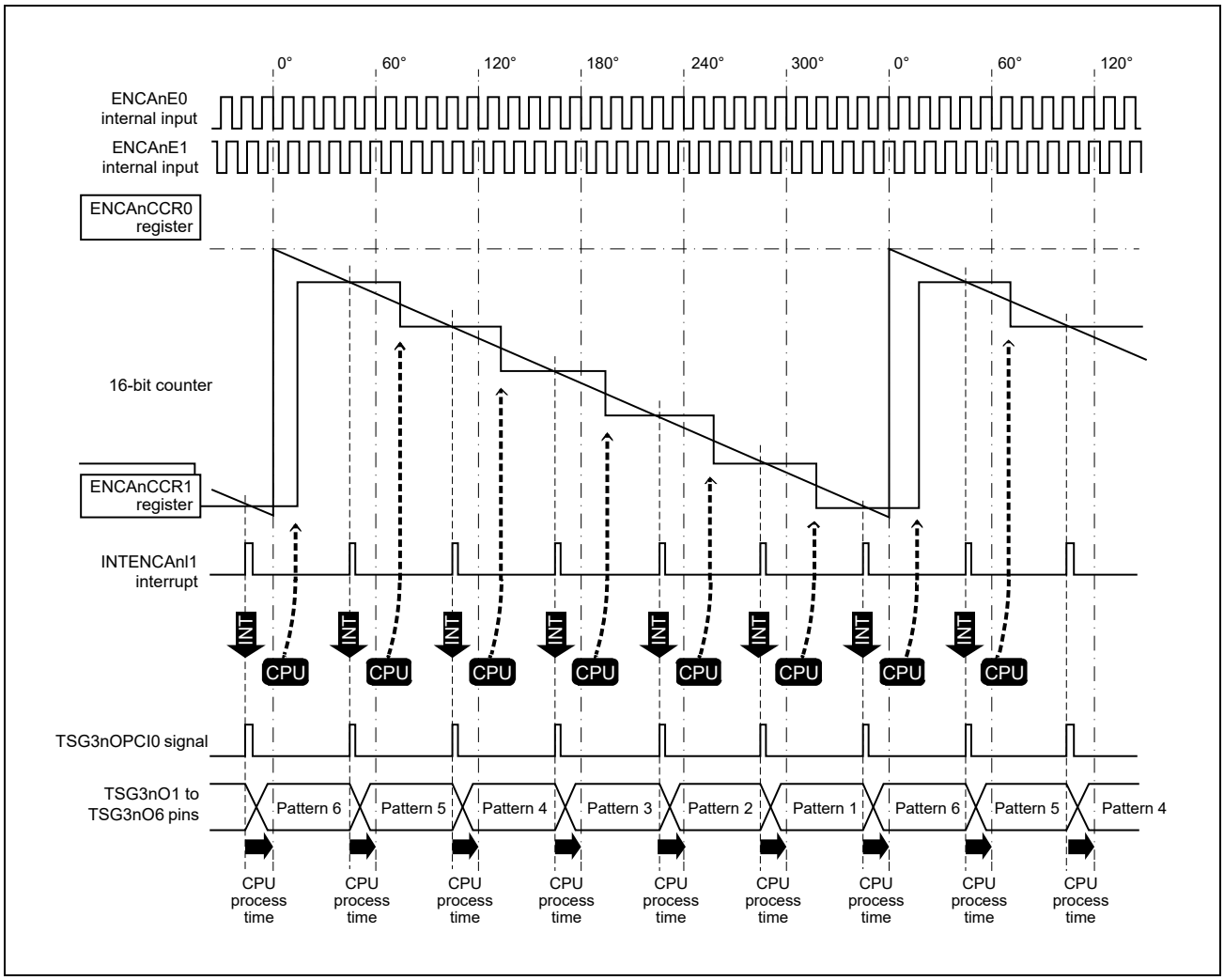


Figure 24.35 Timing Diagram of Two-Phase Encoder Control Function (Control Method 1) at Down Count (Reverse Rotation)

(5) Flow chart

The following figure shows the setting flow of this function.

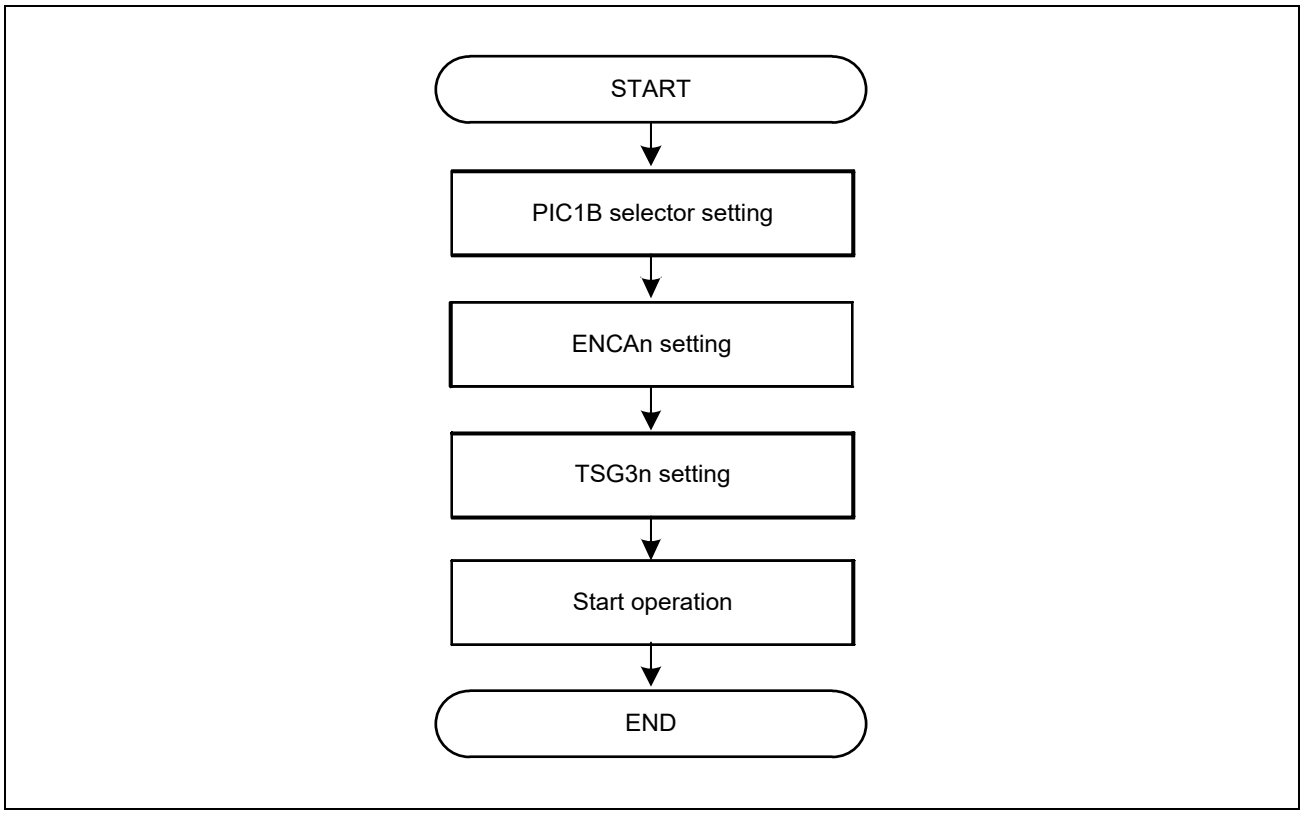


Figure 24.36 Setting Flow

The following figure shows the flow chart after interrupt processing.

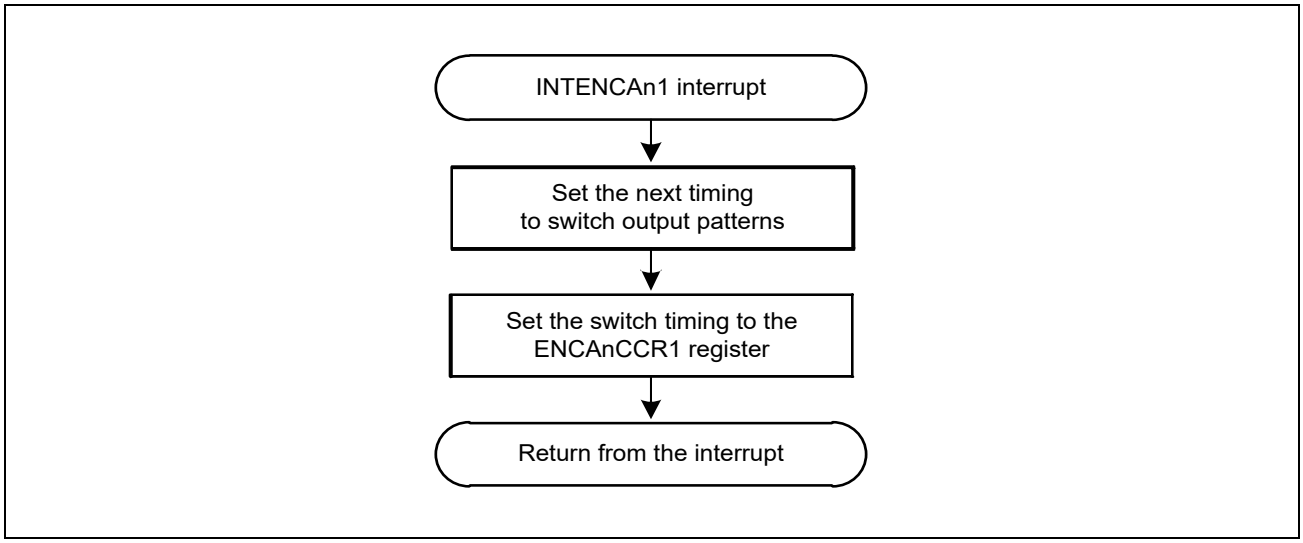


Figure 24.37 Flow after Interrupt Processing

The values of ENCAn registers used in this function are as follows.

ENCAnCTL[15:0] = $1000\_0000\_000x\_01xx_B$

ENCAnIOC1[7:0] = 0000_00xx_B

ENCAnCCR0 = (set any value)

ENCAnCCR1 = (set any value)

ENCAnCNT = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 23, Encoder Timer (ENCA)**.

The values of TSG3n registers used in this function are as follows.

TSG3nCTL0[7:0] = $000x\_0011_B$

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_0001\_xxx0\_0000_B$

TSG3nIOC0[7:0] = 0111_1110_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = 0011_1xx0_B

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 20, Motor Control Timer (TSG3)**.

24.2.3.9 Two-Phase Encoder Control Function (Control Method 2)

(1) Overview

This function allows switching the advance and retard control of the motor control function (TSG3n) output patterns in 120-DC mode using the two-phase encoder control function (ENCAn).

(2) Configuration

The similar configuration as the **Section 24.2.3.8, Two-Phase Encoder Control Function (Control Method 1)** is applied this function. See **(2)Configuration** of **Section 24.2.3.8, Two-Phase Encoder Control Function (Control Method 1)**.

(3) Registers

The similar configuration as the **Section 24.2.3.8, Two-Phase Encoder Control Function (Control Method 1)** is applied this function. See **(3)Registers** of **Section 24.2.3.8, Two-Phase Encoder Control Function (Control Method 1)**.

(4) Function

Detail of the function is described using the two-phase encoder control function (method 2) at advance control (normal rotation) as an example.

The following figure shows the timing diagram.

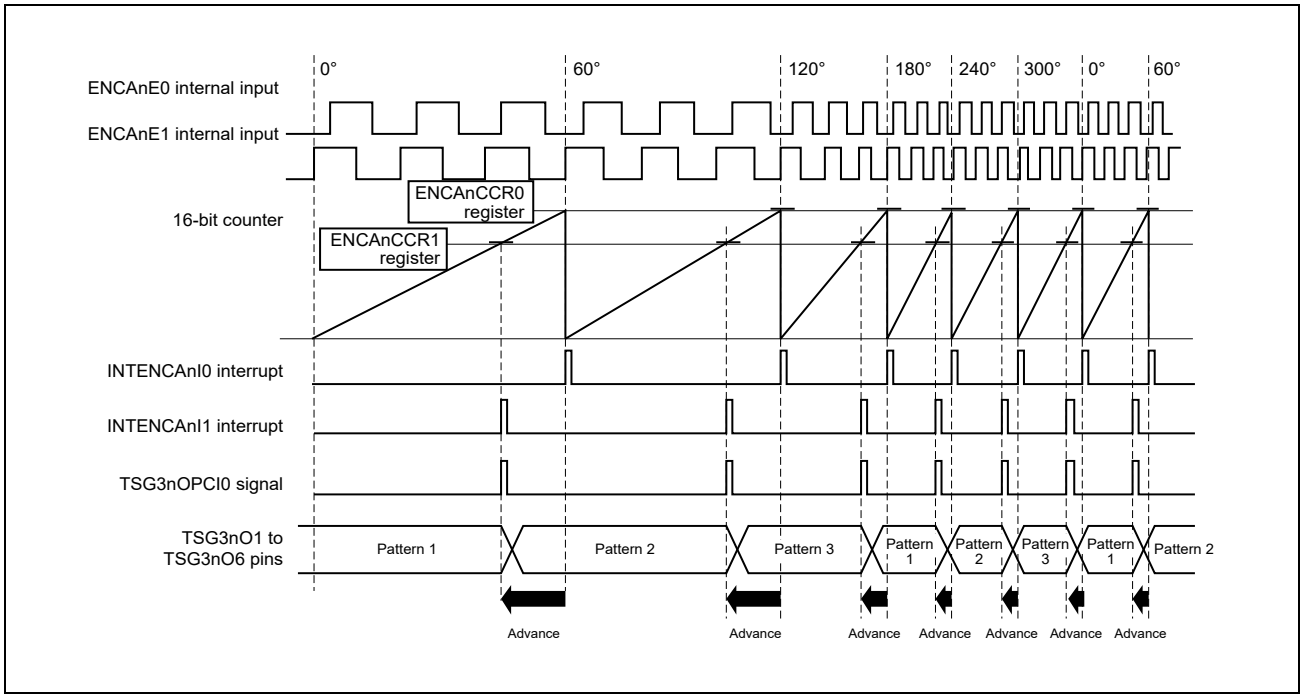


Figure 24.38 Timing Diagram of Two-Phase Encoder Control Function (Method 2) at Advance Control (Normal Rotation)

(1) A match of the value of the encoder counter and ENCAnCCR1 (corresponds to the output pattern switch position for TSG3n) allows INTENCAnI1 to be generated, and the pattern is output from the TSG3nO1 to 6 pins.

(2) With a match of the value of the encoder counter and ENCAnCCR0 (corresponds to the phase advance and retard of the switch position), INTENCAnI0 is generated and the encoder counter is cleared.

CAUTION

It is necessary to set ENCAnCCR1 at each pattern switch (each INTENCAnI1 interrupt). It is necessary to match the initial output pattern of timer TSG3n to the set value of ENCAnCCR0 before start.

Switching between normal and reverse rotations of output patterns should be set with the TSG3nPSC bit of the TSG3nOPT0 register.

The following figure shows the timing diagram at retard control (normal rotation).

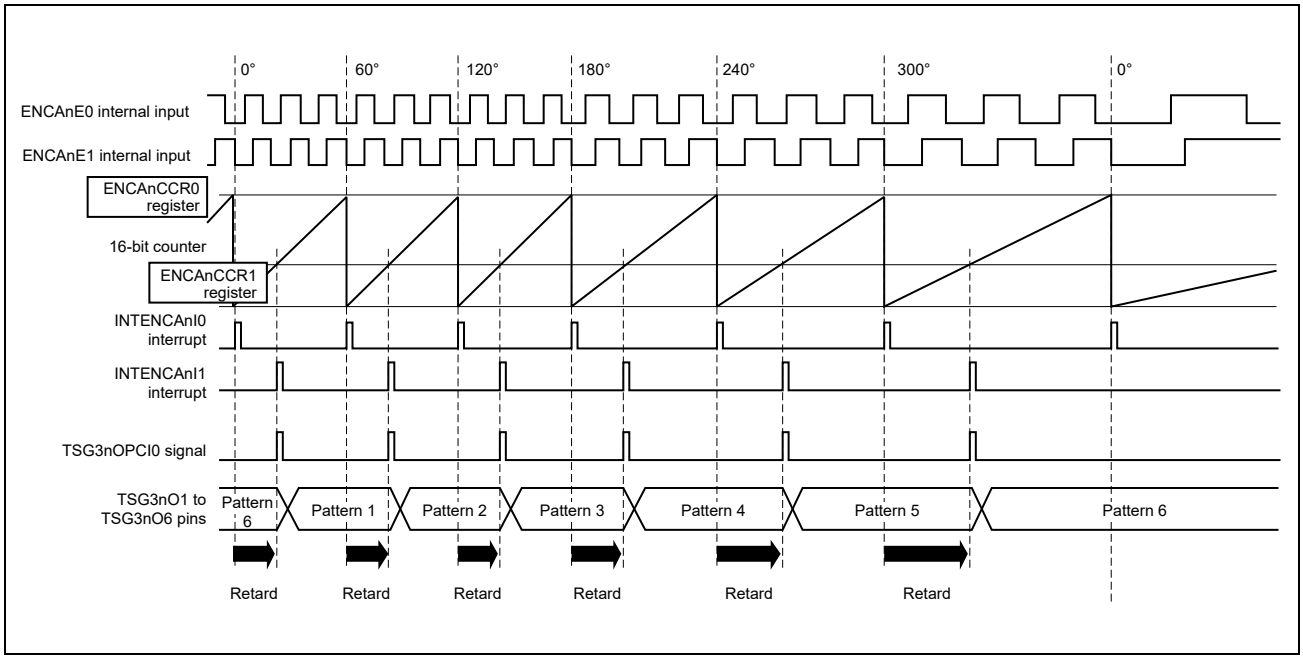


Figure 24.39 Timing Diagram of Two-Phase Encoder Control Function (Method 2) at Retard Control (Normal Rotation)

By setting greater value to ENCAnCCR1 than that of ENCAnCCR0, TSG3n output pattern phase can be retarded.

NOTE

This function can be used for advance control and retard control both at up count and down count.

(5) Flow chart

The flow charts for this function are shown as follows.

Flow chart of main routine

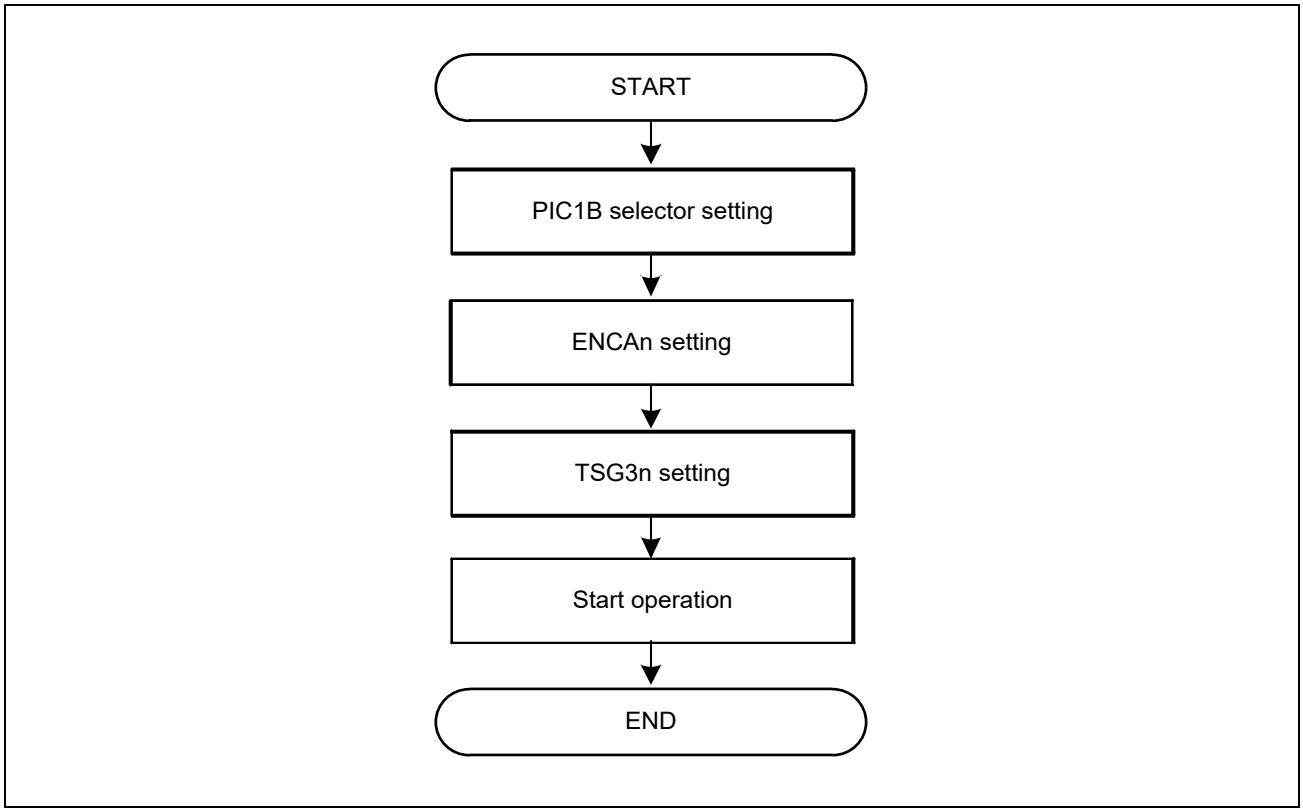


Figure 24.40 Setting Flow

Flow chart of ENCAnCCR1 rewrite processing at advance control

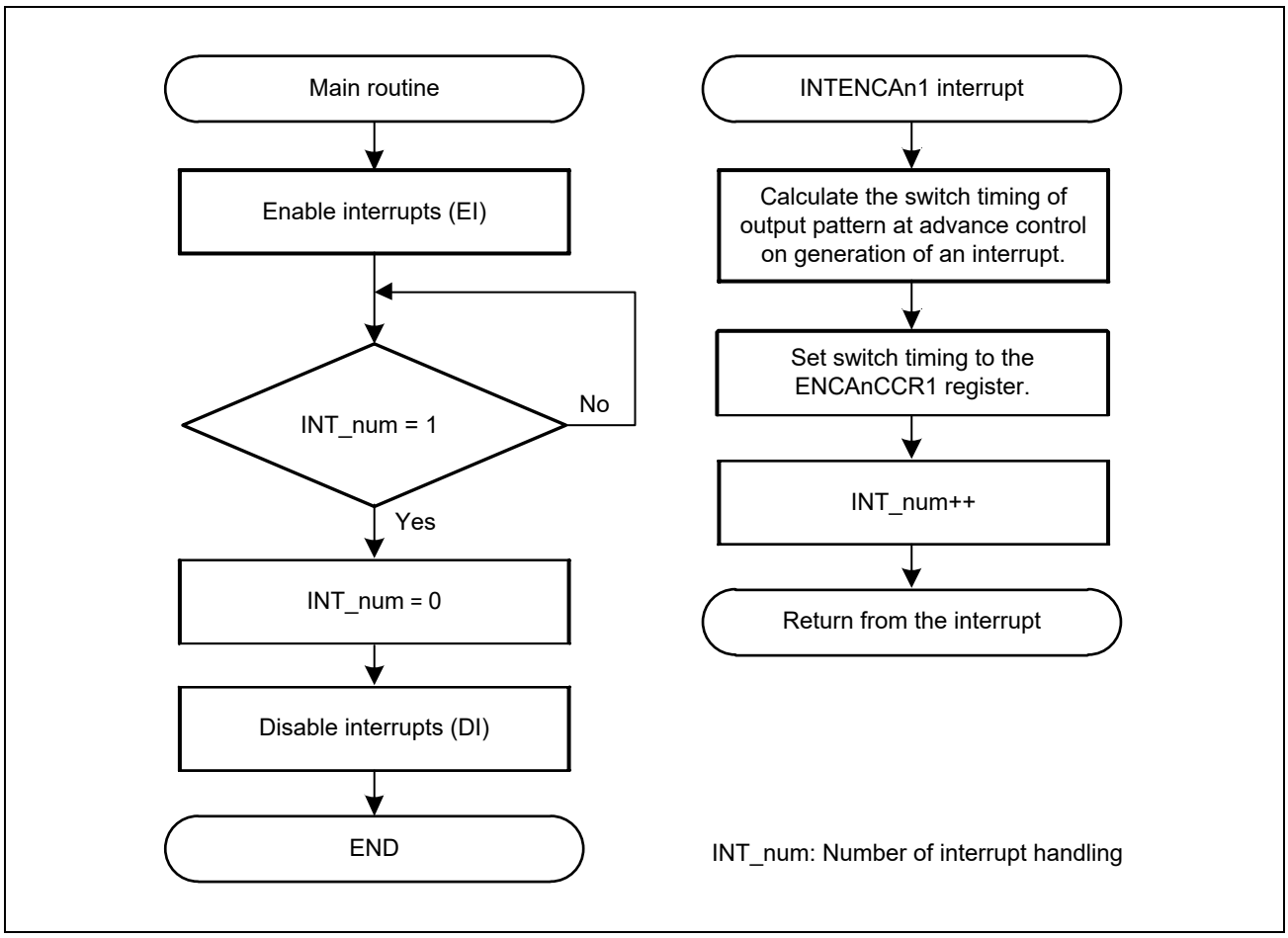


Figure 24.41 ENCAnCCR1 Rewrite Processing at Advance Control (Setting Flow)

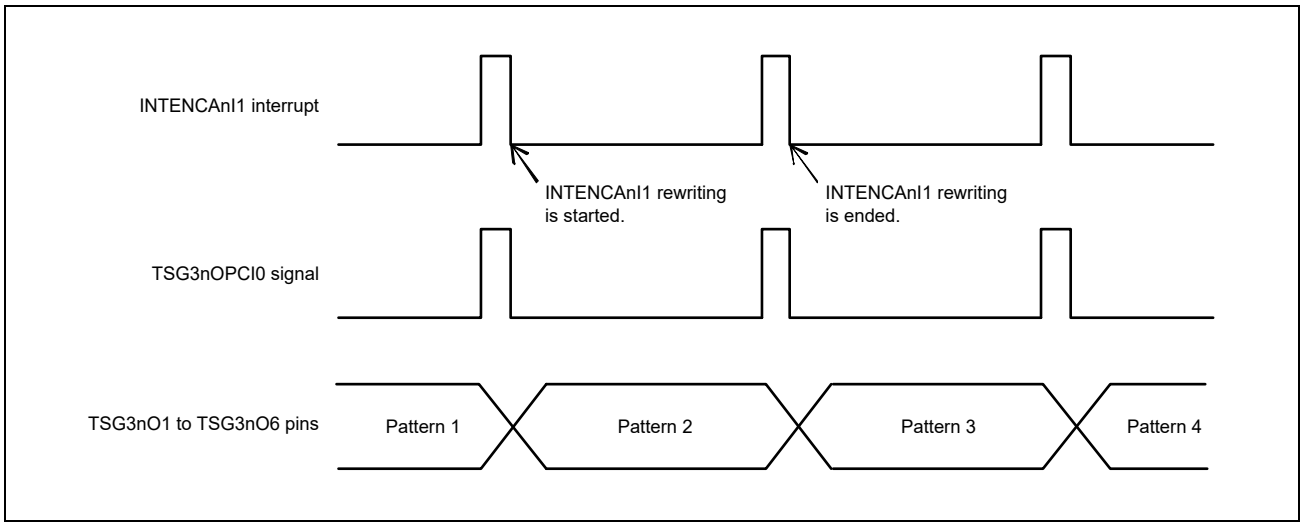


Figure 24.42 ENCAnCCR1 Rewrite Processing at Advance Control (Timing)

Flow chart of ENCAnCCR1 rewrite processing at retard control

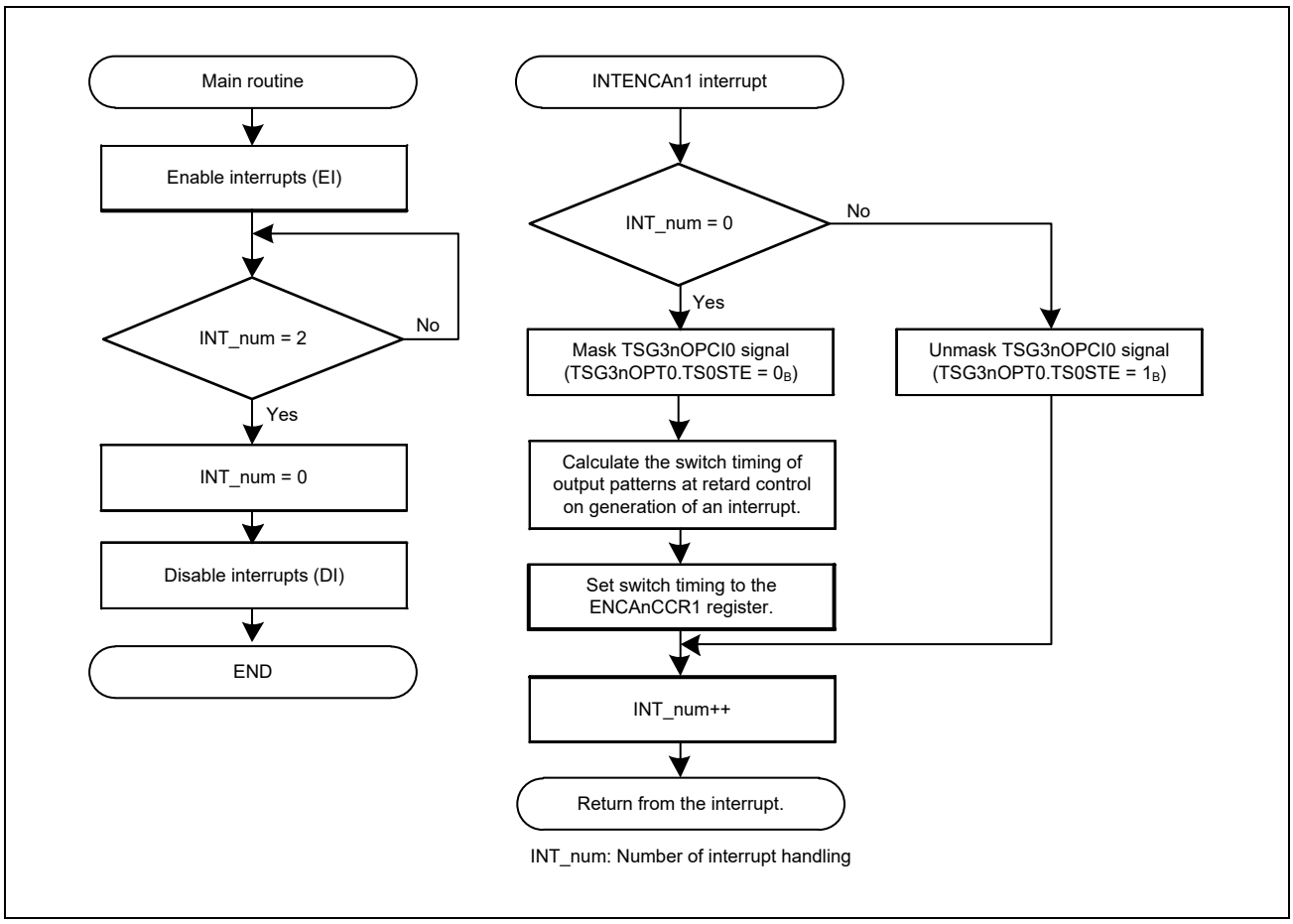


Figure 24.43 ENCAnCCR1 Rewrite Processing at Retard Control (Setting Flow)

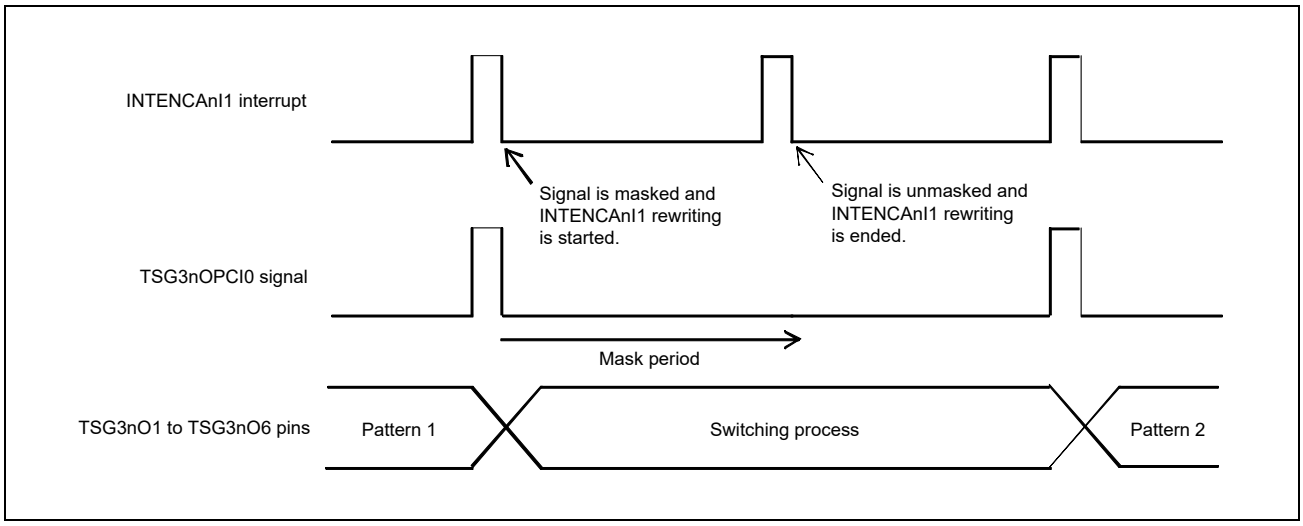


Figure 24.44 ENCAnCCR1 Rewrite Processing at Retard Control (Timing)

The register settings for ENCAn to use this function are described as follows.

ENCAnCTL[15:0] = $1000\_0000\_000x\_01xx_B$

ENCAnIOC1[7:0] = 0000_00xx_B

ENCAnCCR0 = (set any value)

ENCAnCCR1 = (set any value)

ENCAnCNT = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 23, Encoder Timer (ENCA)**.

The register settings for TSG3n to use this function are described as follows.

TSG3nCTL0[7:0] = $000x\_0011_B$

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_0001\_xxx0\_0000_B$

TSG3nIOC0[7:0] = 0111_1110_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = 0011_1xx0_B

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 20, Motor Control Timer (TSG3)**.

24.2.3.10 Two-Phase Encoder Control Function (Control Method 3)

(1) Overview

Using the two-phase encoder control function (ENCAn), variation of the desired angle and phase (up to ±60°) by pattern output control is available regarding the angle of motor rotation indicated by the ENCA0 and ENCA1 timers.

(2) Configuration

Variation of desired angle and phase control by output pattern in 120-DC mode is realized by using ENCAn and TSG3n, and PIC1B in combination.

The following figure describes the block diagram of two-phase encoder control function (control method 3).

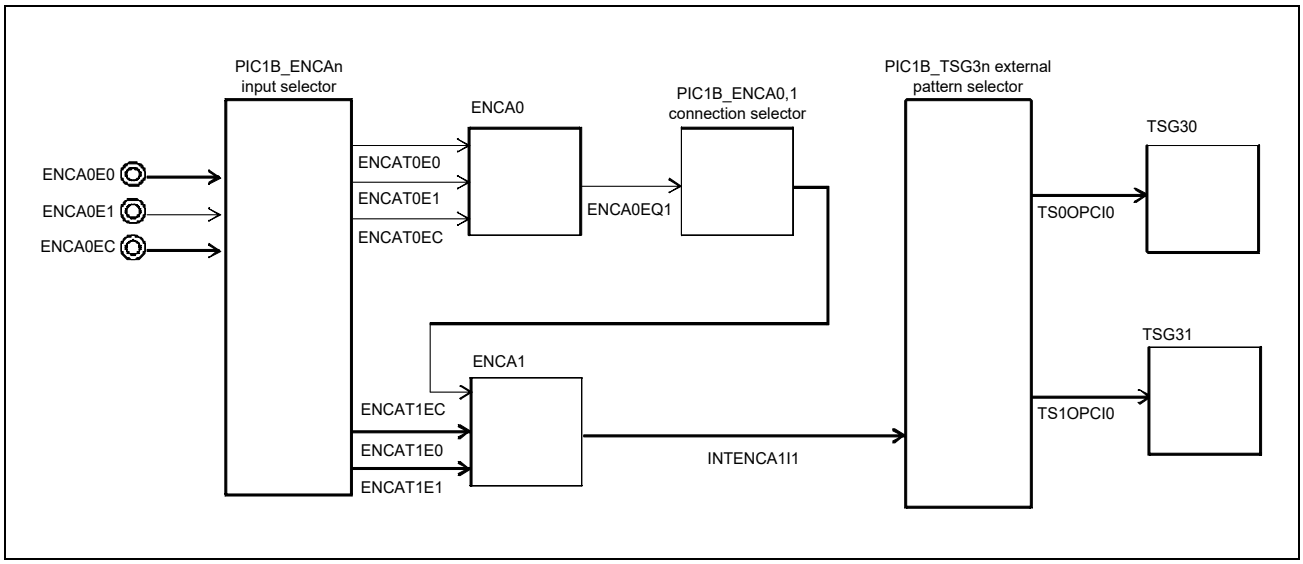


Figure 24.45 Block Diagram of Two-Phase Encoder Control Function (Control Method 3)

The configuration of this function is described as follows.

- [PIC1B_ENCAn input selector]
  ENCA0E0, ENCA0E1, and ENCA0EC pin inputs are selected and output to ENCATnE0, ENCATnE1, and ENCATnEC.
- [ENCA0]
  ENCA0EQ1 is output by two-phase encoder processing.
- [PIC1B_ENCA0, 1 connection selector]
  ENCA0EQ1 is selected and output to ENCAT1EC.
- [ENCA1]
  INTENCA1I1 is output by two-phase encoder processing. If an active level signal is input to ENCAT1EC, the timer count value is cleared.
- [PIC1B_TSG3n external pattern selector]
  INTENCA1I1 is selected and output to TSG3nOPCI0.
- [TSG3n]
  Output patterns in 120-DC mode are switched by TSG3nOPCI0.

(3) Registers

Block diagram of PIC1B is shown in the following figure.

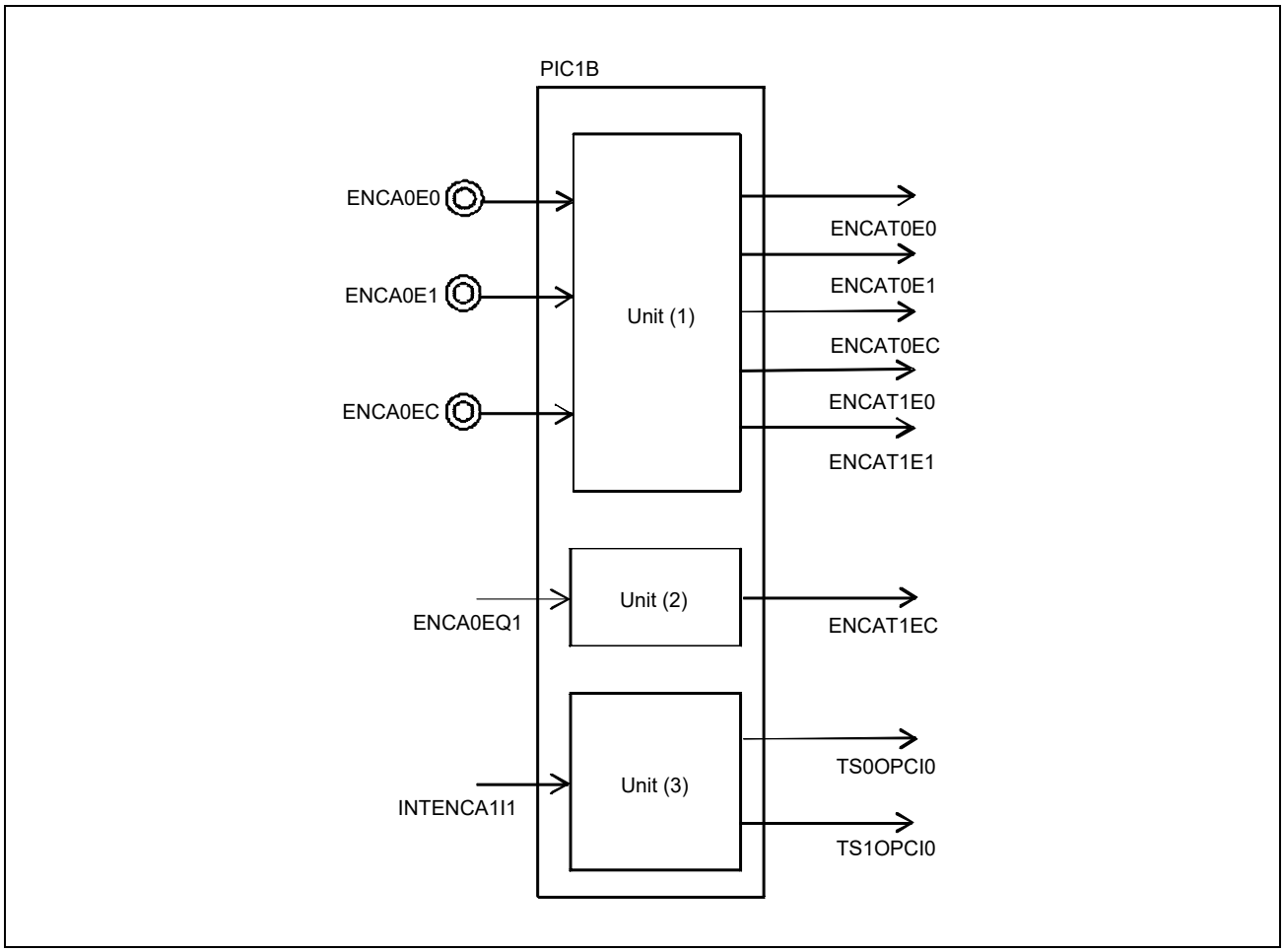


Figure 24.46 Block Diagram of PIC1B

The register settings for PIC1B to use this function are described as follows.

Unit (1): PIC1B_ENCAn input selector

PIC1BREG30[22] = 0_B

PIC1BREG30[17:16] = 00_B

PIC1BREG30[9:6] = 0101_B

PIC1BREG30[1:0] = 00_B

Unit (2): PIC1B_ENCA0, 1 connection selector

PIC1BREG30[11:10] = 11_B

Unit (3): PIC1B_TSG3n external pattern selector

PIC1BREG5n[10] = 0_B

PIC1BREG5n[6:5] = 10_B

(n = 0 when TSG30 is selected and n = 1 when TSG31 is selected.)

(4) Function

Detail of the function is described using the two-phase encoder control function (control method 3) at advance control (normal rotation) as an example.

The following figure shows the timing diagram.

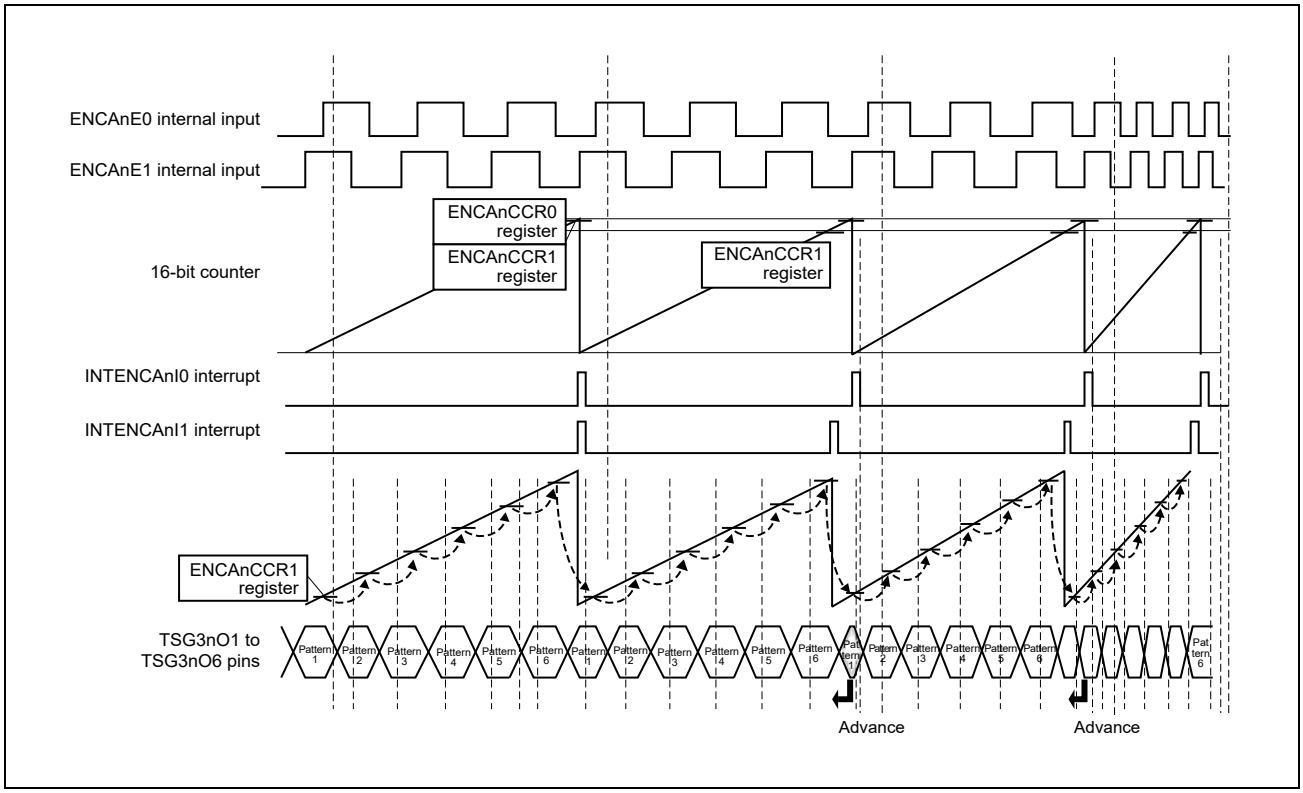


Figure 24.47 Timing Diagram of Two-Phase Encoder Control Function (Control Method 3) at Advance Control (Normal Rotation)

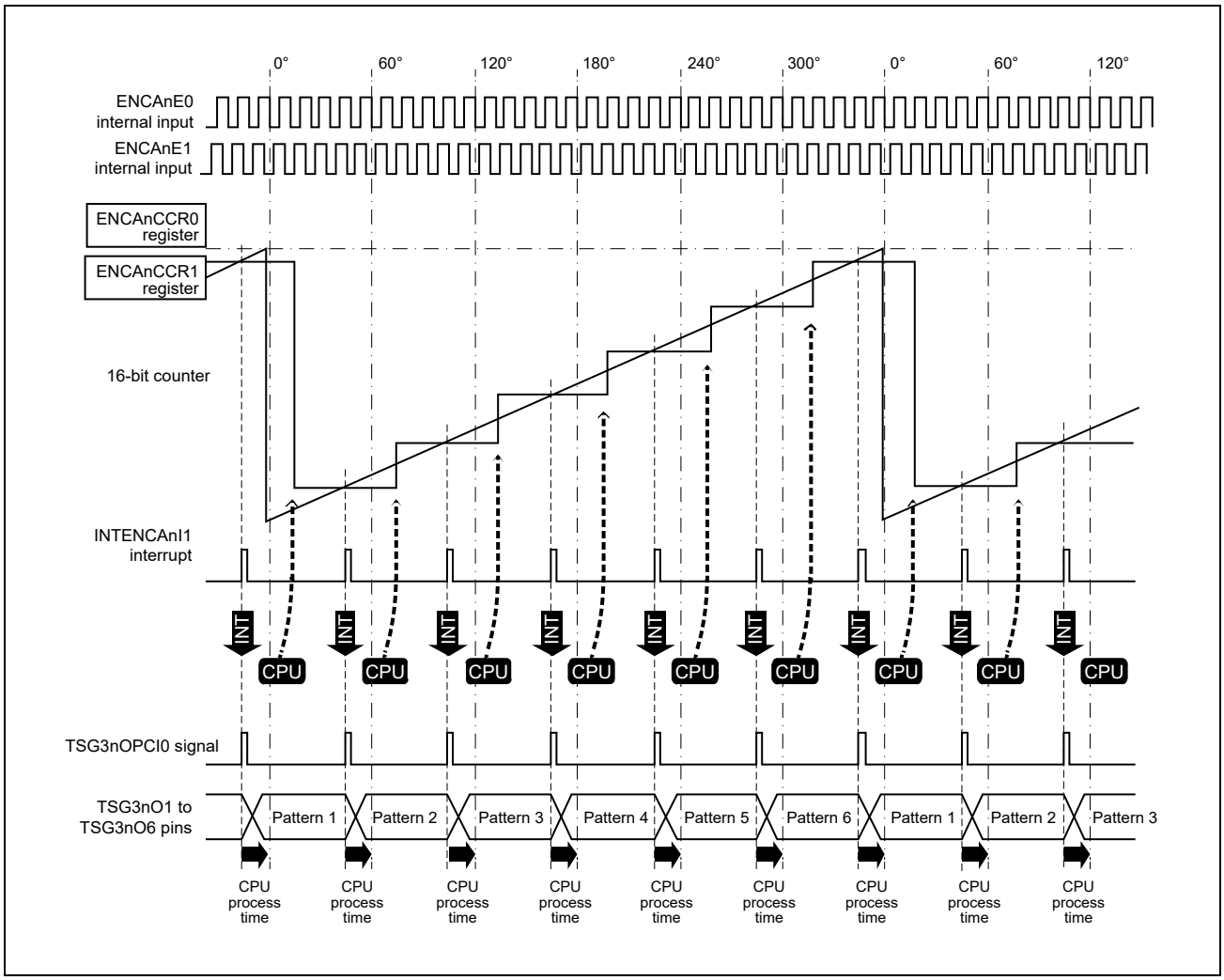


Figure 24.48 Timing Diagram of TSG3n Output Pattern Control (ENCA1) (Normal Rotation)

(1) A match of the value of the ENCA0 encoder counter and ENCA0CCR1 allows INTENCA0I1 to be generated and ENCA1 encoder counter is cleared.

(2) A match of the value of the ENCA1 encoder counter and ENCA1CCR1 allows INTENCA1I1 to be generated, and the pattern is output from the TSG3nO1 to 6 pins.

(3) With a match of the value of the ENCA0 encoder counter and ENCA0CCR0, INTENCA0I0 is generated and the encoder counter is cleared.

(5) Flow chart

The flow charts for this function are shown as follows.

Flow chart of main routine

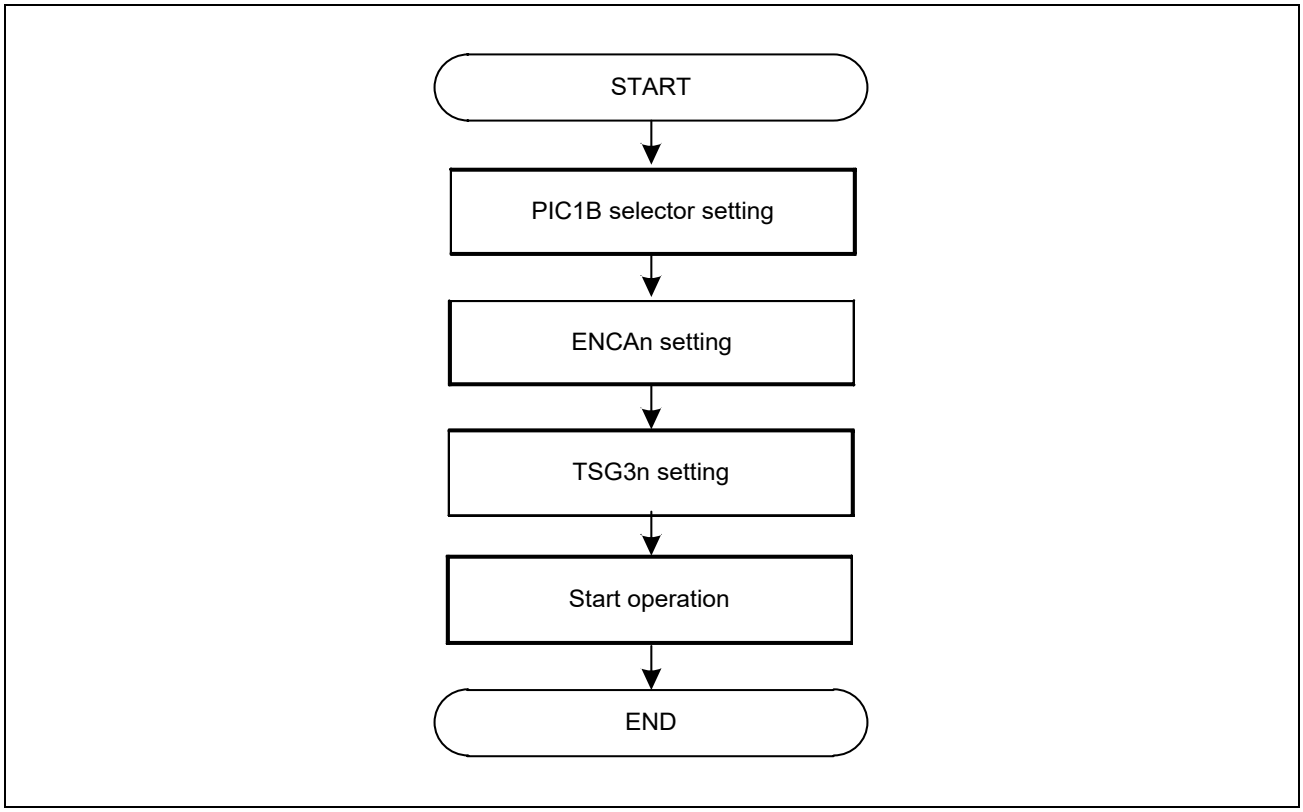


Figure 24.49 Setting Flow

Flow chart of ENCAnCCR1 rewrite processing at advance control

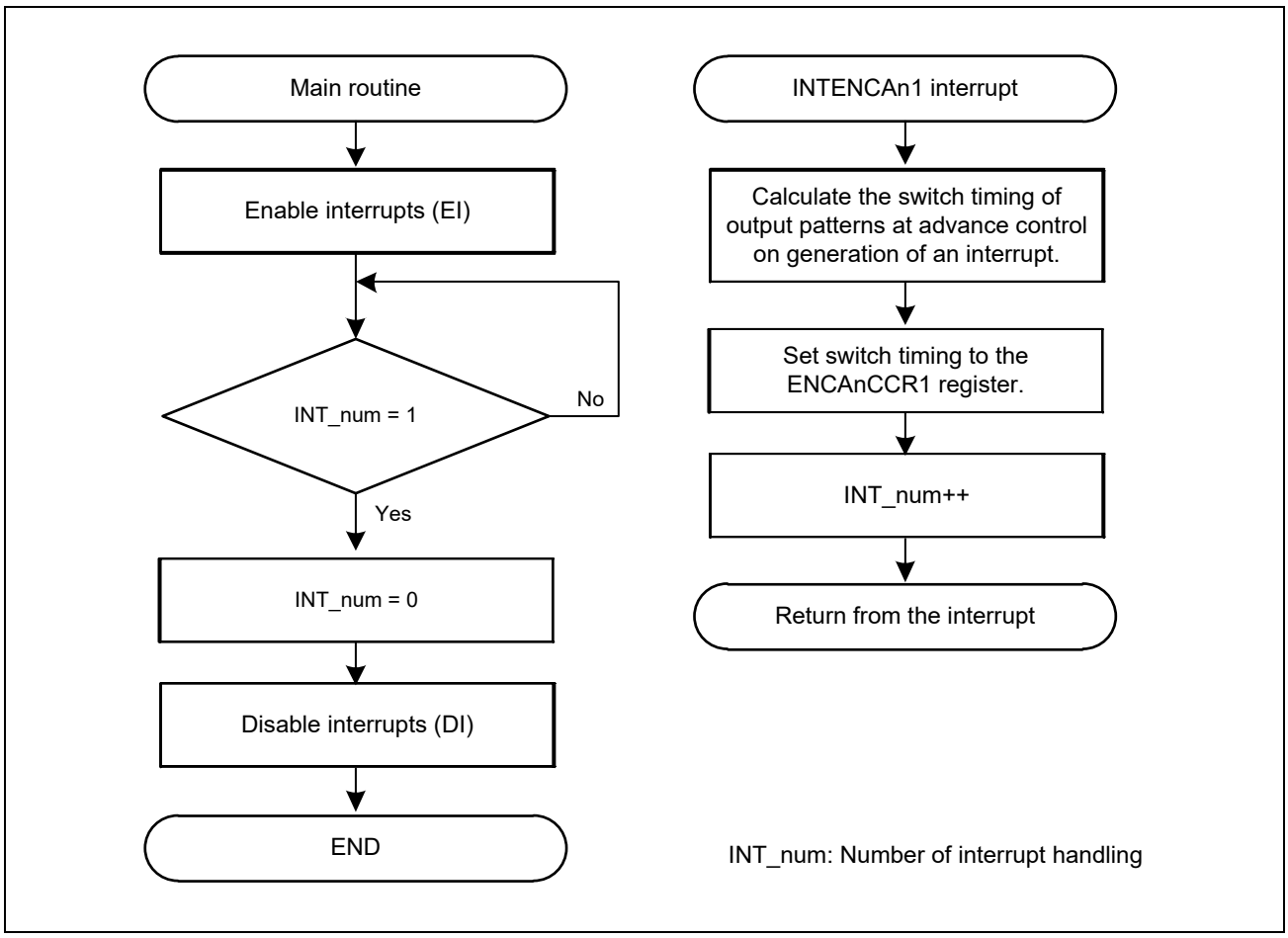


Figure 24.50 ENCAnCCR1 Rewrite Processing at Advance Control

Flow chart of ENCAnCCR1 rewrite processing at retard control

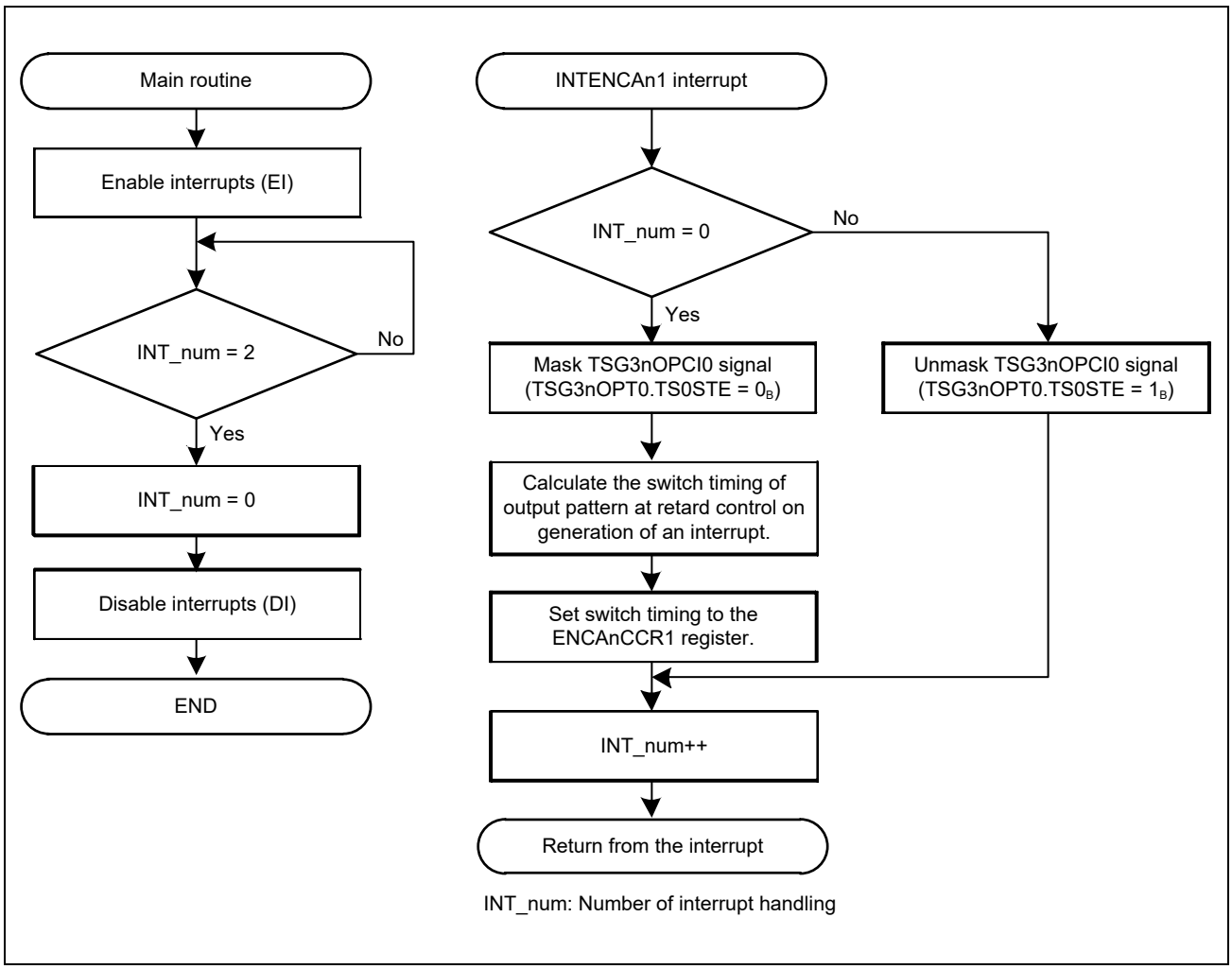


Figure 24.51 ENCAnCCR1 Rewrite Processing at Retard Control (Setting Flow)

The register settings for ENCAn to use this function are described as follows.

ENCAnCTL[15:0] = $1000\_0000\_000x\_01xx_B$

ENCAnIOC1[7:0] = 0000_00xx_B

ENCAnCCR0 = (set any value)

ENCAnCCR1 = (set any value)

ENCAnCNT = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 23, Encoder Timer (ENCA)**.

The register settings for TSG3n to use this function are described as follows.

TSG3nCTL0[7:0] = $000x\_0011_B$

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_0001\_xxx0\_0000_B$

TSG3nIOC0[7:0] = 0111_1110_B

TSG3nIOC2[15:0] = 0xxx_xxx0_0000_0000$_B$

TSG3nOPT0[7:0] = 0011_1xx0$_B$

TSG3nOPT1[7:0] = 0000_0xxx$_B$

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 20, Motor Control Timer (TSG3)**.

24.2.3.11 Three-Phase Pulse Input Control Function

(1) Overview

This function allows variable phase control of TSG3n pattern output in 120-DC mode using TSG3n and TAUDn.

The following diagram shows the method of three-phase pulse input control.

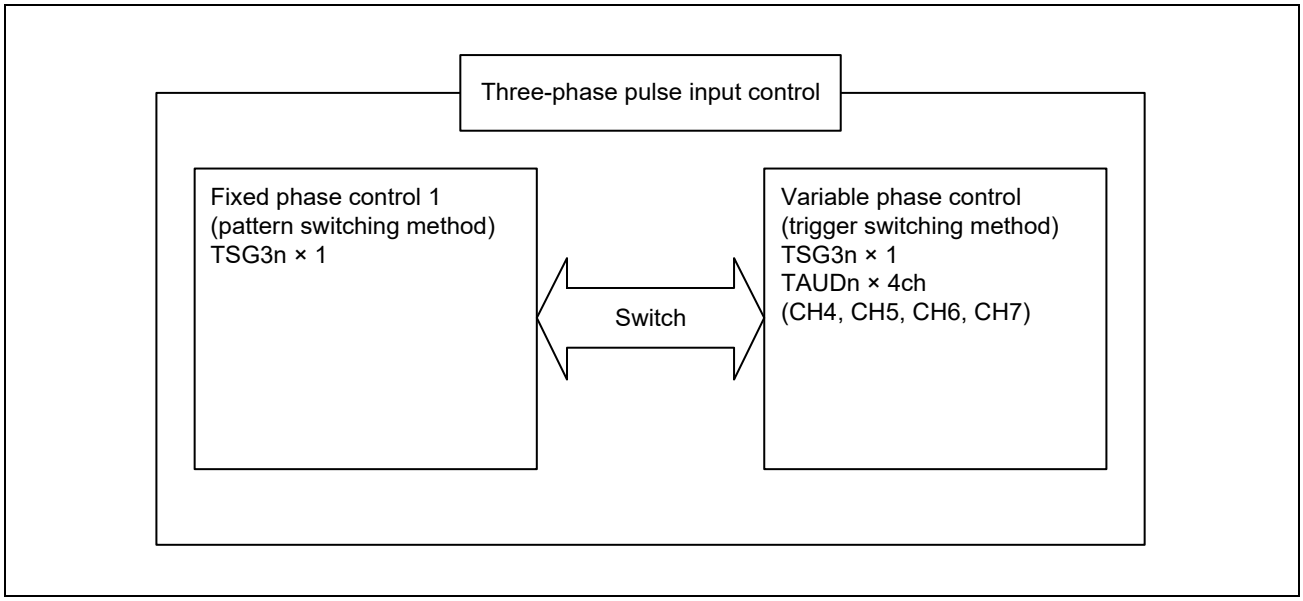


Figure 24.52 Control Method of Three-Phase Pulse Input Control Function

Table 24.76 Control Method of Three-Phase Pulse Input Function

Control Method	Function
Fixed phase control 1 (pattern switching method)	Outputs a fixed pattern at constant rotation angle.
Variable phase control (trigger switching method)	Varies the phase by arbitrary angle (or time) up to ±60 degrees with reference to the rotation angle and outputs the pattern.

(2) Configuration

Three-phase pulse input control function is realized by using three-phase pulse input and TAUDn offset trigger mode, and PIC1B in combination.

The following figure shows the block diagram of three-phase pulse input control function.

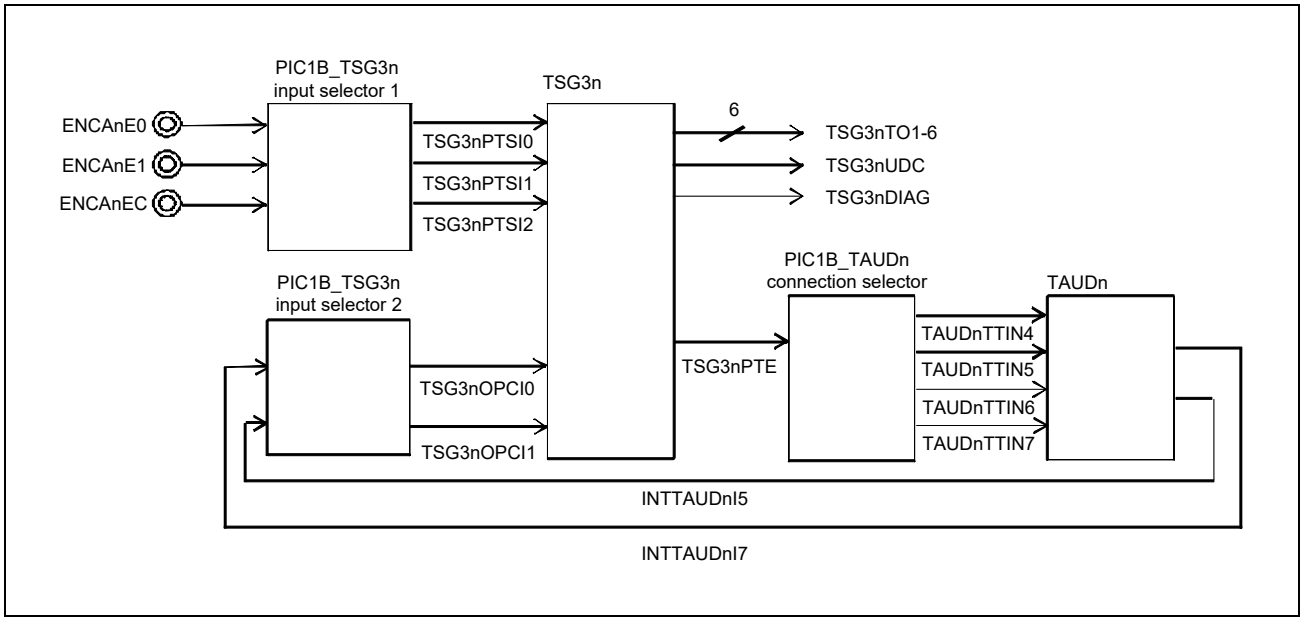


Figure 24.53 Block Diagram of Three-Phase Pulse Input Control

The configuration of this function is described as follows.

- [PIC1B_TSG3n input selector 1]
  ENCAnE0, ENCAnE1, and ENCAnEC pin inputs are selected and output to TSG3nPTSI0 to TSG3nPTSI2.
- [TSG3n]
  Patterns set in TSG3nTO1 to TSG3nTO6 are output in response to the input of TSG3nPTSI0 to TSG3nPTSI2 signals. TSG3nPTE is toggled each time the output patterns are switched.
- [PIC1B_TAUDn connection selector]
  TSG3nPTE input is selected and output to TAUDnTTIN4 to TAUDnTTIN7.
- [TAUDn]
  Interrupt signals INTTAUDnI5 and INTTAUDnI7 for output pattern phase generation are output with the offset trigger mode.
- [PIC1B_TSG3n input selector 2]
  INTTAUDnI5 and INTTAUDnI7 inputs are selected and output as TSG3nOPCI0 and TSG3nOPCI1.

(3) Registers

Block diagram of PIC1B is shown in the following figure.

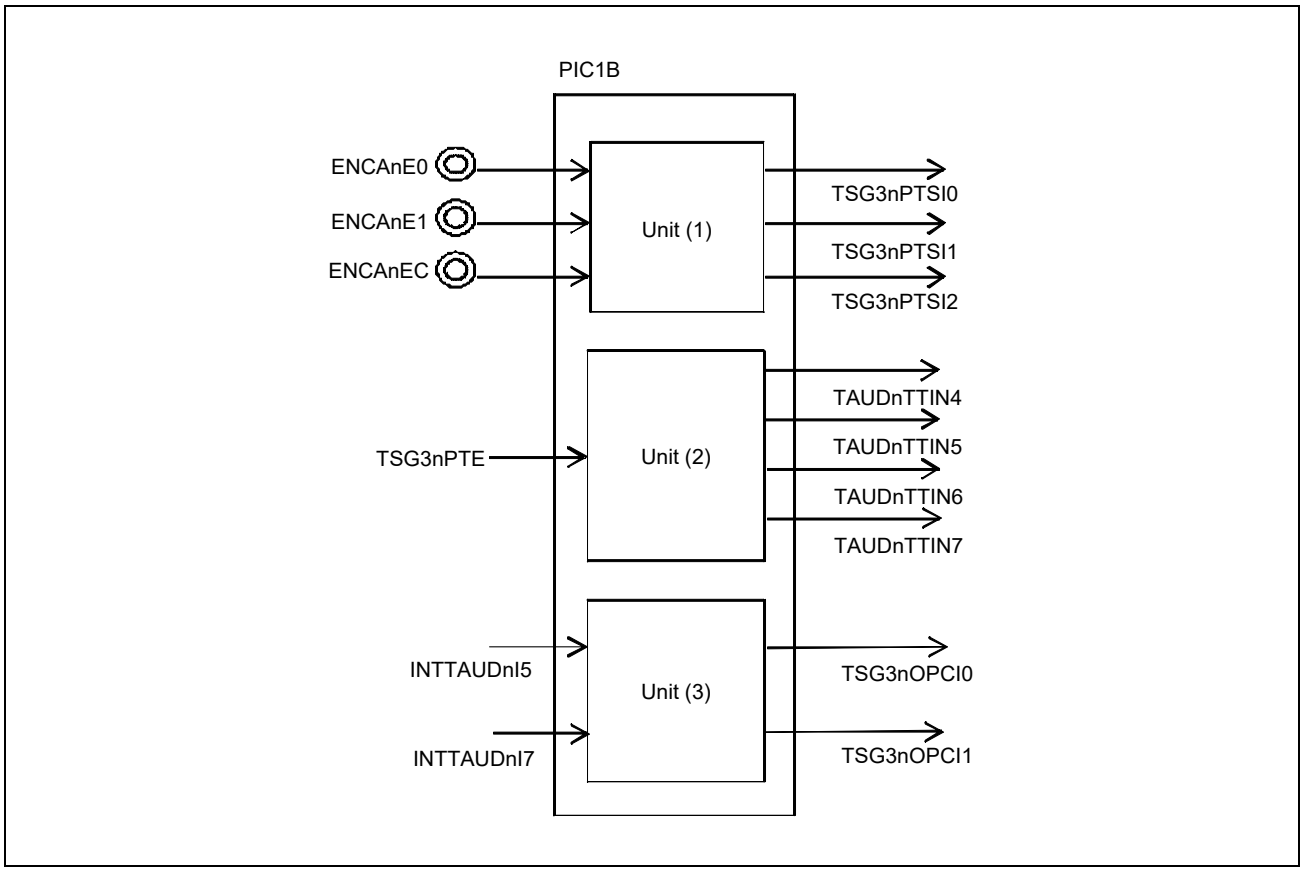


Figure 24.54 Block Diagram of PIC1B

The values of PIC1B registers used in this function are as follows.

Unit (1): PIC1B_TSG3n input selector 1

The values to output ENCA0E0, ENCA0E1, and ENCA0EC as TSG30PTSI0 to TSG30PTSI2
PIC1BTSGHALLSEL[0] = 1_B
PIC1BREG50[0] = 0_B

The values to output ENCA1E0, ENCA1E1, and ENCA1EC as TSG31PTSI0 to TSG31PTSI2
PIC1BTSGHALLSEL[1] = 1_B
PIC1BREG51[0] = 1_B

Unit (2): PIC1B_TAUDn connection selector

The values to output TSG3nPTE to TAUDnTTIN4 to TAUDnTTIN7.
PIC1BREG2n0[11:8] = 1010_B
PIC1BREG2n0[3:0] = 1111_B
PIC1BTAUDnSEL[15:8] = 00_H

Unit (3): PIC1B_ENCAn input selector 2

The values to output INTTAUDnI5 and INTTAUDnI7 to TSG3nOPCI0 and TSG3nOPCI1.
PIC1BREG5n[7:5] = 011_B

(4) Function

Detail of the three-phase pulse input control function is described here.

The following figure shows the timing diagram.

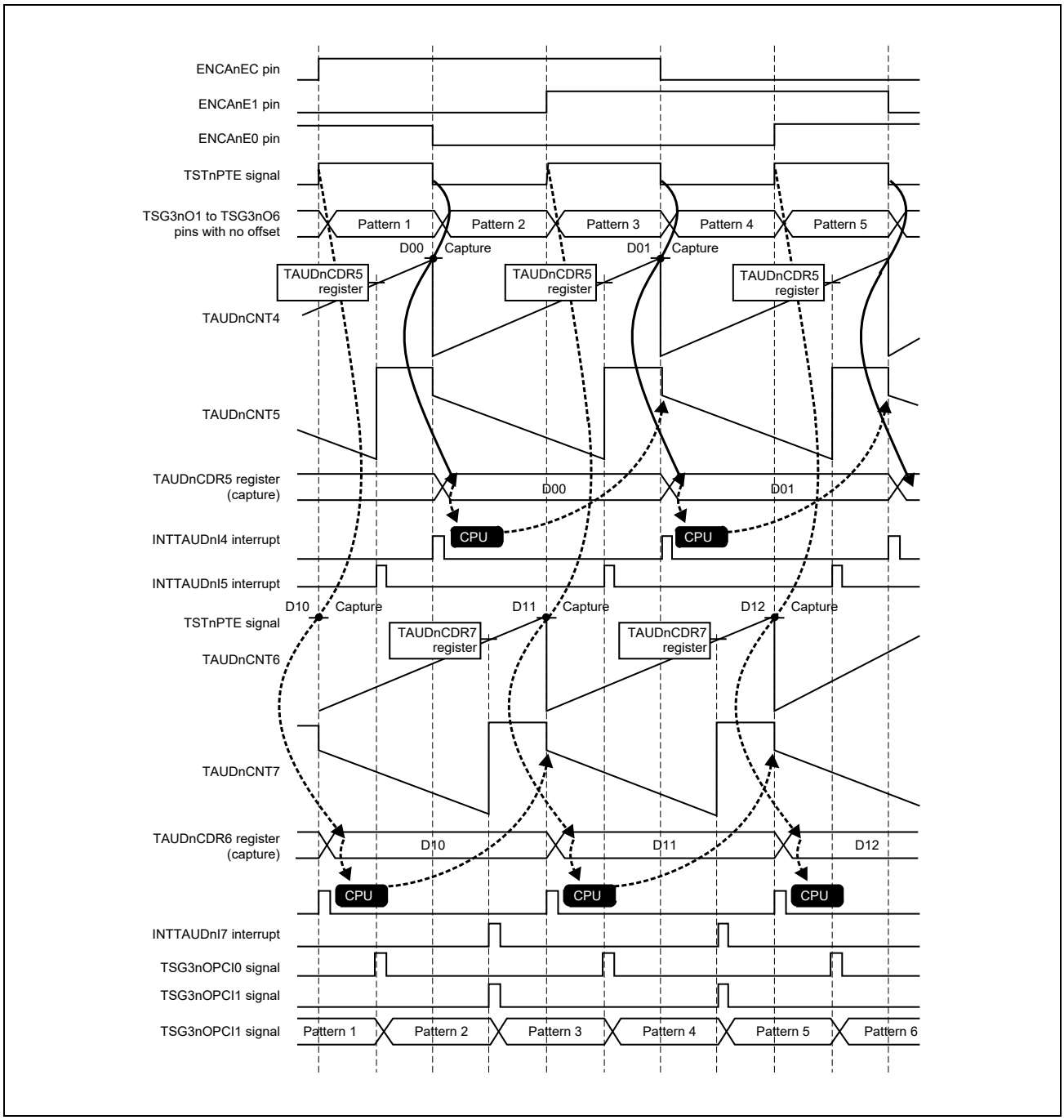


Figure 24.55 Three-Phase Pulse Input Control Function

(1) The output patterns are switched by the TSG pattern switching method on detection of ENCAn pin signal. The TSTnPTE signal toggles each time the patterns are switched.

(2) TAUDnCNT6 and TAUDnCNT4 are captured at the rising edge and falling edge of the TSTnPTE signal. The interval to switch patterns is calculated according to the captured value.

(3) CPU calculates the phase of the next output pattern and set the values to TAUDnCNT5 and TAUDnCNT7. The signals corresponding to the values are output as TSG3nOPCI0 and TSG3nOPCI1. At this time, the patterns delayed for the set phase are output by switching the output patterns by the trigger switching method.

The following table lists the relation between the value set to TAUDnCNTm and the captured value in TAUDnCDR (m-1) (m = 5, 7).

Table 24.77 Setting Value and Captured Value of TAUDnCNTm

TAUDnCNTm Register Setting	TSG3n Pattern Output Switch Timing
TAUDnCNTm = 0000_H	The patterns switched on detection of an edge of TSTnPTE signal with a delay of up to one cycle of the clock signal being counted by TAUDn is generated.
TAUDnCNTm = captured value	The patterns are switched on detection of an edge of TSTnPTE signal.
TAUDnCNTm < captured value	The patterns are switched on the timing after the phase delayed from detection of an edge of TSTnPTE signal.
TAUDnCNTm > captured value	Setting prohibited

An example of switch operation from fixed phase control 1 to variable phase control

The following figure shows an example of switch operation from fixed phase control 1 to variable phase control.

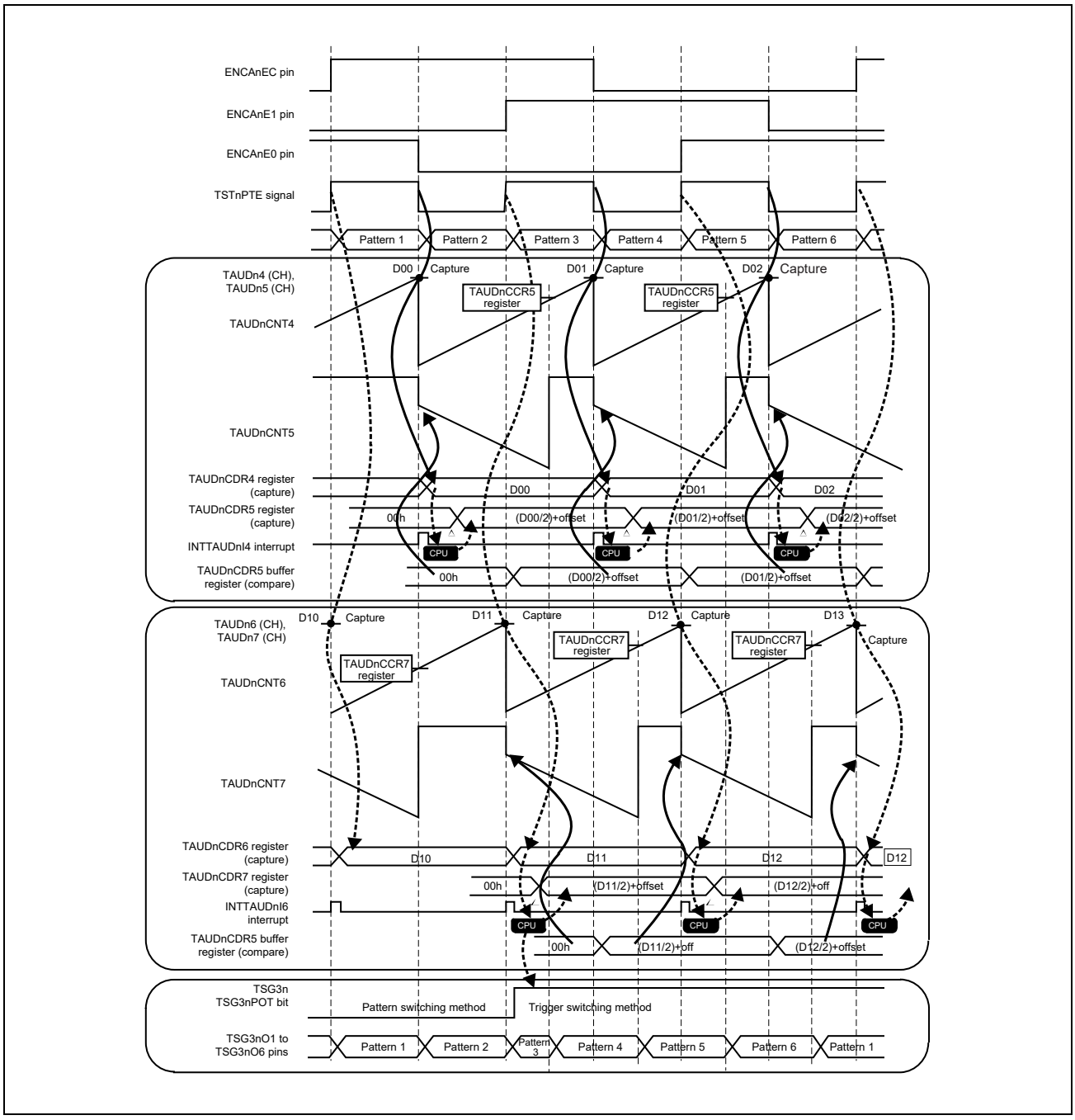


Figure 24.56 An Example of Switch Operation from Fixed Phase Control 1 to Variable Phase Control

The output pattern is changed to the trigger switching method by changing the TSG3nPOT bit from low level to high level, and the variable phase control is enabled.

An example of switch operation from variable phase control to fixed phase control 1

The following figure shows an example of switch operation from variable phase control to fixed phase control 1.

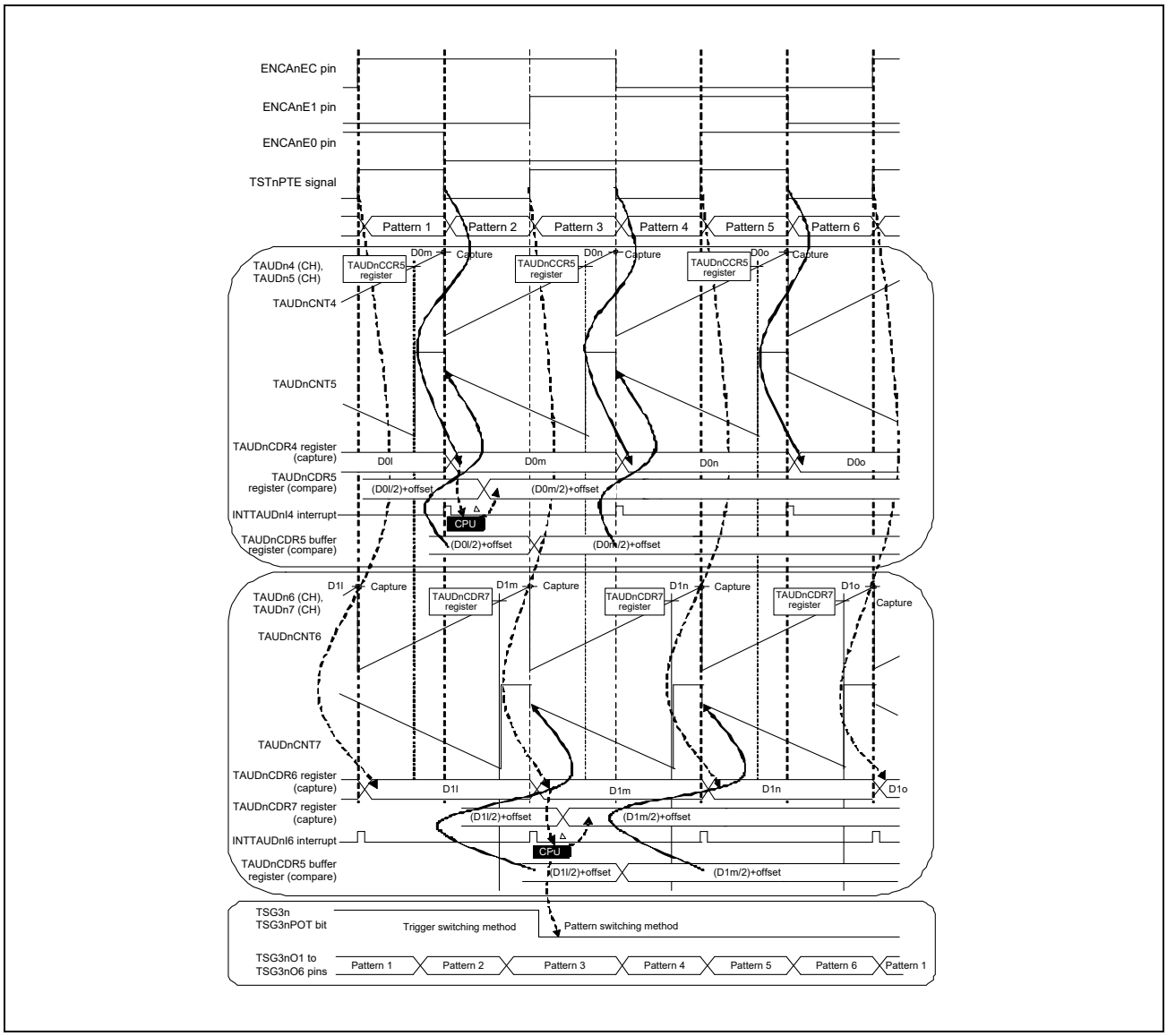


Figure 24.57 An Example of Switch Operation from Variable Phase Control to Fixed Phase Control 1

The output pattern is changed to the pattern switching method by changing the TSG3nPOT from high level to low level, and the fixed phase control 1 is enabled.

(5) Flow chart

The flow charts for this function are shown as follows.

Flow chart of main routine

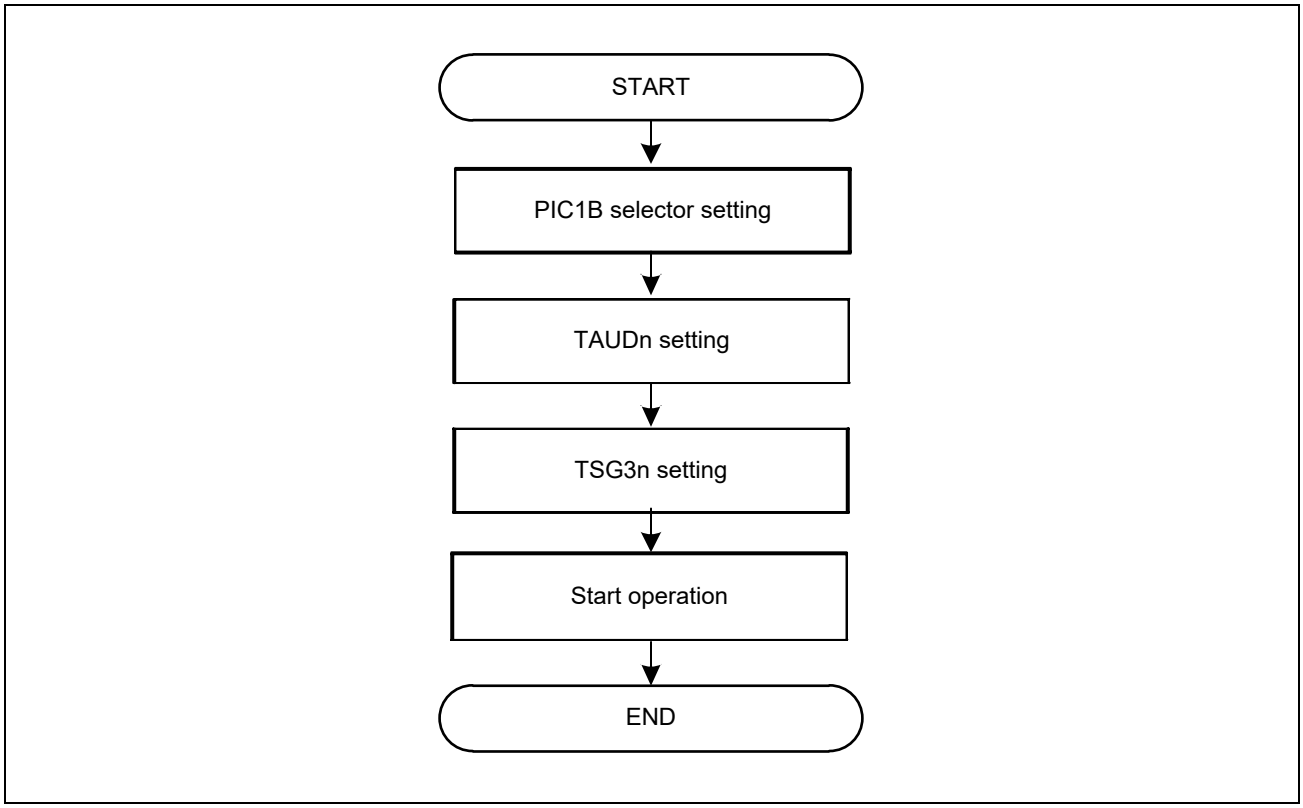
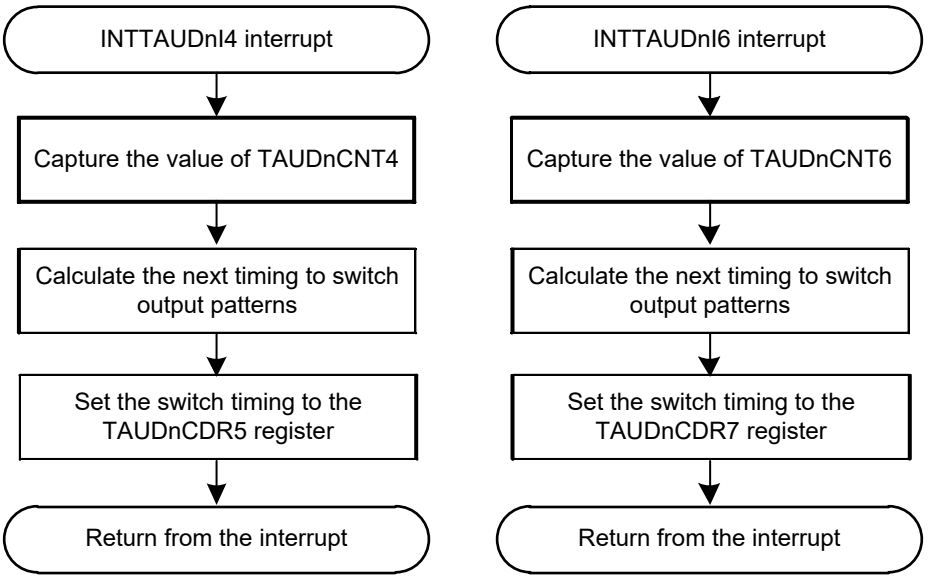


Figure 24.58 Setting Flow

Flow chart of rewriting of TAUDnCDR5 and TAUDnCDR7 during operation



Note: For the TAUDn settings, see **Section 18.4.12.5, Offset Trigger Output Function**.

Figure 24.59 TAUDnCDR5 and TAUDnCDR7 Rewrite Processing

The values of TSG3n registers used in this function are as follows.

TSG3nCTL0[7:0] = 0000_0011_B

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_0001\_xxx0\_0000_B$

TSG3nIOC0[7:0] = 0111_1110_B

TSG3nIOC1[7:0] = 0001_xxxx_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = 0011_1xx0_B

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 20, Motor Control Timer (TSG3)**.

24.2.3.12 Three-Phase Encoder Function

(1) Overview

The function allows three-phase external pattern inputs (TSG3nPTSI0 to TSG3nPTSI2) using ENCAn.

(2) Configuration

The three-phase encoder control function is realized by using TSG3n, ENCAn, and PIC1B in combination.

The following figure shows the block diagram of three-phase encoder control function.

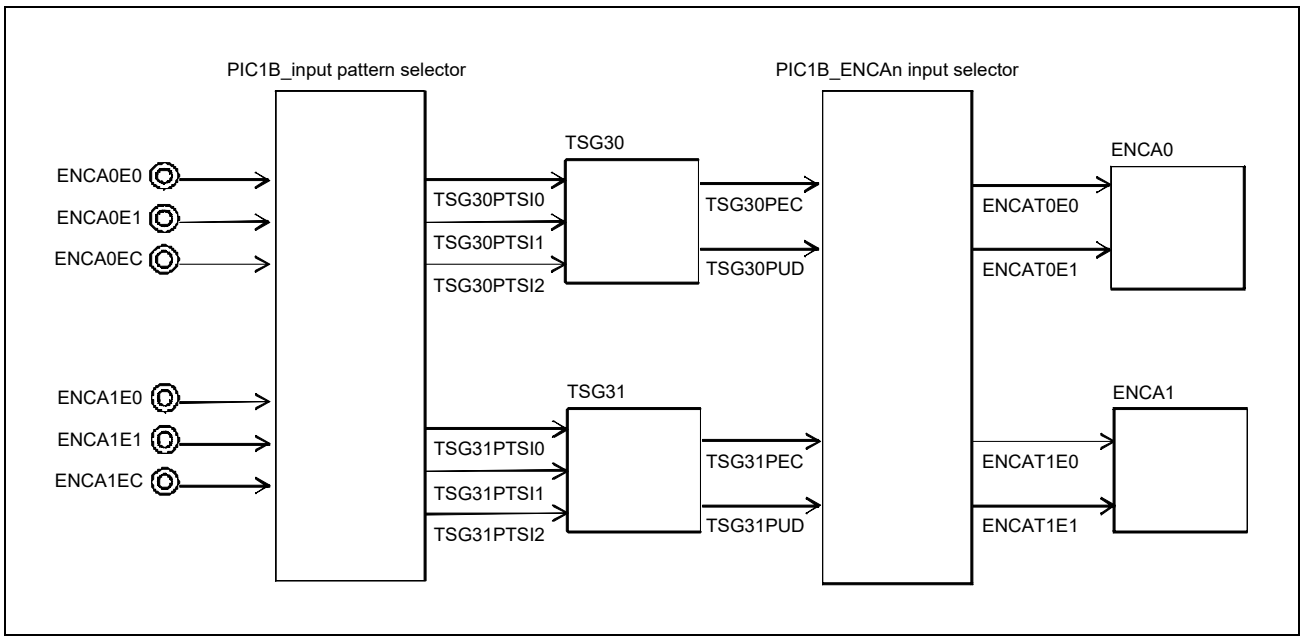


Figure 24.60 Block Diagram of Three-Phase Encoder Control Function

The configuration of this function is described as follows.

- [PIC1B_input pattern selector]
  ENCAnE0, ENCAnE1, and ENCAnEC pin inputs are selected and output to TSG3nPTSI0 to TSG3nPTSI2.
- [TSG3n]
  Patterns set in TSG3nPEC are output in response to the input of the TSG3nPTSI0 to TSG3nPTSI2 signals.
  TSG3nPUD is output depending on rotation in normal or reverse.
- [PIC1B_ENCAn input selector]
  TSG3nPEC is selected and output to ENCATnE0. TSG3nPUD is selected and output to ENCATnE1.
- [ENCAn]
  ENCATnE0 and ENCATnE1 are encoded.

(3) Registers

Block diagram of PIC1B is shown in the following figure.

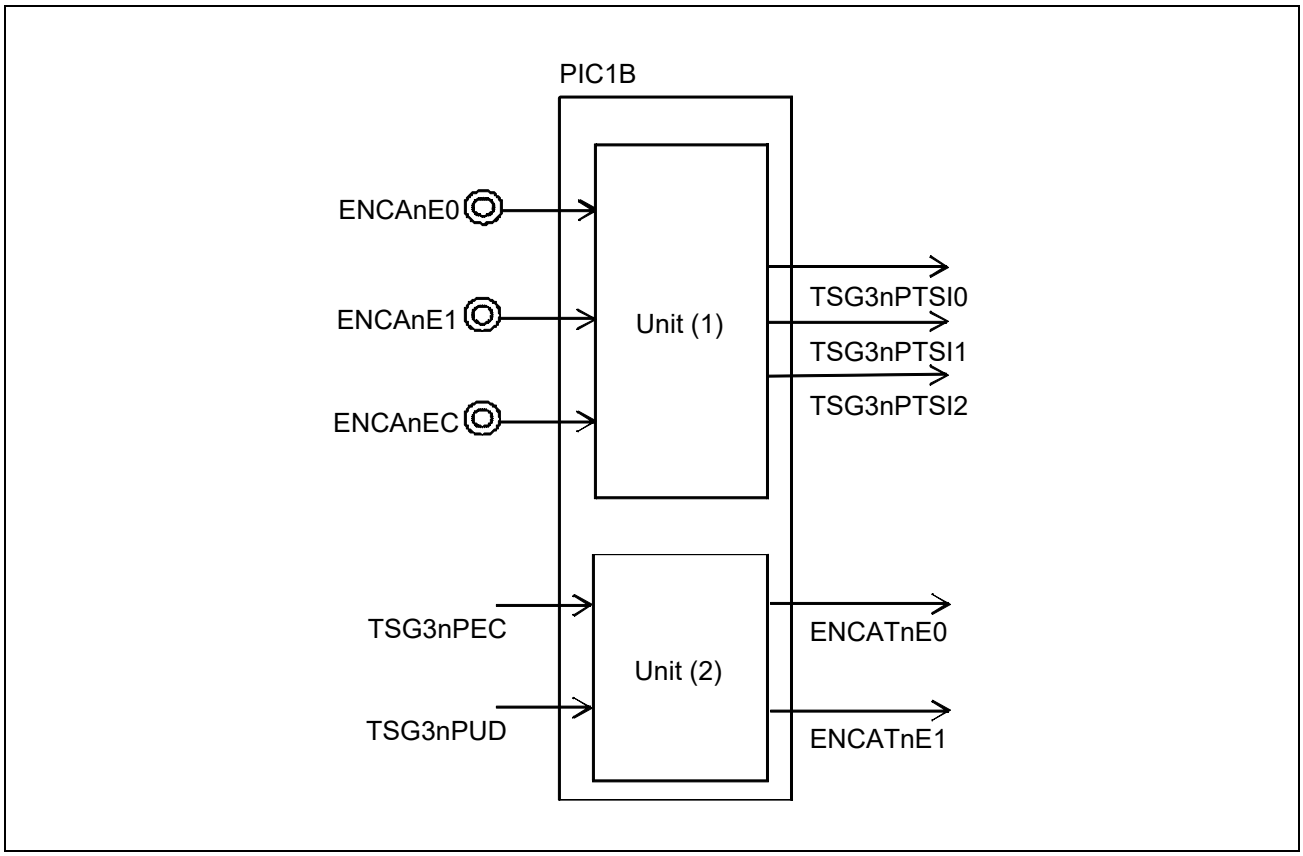


Figure 24.61 Block Diagram of PIC1B

The register settings for PIC1B to use this function are described as follows.

Unit (1): PIC1B_input pattern selector

The values to output ENCA0E0, ENCA0E1, and ENCA0EC as TSG30PTSI0 to TSG30PTSI2
PIC1BTSGHALLSEL[0] = 1_B
PIC1BREG50[0] = 0_B

The value to output ENCA1E0, ENCA1E1, and ENCA1EC as TSG31PTSI0 to TSG31PTSI2
PIC1BTSGHALLSEL[1] = 1_B
PIC1BREG51[0] = 1_B

Unit (2): PIC1B_ENCAn input selector

The values to output TSG30PEC and TSG30PUD as ENCAT0E0 and ENCAT0E1, respectively
PIC1BREG30[22] = 0_B
PIC1BREG30[1:0] = 11_B

The values to output TSG31PEC and TSG31PUD as ENCAT1E0 and ENCAT1E1, respectively
PIC1BREG30[9:8] = 10_B
PIC1BREG30[7:6] = 10_B

(4) Function

Detail of the three-phase encoder function is described as follows.

The following figure shows the timing diagram.

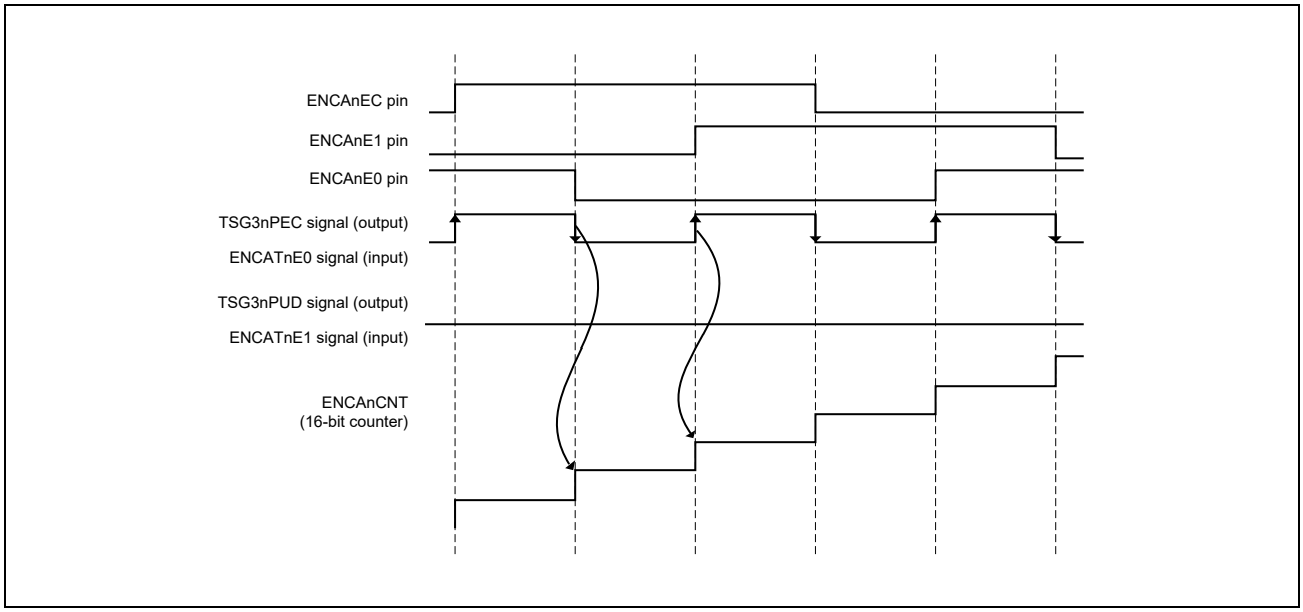


Figure 24.62 Timing Diagram of Three-Phase Encoder Function_ENCAnUDS1 and ENCAnUDS0 = 00_B

(1) When the low level is input to ENCAnE1, the count is incremented each time an active edge is input to ENCAnE0.

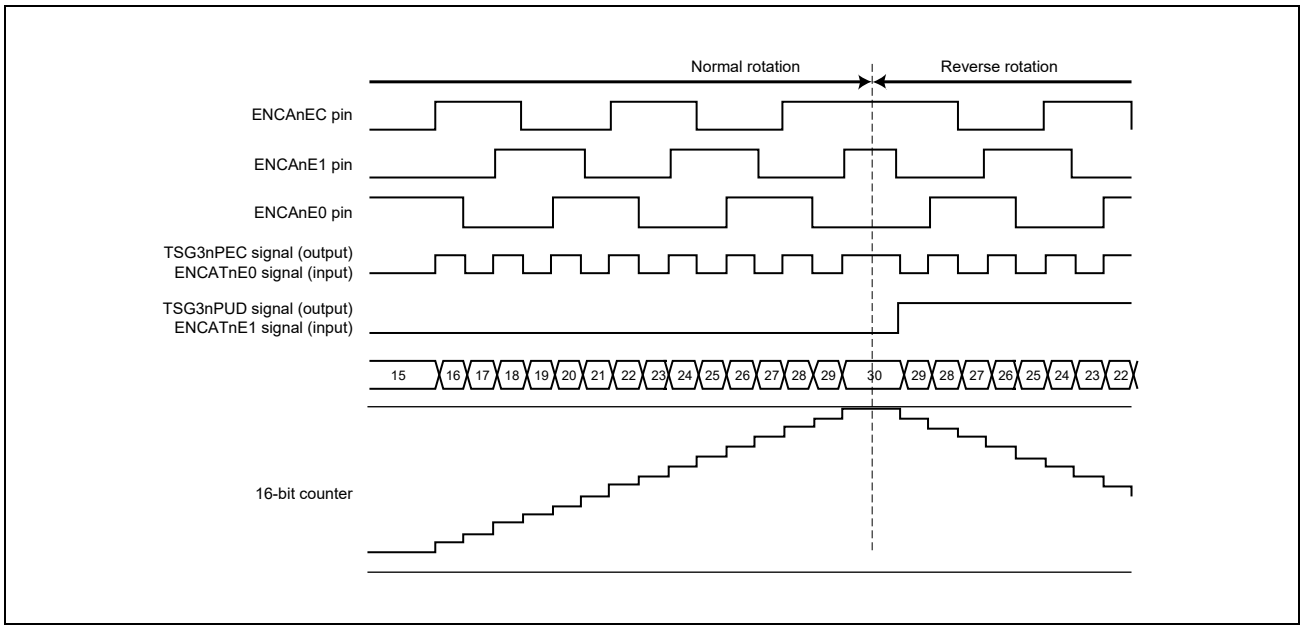


Figure 24.63 Timing Diagram of Three-Phase Encoder Function (Normal Rotation/Reverse Rotation)

(5) Flow chart

Set PIC before using the three-phase encoder control function.

The values of ENCAn registers used in this function are as follows.

ENCAnCTL[15:0] = $xx00\_00xx\_000x\_xx00_B$

ENCAnIOC1[7:0] = 0000_00xx_B*1

ENCAnCCR0 = (set any value)

ENCAnCCR1 = (set any value)

ENCAnCNT = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 23, Encoder Timer (ENCA)**.

Note 1. Except for 00_B (no edge detected) for ENCAnIOC1[1:0] because edge detection is necessary.

The values of TSG3n registers used in this function are as follows.

TSG3nCTL0[7:0] = 0000_0001_B

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_000x\_xxxx\_xxxx_B$

TSG3nIOC0[7:0] = $0xxx\_xxx0_B$

TSG3nIOC1[7:0] = 0001_xxxx_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = $0xxx\_xxx0_B$

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 20, Motor Control Timer (TSG3)**.

24.2.3.13 ENCA Input Select Function

(1) Overview

This function selects ENCAn input signals from among RDC3A1*[1], RDC3A0, encoder signal group 0 pin, and encoder signal group 1 pin.

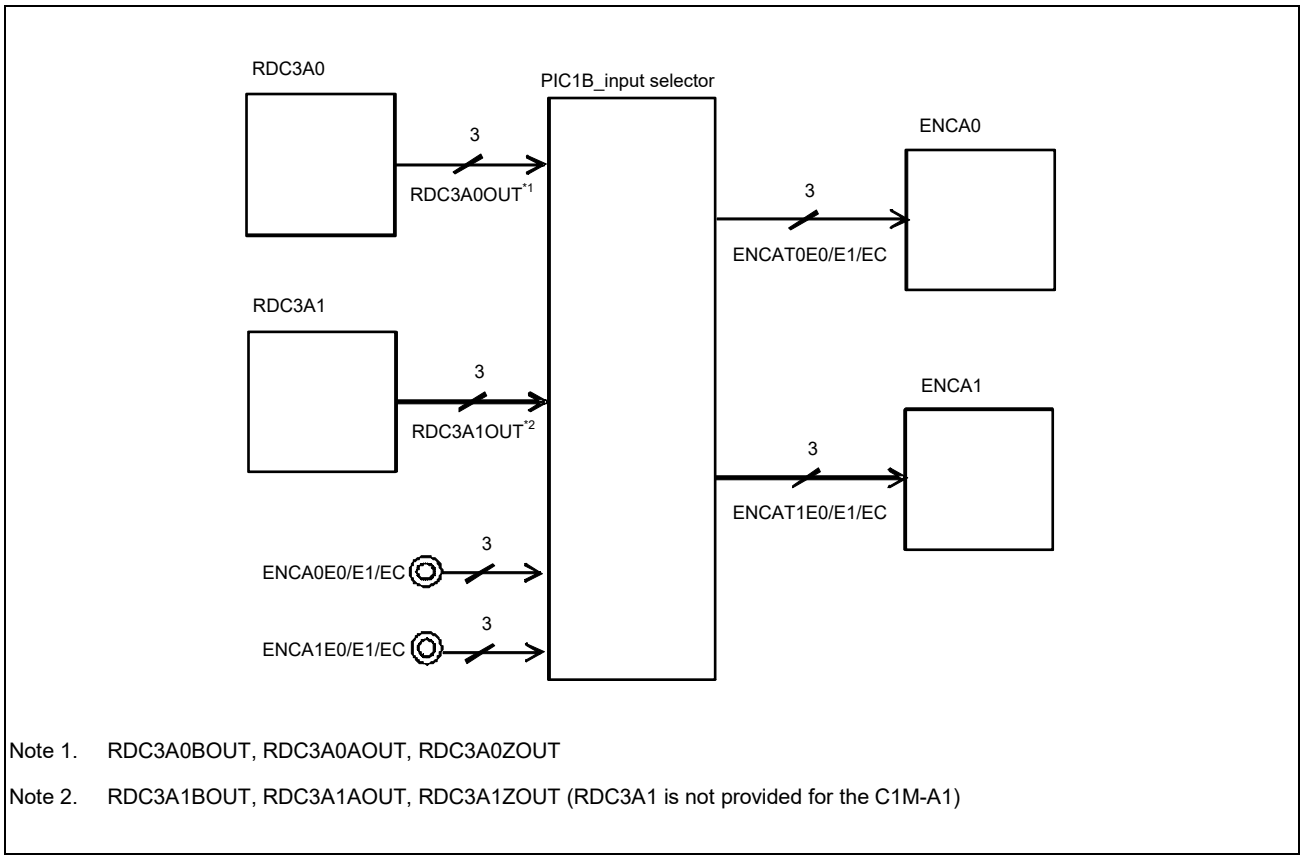
Connecting the same encoder signals to both ENCAn modules so that operation of the two is synchronized virtually increases the number of registers available for comparison and capture.

Note 1. RDC3A1 is not provided for the C1M-A1.

(2) Configuration

The ENCA input select function is realized by using RDC3An output signal and ENCAn input pin signal, and PIC1B in combination.

The following figure shows the block diagram of ENCA input select function.



Note 1. RDC3A0BOUT, RDC3A0AOUT, RDC3A0ZOUT

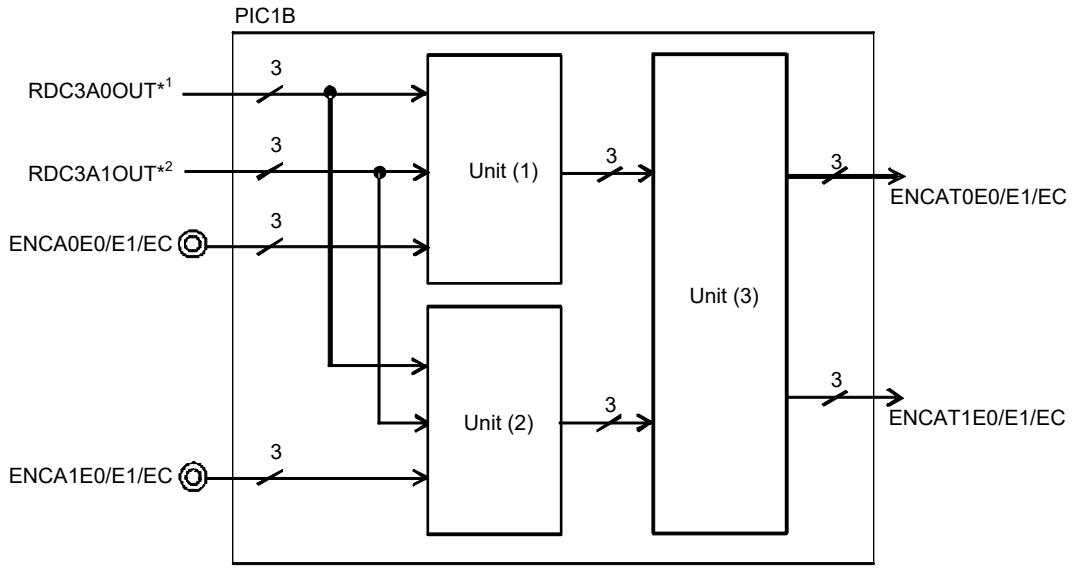
Note 2. RDC3A1BOUT, RDC3A1AOUT, RDC3A1ZOUT (RDC3A1 is not provided for the C1M-A1)

Figure 24.64 Block Diagram of ENCA Input Select Function

The signals to be output to ENCA0 and ENCA1 can be selected from the signals input to PIC1B.

(3) Registers

Block diagram of PIC1B is shown in the following figure.



Note 1. RDC3A0BOUT, RDC3A0AOUT, RDC3A0ZOUT

Note 2. RDC3A1BOUT, RDC3A1AOUT, RDC3A1ZOUT (RDC3A1 is not provided for the C1M-A1)

Figure 24.65 Block Diagram of PIC1B

The values of PIC1B registers used in this function are as follows.

PIC1B_input selector

- ENCA0 connection selector

  The value to select the signal to input to ENCAT0E0, ENCAT0E1, and ENCAT0EC in unit (3) via unit (1)

Table 24.78 Setting Value

Register Setting					
PIC1BREG30					
22	[17:16]	[1:0]	ENCAT0E0	ENCAT0E1	ENCAT0EC
0	01	00	RDC3A0BOUT	RDC3A0AOUT	RDC3A0ZOUT
0	10	00	RDC3A1BOUT*1	RDC3A1AOUT*1	RDC3A1ZOUT*1
0	00	00	ENCA0E0 pin input	ENCA0E1 pin input	ENCA0EC pin input

Note: Do not set the values other than the settings listed above for this function.

Note 1. Not supported in the C1M-A1.

The value to select the signal to input to ENCAT0E0, ENCAT0E1, and ENCAT0EC in unit (3) via unit (2)

Table 24.79 Setting Value

Register Setting				
PIC1BREG30				
22	[20:19]	ENCAT0E0	ENCAT0E1	ENCAT0EC
1	10	RDC3A0BOUT	RDC3A0AOUT	RDC3A0ZOUT
1	01	RDC3A1BOUT*1	RDC3A1AOUT*1	RDC3A1ZOUT*1
1	00	ENCA1E0 pin input	ENCA1E1 pin input	ENCA1EC pin input

Note: Do not set the values other than the settings listed above for this function.

Note 1. Not supported in the C1M-A1.

- ENCA1 connection selector

The value to select the signal to input to ENCAT1E0, ENCAT1E1, and ENCAT1EC in unit (3) via unit (1)

Table 24.80 Setting Value

Register Setting				
PIC1BREG30				
[17:16]	[11:6]	ENCAT1E0	ENCAT1E1	ENCAT1EC
10	100101	RDC3A0BOUT	RDC3A0AOUT	RDC3A0ZOUT
01	100101	RDC3A1BOUT*1	RDC3A1AOUT*1	RDC3A1ZOUT*1
00	100101	ENCA0E0 pin input	ENCA0E1 pin input	ENCA0EC pin input

Note: Do not set the values other than the settings listed above for this function.

Note 1. Not supported in the C1M-A1.

The value to select the signal to input to ENCAT1E0, ENCAT1E1, and ENCAT1EC in unit (3) via unit (2)

Table 24.81 Setting Value

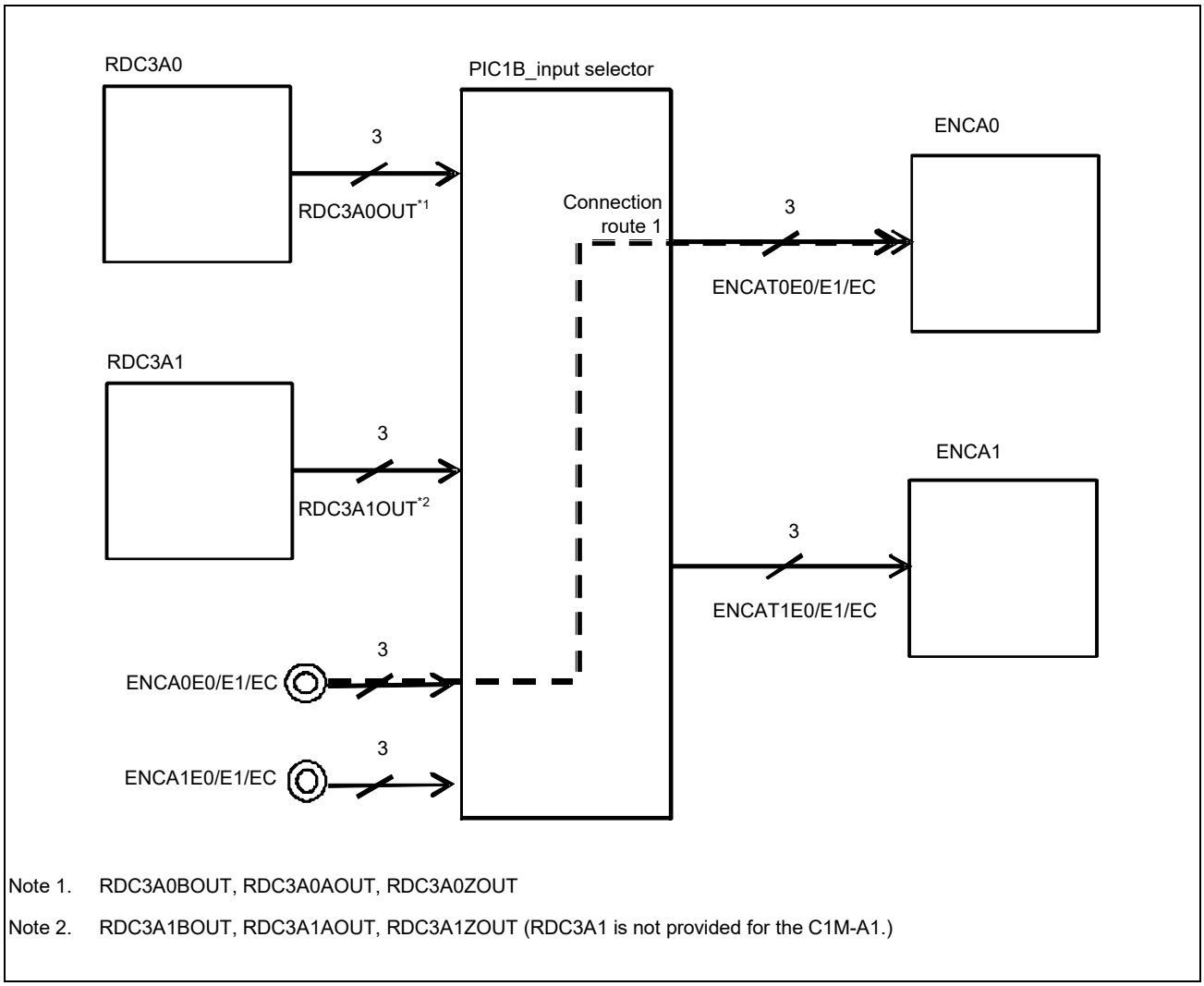
Register Setting				
PIC1BREG30				
[20:19]	[11:6]	ENCAT1E0	ENCAT1E1	ENCAT1EC
10	000000	RDC3A0BOUT	RDC3A0AOUT	RDC3A0ZOUT
01	000000	RDC3A1BOUT*1	RDC3A1AOUT*1	RDC3A1ZOUT*1
00	000000	ENCA1E0 pin input	ENCA1E1 pin input	ENCA1EC pin input

Note: Do not set the values other than the settings listed above for this function.

Note 1. Not supported in the C1M-A1.

(4) Function

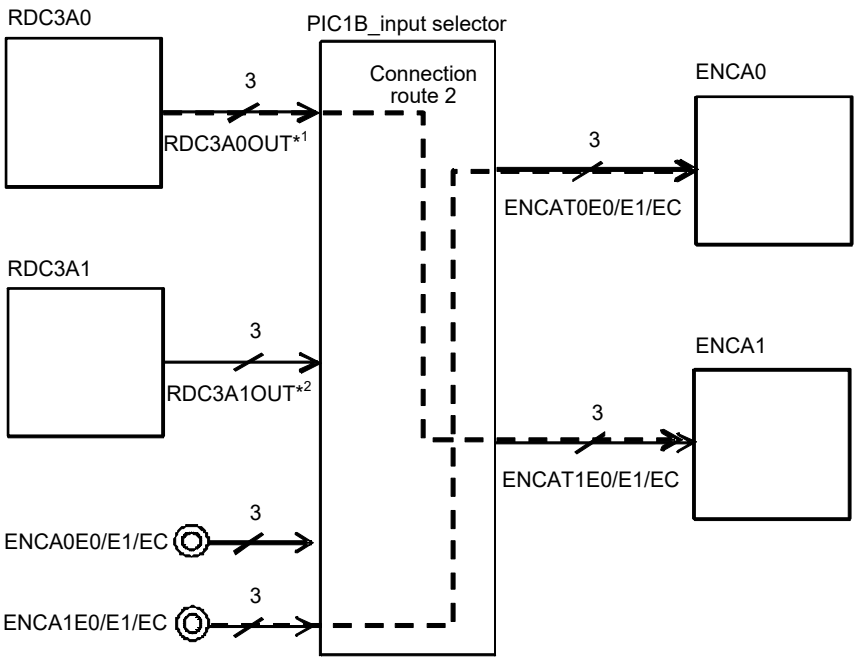
This function allows the connection route 1 and the connection route 2 as shown in the flowing figures.
Each connection route counts the selected input signal.



Note 1. RDC3A0BOUT, RDC3A0AOUT, RDC3A0ZOUT

Note 2. RDC3A1BOUT, RDC3A1AOUT, RDC3A1ZOUT (RDC3A1 is not provided for the C1M-A1.)

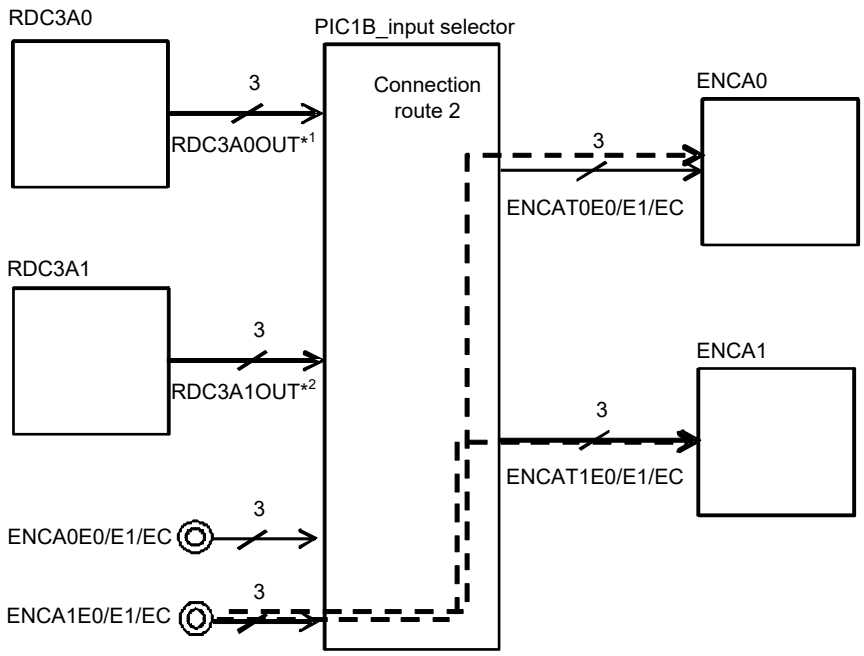
Figure 24.66 Example of Connection Route 1 (ENCA0 Pin Connected to ENCA0 Timer)



Note 1. RDC3A0BOUT, RDC3A0AOUT, RDC3A0ZOUT

Note 2. RDC3A1BOUT, RDC3A1AOUT, RDC3A1ZOUT (RDC3A1 is not provided for the C1M-A1.)

Figure 24.67 Example of Connection Route 2 (ENCA1 Pin and RDC3A0 Connected to ENCA0 and ENCA1, Respectively)



Note 1. RDC3A0BOUT, RDC3A0AOUT, RDC3A0ZOUT

Note 2. RDC3A1BOUT, RDC3A1AOUT, RDC3A1ZOUT (RDC3A1 is not provided for the C1M-A1.)

Figure 24.68 Example of Connection Route 2 (ENCA1 Pin Connected to ENCA0 and ENCA1)

(5) Flow chart

Set PIC before starting the encoder timer.

The values of ENCAn registers when selecting different signals for input to ENCA0 and ENCA1 are as follows.

ENCAnCTL[15:0] = $1000\_0000\_000x\_0101_B$ (ENCA pin input), $1000\_0000\_000x\_0111_B$ (RDC3A input)

ENCAnIOC0[7:0] = 0000_xxxx_B

ENCAnIOC1[7:0] = $xxxx\_xxxx_B$*1

"x" can be set arbitrarily. For the registers specifications, see **Section 23, Encoder Timer (ENCA)**.

Note 1. Except for 00_B (no edge detected) for ENCAnIOC1[3:2] and [1:0] because edge detection is necessary.

The values of ENCAn registers when selecting same signals for input to ENCA0 and ENCA1 are as follows.

ENCAnCTL[15:0] = $0x00\_000x\_x00x\_0xxx_B$ (ENCA pin input), $0x00\_000x\_x00x\_0x11_B$ (RDC3A input)

ENCAnIOC0[7:0] = 0000_xxxx_B

ENCAnIOC1[7:0] = $xxxx\_xxxx_B$*1

"x" can be set arbitrarily. For the registers specifications, see **Section 23, Encoder Timer (ENCA)**.

Note 1. Except for 00_B (no edge detected) for ENCAnIOC1[3:2] and [1:0] because edge detection is necessary.

24.2.3.14 TAUD Input Select Function

(1) Overview

The function selects TAUDn input signal to be input as TAUDnTTINm/m+1 signal from either TAUD0Im/m+1 signal or TAUD1Im/m+1 signal (m is an even number between 0 and 14).

(2) Configuration

The TAUD input select function is realized by using TAUDn input signals and PIC1B in combination.

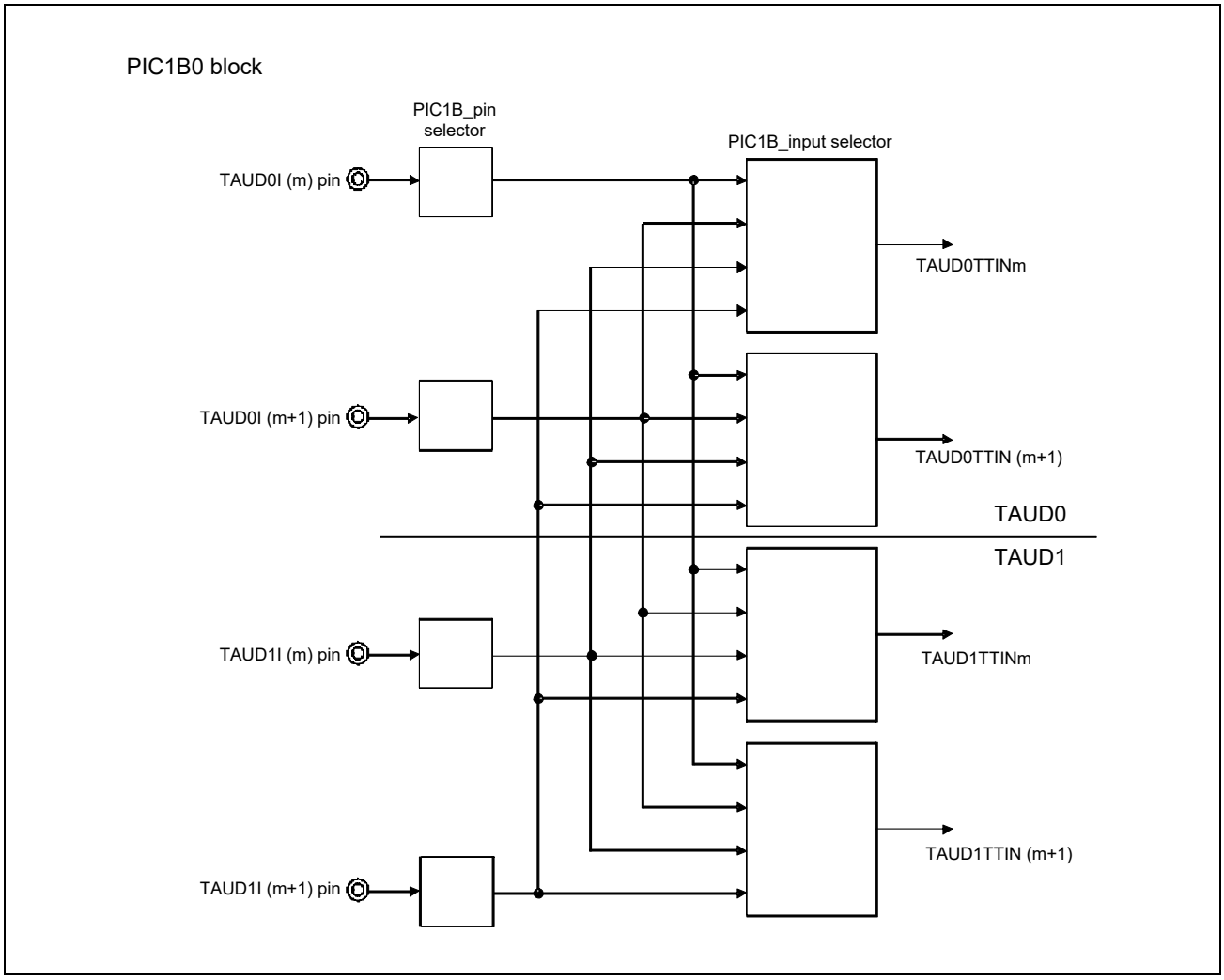
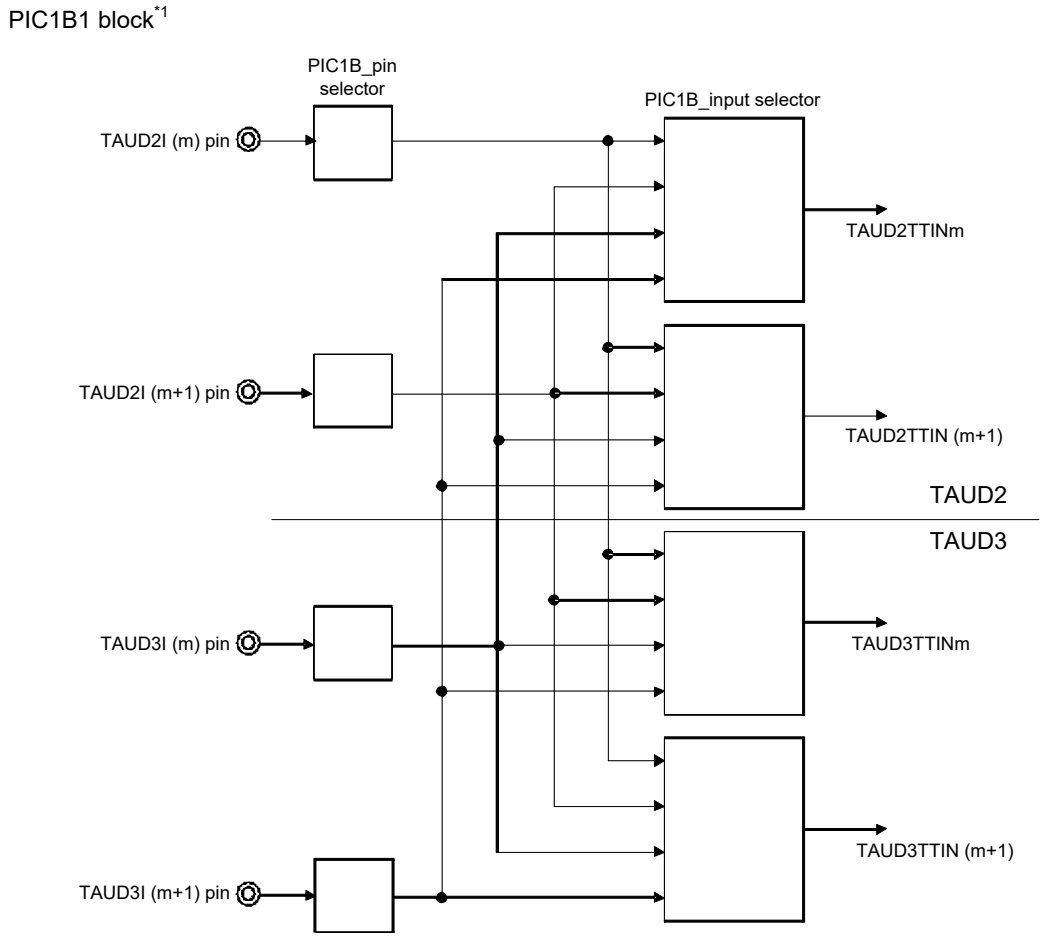


Figure 24.69 Block Diagram of TAUD Input Select Function



Note 1. The PIC1B1 blocks are not provided for the C1M-A1.

Figure 24.70 Block Diagram of TAUD Input Select Function

(3) Registers

The register settings for PIC1B to use this function are described as follows.

PIC1B_pin selector

Select TAUDn external channel input pin for output from PIC1B pin selector.

PIC1B_input selector

Select by the registers listed as follows.

PIC1BTAUD0SEL

PIC1BTAUD1SEL

PIC1BTAUD2SEL*[1]

PIC1BTAUD3SEL*[1]

PIC1BREG2n0

PIC1BREG2n2

PIC1BREG31

Note 1. This register is not provided for the C1M-A1.

For the details of the register settings, see **Sections 24.2.2.22 to 24.2.2.25, 24.2.2.34 to 24.2.2.37, 24.2.2.39, and 24.2.2.42**.

(4) Function

Detail of the function is described using selection of the TAUD0TTIN[1:0] signal as an example.

The following table lists an example of selection of the TAUD0TTIN[1:0] signals. Setting 000000 to PIC1BREG31[11:6], and setting 01 to PIC1BTAUD0SEL[3:2] and PIC1BTAUD0SEL[1:0] allows TAUD0I0 and TAUD0I1 signals to be input to TAUD0TTIN1 and TAUD0TTIN0 input pins of the TAUD0 timer. Setting 1 to PIC1BTAUD0SEL[3] and PIC1BTAUD0SEL[1] selects TIN pin signal of TAUD1.

Table 24.82 Example of Selection of the TAUD0TTIN0 Signals

Register Setting	
PIC1BTAUD0SEL	
[1:0]	TAUD0TTIN0
00_B	TAUD0I0 pin
01_B	TAUD0I1 pin
10_B	TAUD1I0 pin
11_B	TAUD1I1 pin

Table 24.83 Example of Selection of the TAUD0TTIN1 Signals

Register Setting	
PIC1BTAUD0SEL	
[3:2]	TAUD0TTIN1
00_B	TAUD0I1 pin
01_B	TAUD0I0 pin
10_B	TAUD1I1 pin
11_B	TAUD1I0 pin

(5) Flow chart

Set PIC1B before starting the TAUDn timer.

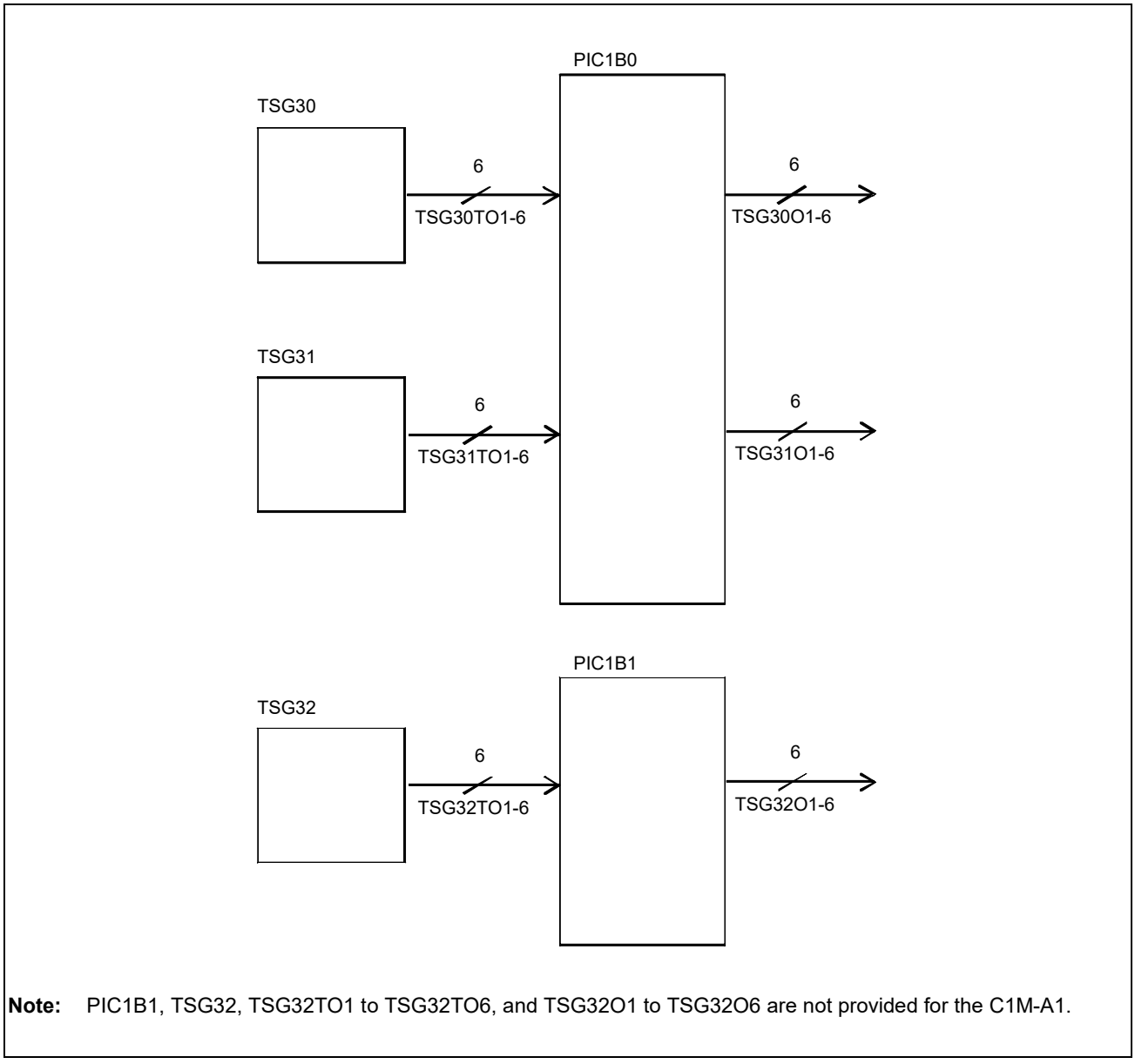
24.2.3.15 Switch Function between TSG Output and Low/High Level Output

(1) Overview

The function allows switching of outputs between TSG output and low/high level output at desired timing.

(2) Configuration

TSG is output via PIC1B. With this function, the TSG output is delayed for one cycle of the clock (CLKC_HSB).



Note: PIC1B1, TSG32, TSG32TO1 to TSG32TO6, and TSG32O1 to TSG32O6 are not provided for the C1M-A1.

Figure 24.71 Block Diagram of Switch Function between TSG Output and Low/High-Level Output

(3) Registers

The block diagram of PIC1B is shown in the following figure.

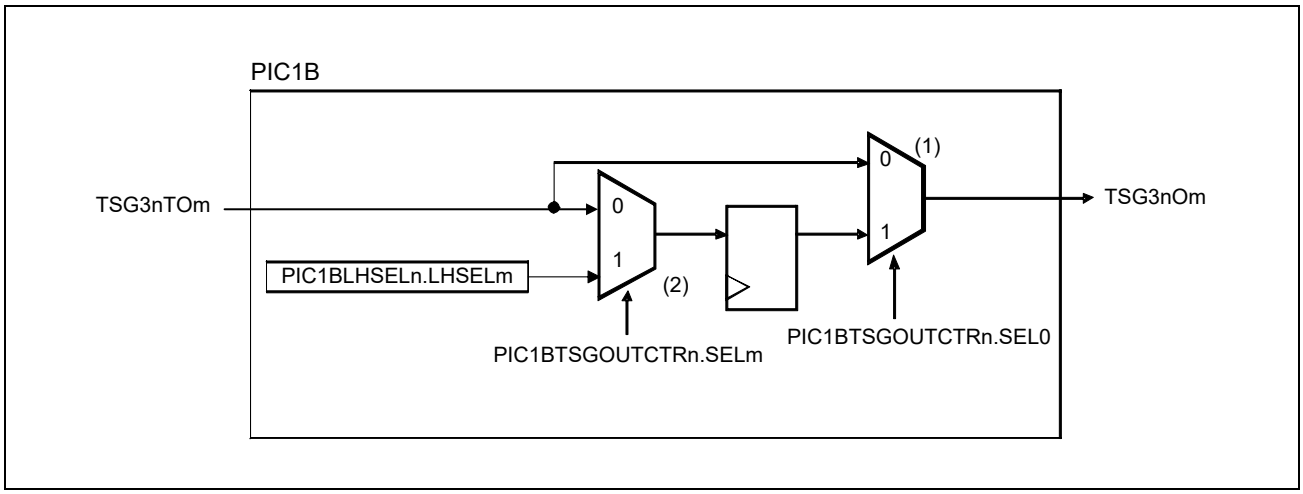


Figure 24.72 Block Diagram of PIC1B

The values of PIC1B registers used in this function are as follows.

(1) Switch on and off the function

PIC1BTSGOUTCTRn[0] = 0_B (turned off), 1_B (turned on)

(2) Switch between low level output and high level output

PIC1BLHSELn[m] = 0_B (output of low level), 1_B (output of high level)

PIC1BTSGOUTCTRn.SELm = 0_B (TSG3n output), 1_B (output of low/high level)

CAUTION

Set (1) before starting TSG3n. Changing at operation is prohibited.

(4) Function

Detail of the function is described using an example timing of switch function.

The following figure shows the timing diagram.

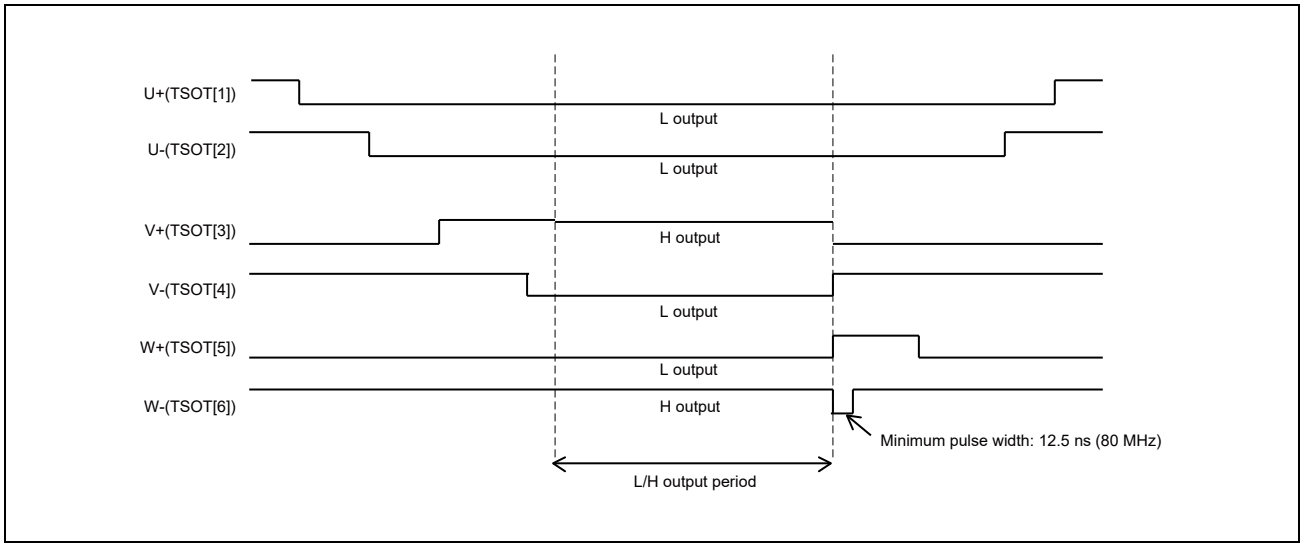
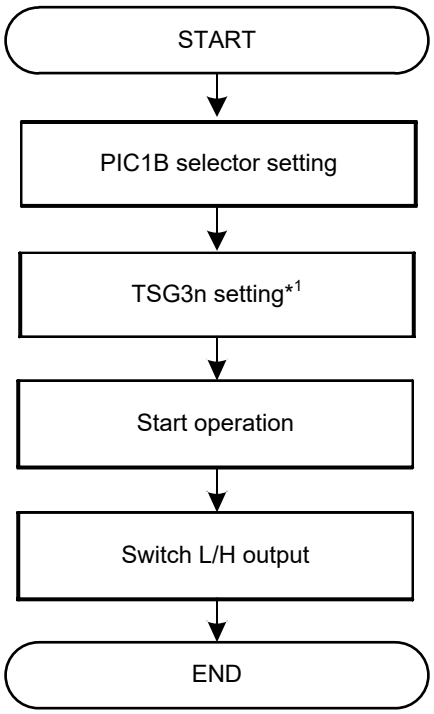


Figure 24.73 TSG Output Switching Timing

The function allows switching of outputs between TSG output and low/high level output at desired timing. The six output phases can be switched between high level and low level both individually and simultaneously. The minimum pulse (internal pulse) of CLKC_HSB may be generated depending on the switch timing.

(5) Flow chart

The flow charts for this function are shown as follows.



Note 1. For the TSG3n settings, see **Section 20, Motor Control Timer (TSG3)**.

Figure 24.74 Setting Flow

24.2.3.16 Hi-Z Control Function

(1) Overview

The function disconnects three-phase output signal and changes to Hi-Z state.

For the detail of the purpose and operation of the Hi-Z control function, see **Section 21.4.1, Asynchronous Hi-Z Control Function**.

(2) Configuration

The ESOn, ERROROUTZ, and INTTSG3nIER signals are masked and OR'ed in PIC1B, and output to TAPAn as the signal for Hi-Z control.

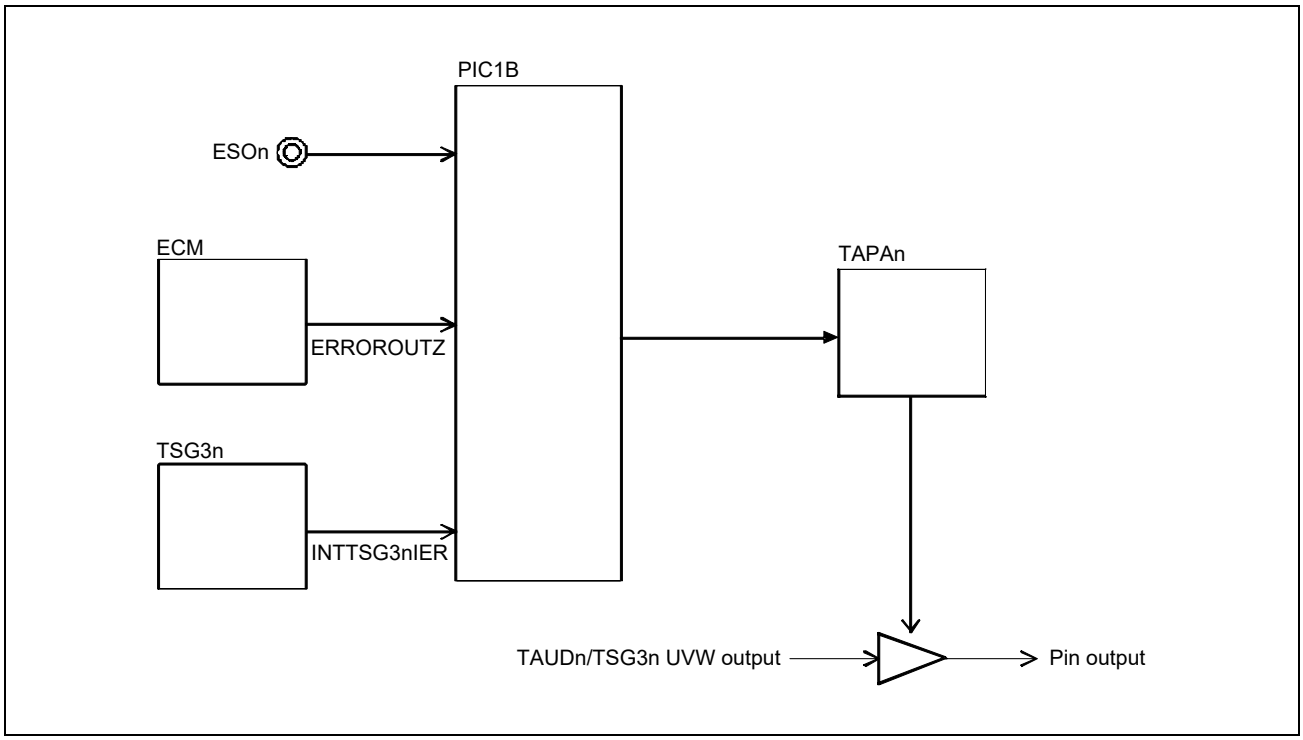


Figure 24.75 Block Diagram of Hi-Z Control

(3) Registers

The block diagram of PIC1B is shown in the following figure.

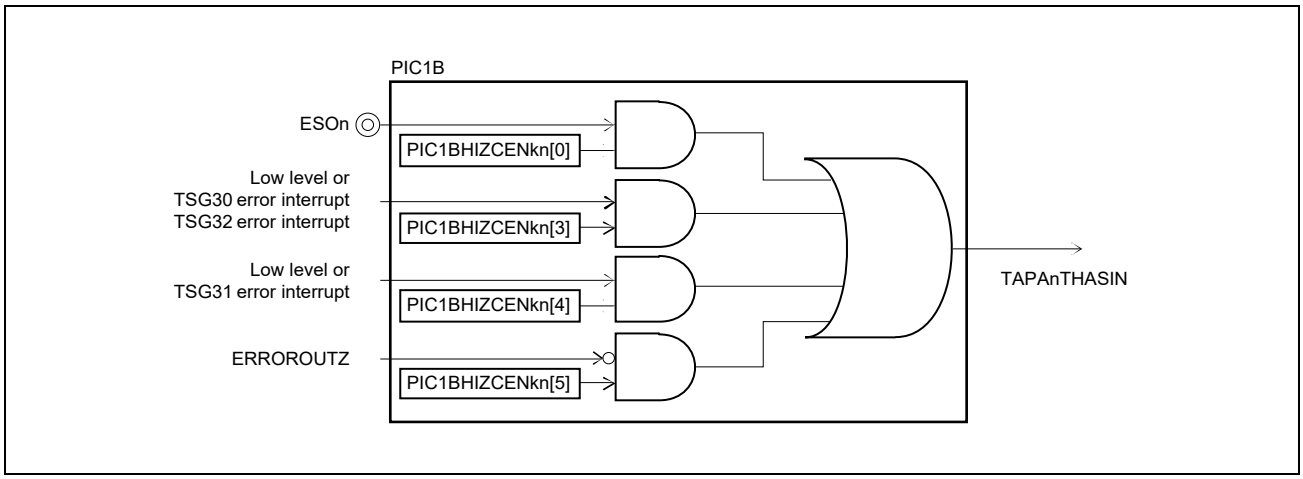


Figure 24.76 Block Diagram of PIC1B

The register settings for PIC1B to be used with this function are described as follows.

Table 24.84 Setting Value

Register Name	Bit 5	Bit 4	Bit 3	Bit 0	Register to be Controlled	TAPA to be Connected
PIC1BHIZCEN00	ERROROUTZ	—	—	ESO0	TAUD0	TAPA0
PIC1BHIZCEN01	ERROROUTZ	—	—	ESO1	TAUD1	TAPA1
PIC1BHIZCEN02	ERROROUTZ	—	INTTSG30IER	ESO3	TSG30	TAPA3
PIC1BHIZCEN03	ERROROUTZ	INTTSG31IER	—	ESO4	TSG31	TAPA4
PIC1BHIZCEN10*1	ERROROUTZ	—	—	ESO2	TAUD2	TAPA2
PIC1BHIZCEN12*1	ERROROUTZ	—	INTTSG32IER	ESO5	TSG32	TAPA5

Bit 5 = 1_B (enabled), 0_B (disabled): ERROROUTZ
Bit 4 = 1_B (enabled), 0_B (disabled): INTTSG31IER
Bit 3 = 1_B (enabled), 0_B (disabled): INTTSG30IER or INTTSG32IER
Bit 0 = 1_B (enabled), 0_B (disabled): ESOm

Note 1. PIC1BHIZCEN10, PIC1BHIZCEN12, ESO2, ESO5, TAUD2, TSG32, TAPA2, TAPA5 are not provided for the C1M-A1.

(4) Function

The signals ESOn pin, ERROROUTZ, and INTTSG3nIER are masked and OR'ed in PIC1B, and output to TAPAn.

For Hi-Z control by TAPA, see **Section 21.4.2.2, Basic Operation**.

(5) Flow chart

Set PIC1B before starting Hi-Z control.

For the operation flow of TAPA, see **Section 21.4.2.3, Operating Procedure**.

24.3 Peripheral Interconnection 2 (PIC2D)

24.3.1 Overview

24.3.1.1 Functional Overview

The peripheral interconnection 2 (PIC2D) allows ADCC hardware trigger signal to be generated using the internal and external trigger signals output from individual IPs.

24.3.2 Registers

24.3.2.1 List of Registers

PIC2D registers are listed in the following table.

See **Section 24.1.2, Register Base Address** for <PIC2D_base>.

Table 24.85 List of Registers (1/2)

Module Name	Register Name	Symbol	Address
PIC2D	A/D converter 0 trigger select control register 0	PIC2DADCC0TSEL0	<PIC2D_base> + 00_H
PIC2D	A/D converter 0 trigger select control register 1	PIC2DADCC0TSEL1	<PIC2D_base> + 04_H
PIC2D	A/D converter 0 trigger select control register 2	PIC2DADCC0TSEL2	<PIC2D_base> + 08_H
PIC2D	A/D converter 0 trigger select control register 3	PIC2DADCC0TSEL3	<PIC2D_base> + $0C_H$
PIC2D	A/D converter 0 trigger select control register 4	PIC2DADCC0TSEL4	<PIC2D_base> + 10_H
PIC2D	A/D converter 0 trigger edge select control register	PIC2DADCC0EDGSEL	<PIC2D_base> + $1C_H$
PIC2D	A/D converter 1 trigger select control register 0	PIC2DADCC1TSEL0	<PIC2D_base> + 20_H
PIC2D	A/D converter 1 trigger select control register 1	PIC2DADCC1TSEL1	<PIC2D_base> + 24_H
PIC2D	A/D converter 1 trigger select control register 2	PIC2DADCC1TSEL2	<PIC2D_base> + 28_H
PIC2D	A/D converter 1 trigger select control register 3	PIC2DADCC1TSEL3	<PIC2D_base> + $2C_H$
PIC2D	A/D converter 1 trigger select control register 4	PIC2DADCC1TSEL4	<PIC2D_base> + 30_H
PIC2D	A/D converter 1 trigger edge select control register	PIC2DADCC1EDGSEL	<PIC2D_base> + $3C_H$
PIC2D	A/D converter 2 trigger select control register 0	PIC2DADCC2TSEL0	<PIC2D_base> + 40_H
PIC2D	A/D converter 2 trigger select control register 1	PIC2DADCC2TSEL1	<PIC2D_base> + 44_H
PIC2D	A/D converter 2 trigger select control register 2	PIC2DADCC2TSEL2	<PIC2D_base> + 48_H
PIC2D	A/D converter 2 trigger select control register 3	PIC2DADCC2TSEL3	<PIC2D_base> + $4C_H$
PIC2D	A/D converter 2 trigger select control register 4	PIC2DADCC2TSEL4	<PIC2D_base> + 50_H
PIC2D	A/D converter 2 trigger edge select control register	PIC2DADCC2EDGSEL	<PIC2D_base> + $5C_H$
PIC2D	Common to ADCC0 to ADCC2		
PIC2D	A/D converter trigger output control register 400	PIC2DADTEN400	<PIC2D_base> + 80_H
PIC2D	A/D converter trigger output control register 401	PIC2DADTEN401	<PIC2D_base> + 84_H
PIC2D	A/D converter trigger output control register 402	PIC2DADTEN402	<PIC2D_base> + 88_H
PIC2D	A/D converter trigger output control register 403	PIC2DADTEN403	<PIC2D_base> + $8C_H$
PIC2D	A/D converter trigger output control register 404	PIC2DADTEN404	<PIC2D_base> + 90_H
PIC2D	A/D converter trigger output control register 410	PIC2DADTEN410	<PIC2D_base> + $A0_H$
PIC2D	A/D converter trigger output control register 411	PIC2DADTEN411	<PIC2D_base> + $A4_H$
PIC2D	A/D converter trigger output control register 412	PIC2DADTEN412	<PIC2D_base> + $A8_H$
PIC2D	A/D converter trigger output control register 413	PIC2DADTEN413	<PIC2D_base> + AC_H
PIC2D	A/D converter trigger output control register 414	PIC2DADTEN414	<PIC2D_base> + $B0_H$
PIC2D	A/D converter trigger output control register 420*1	PIC2DADTEN420	<PIC2D_base> + $C0_H$

Table 24.85 List of Registers (2/2)

Module Name	Register Name	Symbol	Address
PIC2D	A/D converter trigger output control register 421*1	PIC2DADTEN421	<PIC2D_base> + $C4_H$
PIC2D	A/D converter trigger output control register 422*1	PIC2DADTEN422	<PIC2D_base> + $C8_H$
PIC2D	A/D converter trigger output control register 423*1	PIC2DADTEN423	<PIC2D_base> + CC_H
PIC2D	A/D converter trigger output control register 424*1	PIC2DADTEN424	<PIC2D_base> + $D0_H$
PIC2D	A/D converter trigger output control register 430*1	PIC2DADTEN430	<PIC2D_base> + $E0_H$
PIC2D	A/D converter trigger output control register 431*1	PIC2DADTEN431	<PIC2D_base> + $E4_H$
PIC2D	A/D converter trigger output control register 432*1	PIC2DADTEN432	<PIC2D_base> + $E8_H$
PIC2D	A/D converter trigger output control register 433*1	PIC2DADTEN433	<PIC2D_base> + EC_H
PIC2D	A/D converter trigger output control register 434*1	PIC2DADTEN434	<PIC2D_base> + $F0_H$

Note 1. This register is not provided for the C1M-A1.

24.3.2.2 PIC2DADCCnTSELx ─ A/D Converter n Trigger Select Control Register x

The PIC2DADCCnTSELx register selects triggers for ADCCn scan group x (n = 0 to 2; x = 0 to 4).

Access: Readable/writable in 32-bit units.

Address: <PIC2D_base> + 00_H (n = 0, x = 0), <PIC2D_base> + 04_H (n = 0, x = 1), <PIC2D_base> + 08_H (n = 0, x = 2), <PIC2D_base> + $0C_H$ (n = 0, x = 3), <PIC2D_base> + 10_H (n = 0, x = 4), <PIC2D_base> + 20_H (n = 1, x = 0), <PIC2D_base> + 24_H (n = 1, x = 1), <PIC2D_base> + 28_H (n = 1, x = 2), <PIC2D_base> + $2C_H$ (n = 1, x = 3), <PIC2D_base> + 30_H (n = 1, x = 4) <PIC2D_base> + 40_H (n = 2, x = 0), <PIC2D_base> + 44_H (n = 2, x = 1), <PIC2D_base> + 48_H (n = 2, x = 2), <PIC2D_base> + $4C_H$ (n = 2, x = 3), <PIC2D_base> + 50_H (n = 2, x = 4)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PIC2DADCCnTSELx[27:24]				—	—	—	PIC2DADCCnTSELx[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2DADCCnTSELx[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.86 PIC2DADCCnTSELx Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, the value after reset is read. When writing, write the value after reset.
27	PIC2DADCCnTSELx27	Selects the RDC3A1 excitation timer (ET) A/D conversion start trigger signal output as the trigger source for ADCCn scan group x.*1 0: Not selected 1: Selected
26	PIC2DADCCnTSELx26	Selects the RDC3A1 compare 0 match interrupt request signal as the trigger source for ADCCn scan group x.*1 0: Not selected 1: Selected
25	PIC2DADCCnTSELx25	Selects the RDC3A0 excitation timer (ET) A/D conversion start trigger signal output as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
24	PIC2DADCCnTSELx24	Selects the RDC3A0 compare 0 match interrupt request signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
23 to 21	Reserved	When read, the value after reset is read. When writing, write the value after reset.
20	PIC2DADCCnTSELx20	Selects the ADC scan group conversion start signal of EMU3 channel 1, EMU31 ADTRG, as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected

Table 24.86 PIC2DADCCnTSELx Register Contents (2/3)

Bit Position	Bit Name	Function
19	PIC2DADCCnTSELx19	Selects the ADC scan group conversion start signal of EMU3 channel 0, EMU30 ADTRG, as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
18	PIC2DADCCnTSELx18	Selects the TAPA2TADOUT1 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
17	PIC2DADCCnTSELx17	Selects the TAPA2TADOUT0 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
16	PIC2DADCCnTSELx16	Selects the TAPA1TADOUT1 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
15	PIC2DADCCnTSELx15	Selects the TAPA1TADOUT0 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
14	PIC2DADCCnTSELx14	Selects the TAPA0TADOUT1 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
13	PIC2DADCCnTSELx13	Selects the TAPA0TADOUT0 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
12	PIC2DADCCnTSELx12	Selects the ADTRGnZ pin as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
11	PIC2DADCCnTSELx11	Selects the TSG2TSTADT1 signal as the trigger source for ADCCn scan group x.*1 0: Not selected 1: Selected
10	PIC2DADCCnTSELx10	Selects the TSG2TSTADT0 signal as the trigger source for ADCCn scan group x.*1 0: Not selected 1: Selected
9	PIC2DADCCnTSELx09	Selects the TSG1TSTADT1 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
8	PIC2DADCCnTSELx08	Selects the TSG1TSTADT0 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
7	PIC2DADCCnTSELx07	Selects the TSG0TSTADT1 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
6	PIC2DADCCnTSELx06	Selects the TSG0TSTADT0 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
5	PIC2DADCCnTSELx05	Selects the ENCAT1INT1 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected

Table 24.86 PIC2DADCCnTSELx Register Contents (3/3)

Bit Position	Bit Name	Function
4	PIC2DADCCn TSELx04	Selects the ENCAT0INT1 signal as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
3	PIC2DADCCn TSELx03	Selects the trigger selected by the PIC2DADTEN43x register as the trigger source for ADCCn scan group x.*1 0: Not selected 1: Selected
2	PIC2DADCCn TSELx02	Selects the trigger selected by the PIC2DADTEN42x register as the trigger source for ADCCn scan group x.*1 0: Not selected 1: Selected
1	PIC2DADCCn TSELx01	Selects the trigger selected by the PIC2DADTEN41x register as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected
0	PIC2DADCCn TSELx00	Selects the trigger selected by the PIC2DADTEN40x register as the trigger source for ADCCn scan group x. 0: Not selected 1: Selected

Note 1. Set this bit to 0 as this register is not provided for the C1M-A1.

24.3.2.3 PIC2DADCCnEDGSEL ─ A/D Converter n Trigger Edge Control Register

The PIC2DADCCnEDGSEL register selects an effective edge for the one-shot pulse generation circuit which generates an ADCC trigger.

The ADC external pin trigger is input in negative logic, but it is converted to positive logic for trigger source selection. Since edge detection is made for a trigger source after selection, note that the definition of an edge is reversed for an ADC external pin signal. (Setting 00 enables selection of a falling edge of ADC external pin triggers ADTRG0Z, ADTRG1Z, and ADTRG2Z).

Access: Readable/writable in 16-bit units.

Address: <PIC2D_base> + $1C_H$ (n = 0), <PIC2D_base> + $3C_H$ (n = 1), <PIC2D_base> + $5C_H$ (n = 2)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIC2DADCCnEDGSEL[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.87 PIC2DADCCnEDGSEL Register Contents

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after reset is read. When writing, write the value after reset.
9, 8	PIC2DADCCn EDGSEL[9:8]	Select an effective edge of ADCCn scan group 4. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
7, 6	PIC2DADCCn EDGSEL[7:6]	Select an effective edge of ADCCn scan group 3. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
5, 4	PIC2DADCCn EDGSEL[5:4]	Select an effective edge of ADCCn scan group 2. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
3, 2	PIC2DADCCn EDGSEL[3:2]	Select an effective edge of ADCCn scan group 1. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
1, 0	PIC2DADCCn EDGSEL[1:0]	Select an effective edge of ADCCn scan group 0. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)

24.3.2.4 PIC2DADTEN4nx ─ A/D Converter Trigger Output Select Control Register*1

The PIC2DADTEN4nx register enables selecting a trigger source from TAUDn channel m as the ADCC trigger (n = 0 to 3; x = 0 to 4). This register is common to ADCC0 to ADCC2.

Access: Readable/writable in 16-bit units.

Address: <PIC2D_base> + 80_H (n = 0, x = 0), <PIC2D_base> + 84_H (n = 0, x = 1), <PIC2D_base> + 88_H (n = 0, x = 2), <PIC2D_base> + $8C_H$ (n = 0, x = 3), <PIC2D_base> + 90_H (n = 0, x = 4), <PIC2D_base> + $A0_H$ (n = 1, x = 0), <PIC2D_base> + $A4_H$ (n = 1, x = 1), <PIC2D_base> + $A8_H$ (n = 1, x = 2), <PIC2D_base> + AC_H (n = 1, x = 3), <PIC2D_base> + $B0_H$ (n = 1, x = 4), <PIC2D_base> + $C0_H$ (n = 2, x = 0), <PIC2D_base> + $C4_H$ (n = 2, x = 1), <PIC2D_base> + $C8_H$ (n = 2, x = 2), <PIC2D_base> + CC_H (n = 2, x = 3), <PIC2D_base> + $D0_H$ (n = 2, x = 4), <PIC2D_base> + $E0_H$ (n = 3, x = 0), <PIC2D_base> + $E4_H$ (n = 3, x = 1), <PIC2D_base> + $E8_H$ (n = 3, x = 2), <PIC2D_base> + EC_H (n = 3, x = 3), <PIC2D_base> + $F0_H$ (n = 3, x = 4)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2DADTEN4nx[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.88 PIC2DADTEN4nx Register Contents

Bit Position	Bit Name	Function
15 to 0	PIC2DADTEN4nxm (m = 0 to 15)	Set a trigger source from TAUDn channel m. 0: Trigger source of TAUDn channel m cannot be selected as the ADCC trigger. 1: Trigger source of TAUDn channel m can be selected as the ADCC trigger.

Note 1. The PIC2DADTEN42x registers (TAUD2) and the PIC2DADTEN43x registers (TAUD3) are not provided for the C1M-A1.

24.3.3 Function

24.3.3.1 ADCC Trigger Select Function

(1) Overview

The function allows generation of ADCC hardware trigger signal for individual channel groups by the signals from each IP. The IPs can be selected from among TAUD0, TAUD1, TAUD2*[1], TAUD3*[1], ENCA0, ENCA1, TSG30, TSG31, TSG32*[1], TAPA0, TAPA1, TAPA2, EMU3, RDC3A0, and RDC3A1*[1].

The external trigger signal (ADTRG) is active low and it is converted by PIC2D.

Note 1. This IP is not provided for the C1M-A1.

(2) Configuration

The ADCC trigger select function is realized by using individual IPs and PIC2D in combination. The following figure shows the block diagram of ADCC trigger select function.

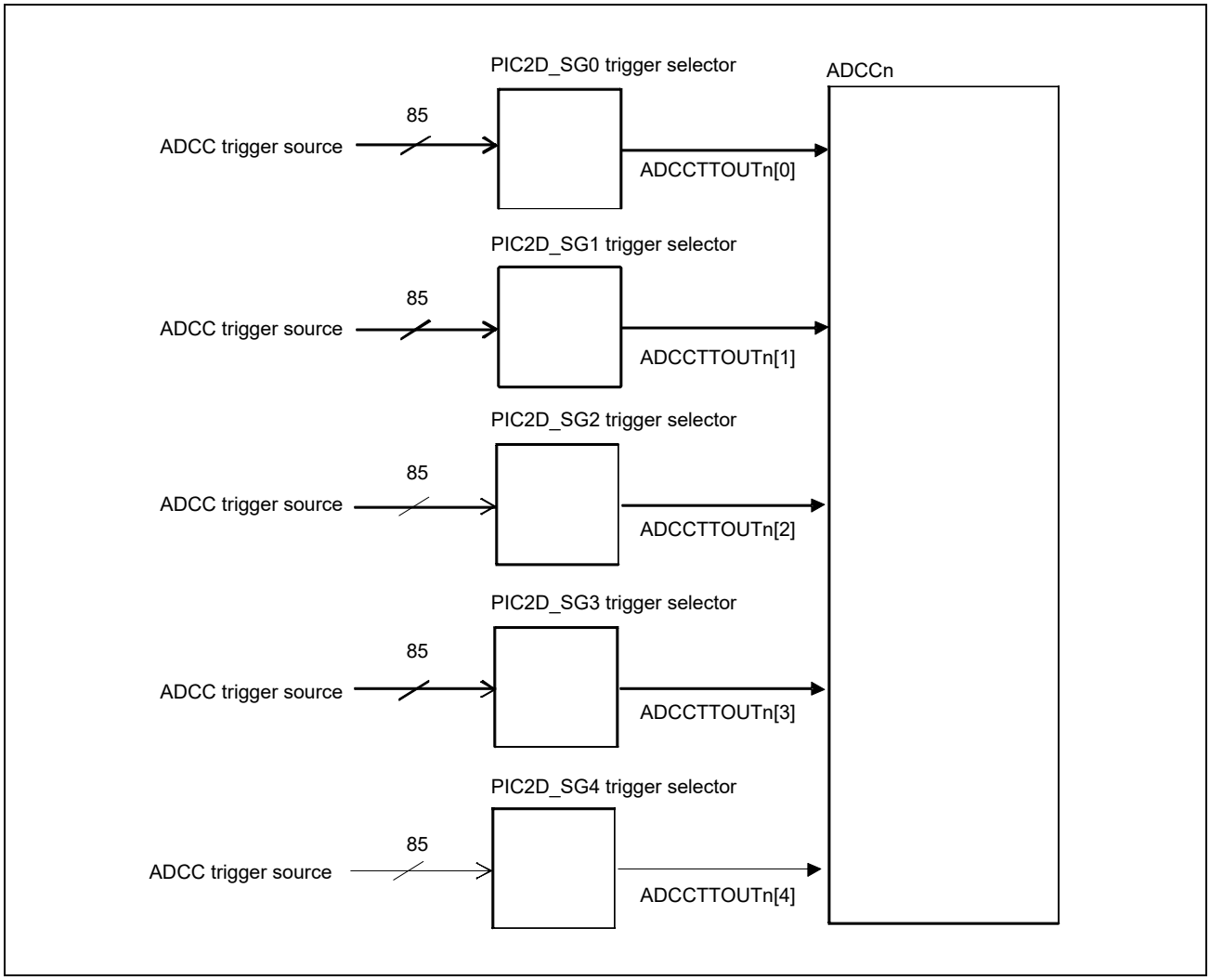


Figure 24.77 Block Diagram of ADCC Trigger Select Function

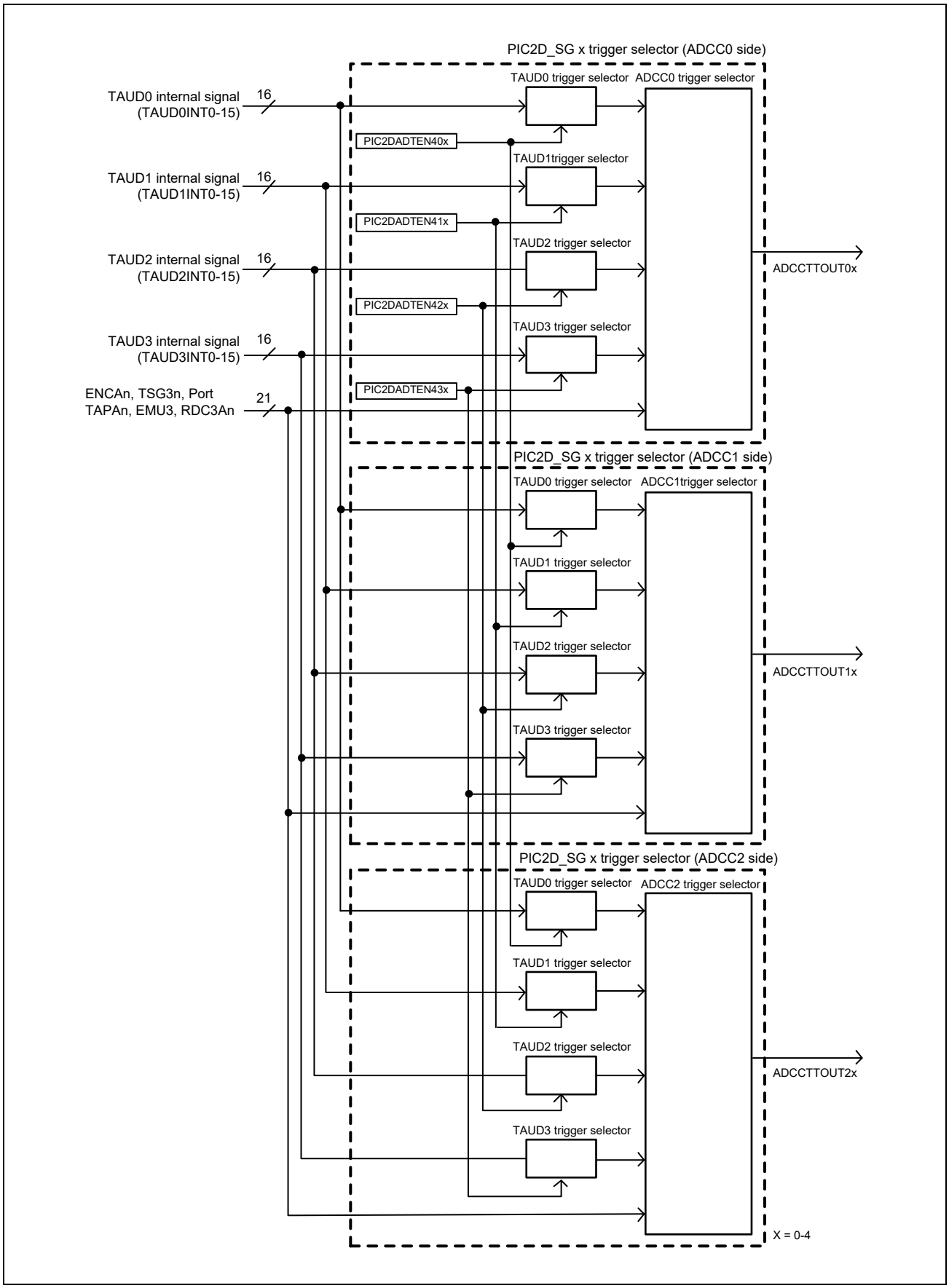
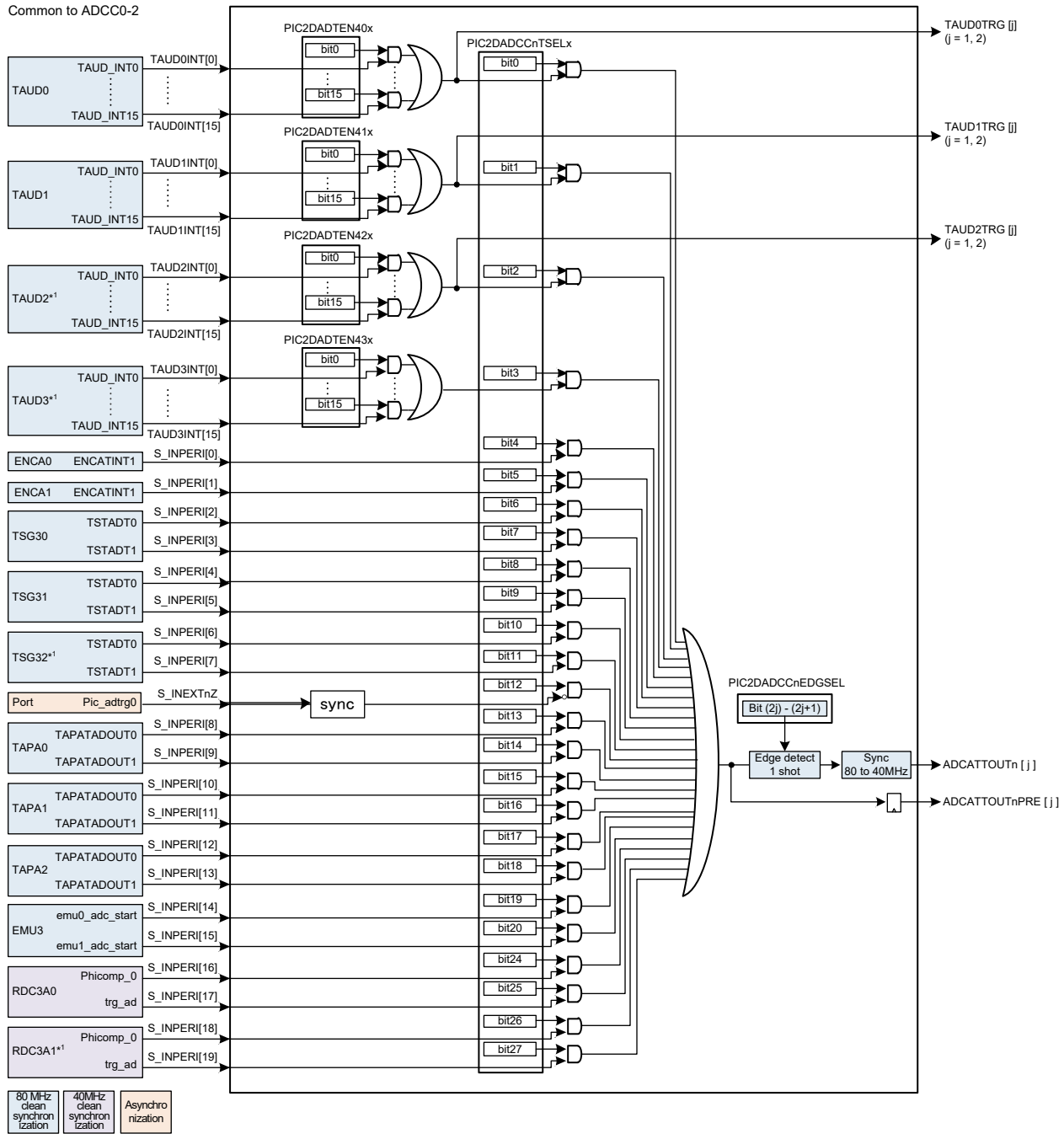


Figure 24.78 Block Diagram of PIC2D_SGx

Note 1. RDC3A1, TSG32, TAUD2, TAUD3, and TAUD2/TAUD3 related signals are not provided for the C1M-A1.

(3) Registers

For the settings of the registers used in this function, see **Figure 24.79, Block Diagram of PIC2D**, and **Sections from 24.3.2.2 to 24.3.2.4**.



Note 1. This is not provided for the C1M-A1.

Note 2. For external trigger signal, a delay of maximum two cycles of the clock (12.5 ns (80 MHz) @1clk) is generated because of the synchronization circuit (Sync) which needs to be considered when selecting an external trigger signal as the ADCC trigger.

Figure 24.79 Block Diagram of PIC2D

(4) Function

AD trigger signal can be selected for individual ADCCn channel groups and output can be selected from rising edge, falling edge, and both edges. TAUD trigger for the scan group with the same number is shared among ADCC0, ADCC1, and ADCC2.

(5) Flow chart

Set this function before starting A/D converter.

24.3.3.2 TAUD Trigger Output Function

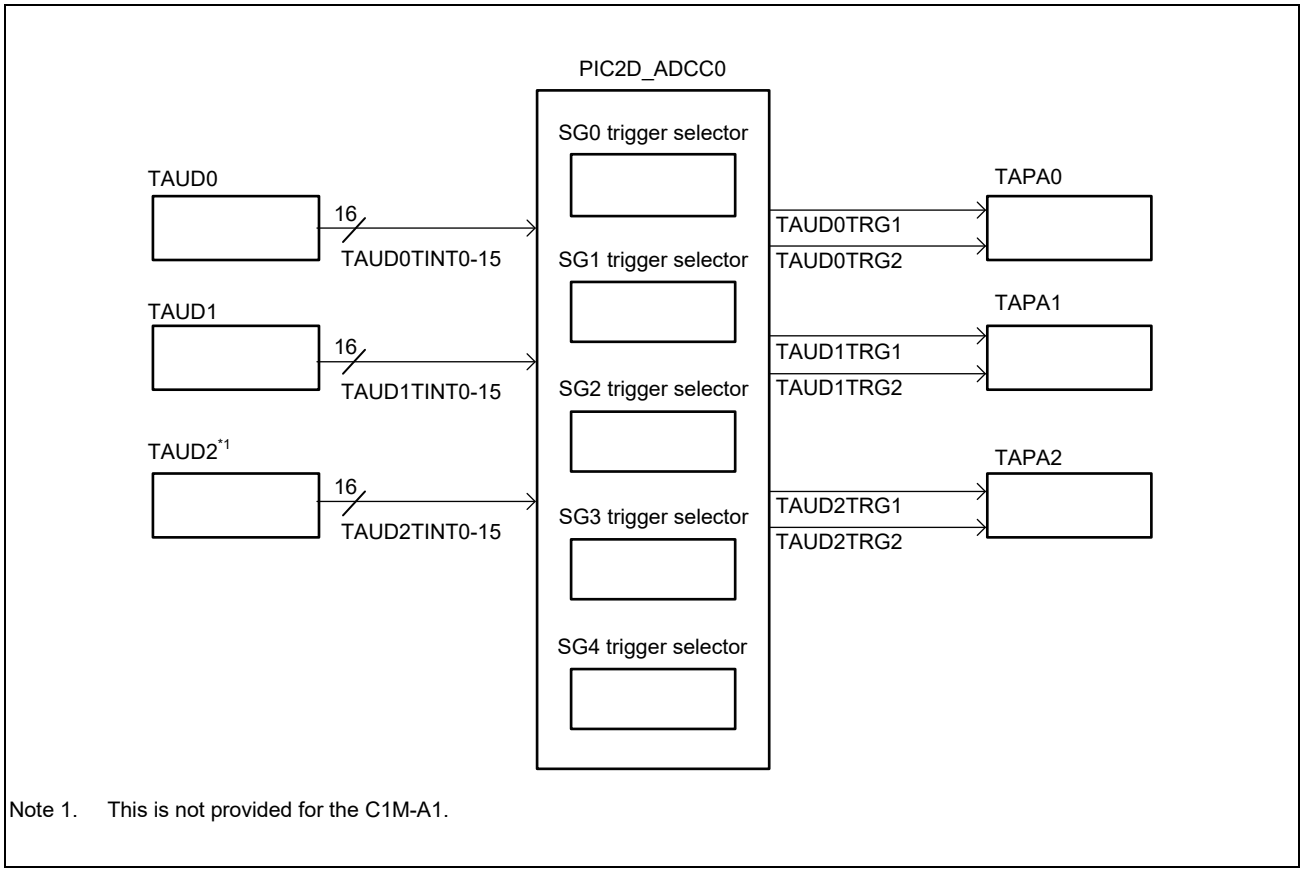
(1) Overview

The interrupt request signal of each channel of TAUD is masked and OR'ed, and output to TAPA as the TAUD trigger signal. This function is available only for the scan groups 1 and 2.

The TAUD trigger signal is used as the trigger source for A/D converter conversion trigger signal for TAPA. For the detail, see **Section 21.4.3, Selecting a Trigger to Start Conversion by the A/D Converter**.

(2) Configuration

The TAUD trigger output function is realized by PIC2D. The following figure shows the block diagram of the TAUD trigger output function.



Note 1. This is not provided for the C1M-A1.

Figure 24.80 Block Diagram of TAUD Trigger Output Function

(3) Registers

For the settings of the registers used in this function, see **Figure 24.79, Block Diagram of PIC2D**, and **Section 24.3.2.4, PIC2DADTEN4nx — A/D Converter Trigger Output Select Control Register**.

The values of PIC2D registers used in this function are as follows.

PIC2DADTEN40x = (set any value)

PIC2DADTEN41x = (set any value)

PIC2DADTEN42x = (set any value)

PIC2DADTEN43x = 0000 0000 0000 0000_B

PIC2DADCCnTSELx[3:0] = 0000_B

x = 1, 2

(4) Function

The interrupt request signal of each channel of TAUD is masked and OR'ed, and output to TAPA as the TAUD trigger signal.

CAUTION

This function is available only for the scan groups 1 and 2.

When using this function with the scan groups 1 and 2, select the TAUD trigger via TAPA. Do not select the TAUD trigger directly by the ADCC trigger select function.

(5) Flow chart

Set this function before starting A/D converter.

Section 25 Enhanced Motor Control Unit 3 (EMU3)

25.1 Features of the RH850/C1M-A EMU3

The enhanced motor control unit 3 (EMU3) is a set of motor control accelerator engines that calculate the 3-phase PWM compare value using the vector control algorithm and generate rectangle wave patterns based on the current value measured by an A/D converter and the motor's angle value obtained through an R/D converter. The calculation results of the EMU3 are used by the TSG3, a 3-phase motor timer, to output PWM and rectangle waves.

The EMU3 is provided with a SubCPU in addition to the high-speed arithmetic hardware, so that it can exercise flexible motor control in combination with software interventions.

25.1.1 Number of Units

This product has the number of EMU3 units listed below.

Table 25.1 Number of EMU3 Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	1	1
Name	EMU3	EMU3

25.1.1.1 Number of EMU3 Subunits

The EMU3 has the number of SubCPU units listed in **Table 25.2**.

Table 25.2 Number of SubCPU Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	1	1
Name	SubCPU	SubCPU

The EMU3 has the number of H/W accelerator units listed in **Table 25.3**.

Table 25.3 Number of H/W Accelerator Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	2	1
Name	EMU3n (n = 0, 1)	EMU3n (n = 0)

Table 25.4 Indices

Index	Meaning
n	In this section, the number of motor control H/W accelerator units is represented by “n” (n = 0 or 1) *1. For example, the EMU3n protect register is represented as EMU3nPRT.
m, k	In this section, variables appearing in the description column are represented by “m” and “k”.

Note 1. The H/W accelerator (EMU31) is not provided for the C1M-A1.

25.1.2 Register Base Addresses

EMU3 base addresses are listed in the following table.

EMU3 register addresses are given as offsets from the base addresses in general.

Table 25.5 Register Base Addresses

Base Address Name	Base Address of CPU1 and CPU2	Base Address of SubCPU
<EMU_base>	FF70 0000$_H$	FC00 0000$_H$
<EMU3n_base> (n = 0, 1)	FF70 n000$_H$	FC00 n000$_H$

25.1.3 Clock Supply

The clocks that are supplied to the EMU3 are listed below.

Table 25.6 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
EMU3	CLK_CPU	CLK_CPU (CPU clock)
	CLK_EMU_H	CLK_EMU_H (SubCPU clock)
	CLK_EMU_L	CLK_EMU_L (EMU3 motor control H/W accelerator clock)
	CLK_HSB	CLK_HSB (High-speed peripheral clock)
	CLKC_HSB	CLKC_HSB (Unmodulated high-speed peripheral clock)

25.1.4 Interrupt Requests

Table 25.7 gives a list of EMU3 interrupts.

For details of the interrupt sources, see **Section 25.4.15, Interrupt Control**.

Table 25.7 List of EMU3 Interrupts

Unit Interrupt Signal	Interrupt Number	DMA Trigger Number		DTS Trigger Number	
		1st	2nd	1st	2nd
EMU30					
EMU30 interrupt 0	56	77	—	77	—
EMU30 interrupt 1	57	78	—	78	—
EMU30 interrupt 2	58	79	—	79	—
EMU30 interrupt 3	59	80	—	80	—
EMU30 interrupt 4	60	81	—	81	—
EMU30 interrupt 5	61	—	35	—	35
EMU30 interrupt 6	62	—	36	—	36
EMU30 interrupt 7	63	—	37	—	37
EMU31					
EMU31 interrupt 0*1	66	82	—	82	—
EMU31 interrupt 1*1	67	83	—	83	—
EMU31 interrupt 2*1	68	84	—	84	—
EMU31 interrupt 3*1	69	85	—	85	—
EMU31 interrupt 4*1	70	86	—	86	—
EMU31 interrupt 5*1	71	—	40	—	40
EMU31 interrupt 6*1	72	—	41	—	41
EMU31 interrupt 7*1	73	—	42	—	42

Note 1. EMU31 interrupts 0 to 7 are only supported for the RH850/C1M-A2.

25.1.5 Reset Sources

The EMU3 reset sources are listed in the following table.

Table 25.8 Reset Sources

Unit Name	Reset Source
EMU3	Any reset source

25.2 Overview

25.2.1 Functional Overview

The EMU3 is furnished with two channels of motor control accelerator engines (motor control H/W accelerators)*[1] that calculate the 3-phase PWM compare value using the vector control algorithm and generate rectangle wave patterns based on the current value measured by an A/D converter and the motor's angle value obtained through an R/D converter and with a SubCPU that is used for control of the motor control H/W accelerators and for flexible motor control with software interventions.

Note 1. The H/W accelerator (EMU31) of channel 1 is not provided for the C1M-A1.

Figure 25.1 shows the configuration of the EMU3.

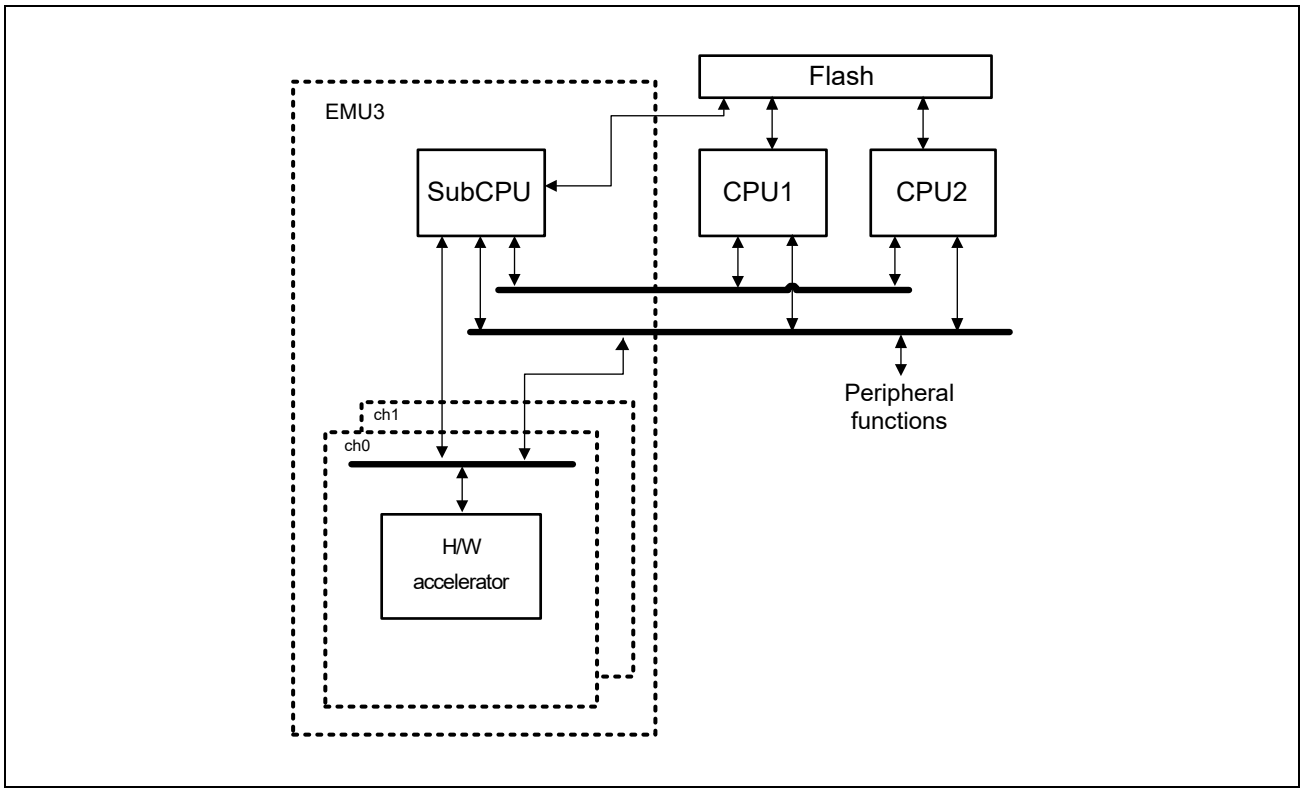


Figure 25.1 EMU3 Configuration Schematic

25.2.1.1 SubCPU

The EMU3 incorporates an RH850 G3MH core as the SubCPU that enables the control of the H/W accelerators and the modification/addition of motor control algorithms. The SubCPU has a dedicated bus to the H/W accelerators so that registers can be accessed at high speeds. See **Section 3, CPU System**, for the specifications for the RH850 G3MH core.

For the address map of the SubCPU, see **Section 4, Address Space**.

25.2.1.2 H/W Accelerators

A H/W accelerator is provided with the vector arithmetic control function that is used primarily to calculate the PWM duty value and the rectangle wave generation function. The accelerator can execute these PWM control and rectangle wave control functions simultaneously and can generate both the PWM compare value and rectangle wave pattern simultaneously.

Since the H/W accelerator can generate an interrupt on completion of processing in each IP and on a compare match and so on, it provides flexibility such that parts of arithmetic operations can be performed on a CPU.

Table 25.9 shows the outline of the H/W accelerator specifications.

Table 25.9 Major H/W Accelerator Specifications

Item	Function	Description
Inputs		Get various types of sensor information and triggers from A/D and R/D converters and so on.
	U/V/W phase current value inputs from an A/D converter	Get the results of A/D conversion on the U/V/W phase current values
	Resolver angle inputs from an R/D converter	Get the R/D conversion results and Z phase of the resolver.
	Carrier peak and trough trigger inputs from TSG3	Gets the peak and trough trigger signals from the carrier counter.
Outputs		Output rectangle wave pattern and PWM control signals to TSG3.
	Rectangle wave pattern outputs	1 bits×3 lines (U-phase pattern, V-phase pattern, W-phase pattern)
	PWM control value outputs	18 bits×4 lines (carrier period value, U-phase PWM compare value, V-phase PWM compare value, W-phase PWM compare value) 1 bit×1 line (Write Enable signal)
Register interface		Permits the SubCPU, CPU1, and CPU2 to manipulate the registers in the motor control H/W accelerators.
Interrupt outputs		8 output lines, selectable from 25 types of interrupt sources.
Motor control functions		The motor control functions are implemented in blocks as shown below.
	Input arithmetic function (input IP)	Performs dq conversion using the motor current value and electrical angle.
	PI control arithmetic function (PI control IP)	Exercises PI control using the current values of the dq-axes.
	PWM value arithmetic function (PWM IP)	Calculate the PWM duty value with 3-phase conversion using the voltage values of the dq-axes and electrical angles.
	Rectangle wave pattern arithmetic function (Rectangle IP)	Calculates the rectangle wave output level and compare value from the voltage phase value.
	Angle generation function (Angle generation IP)	Calculates the electrical angle from the resolver angle and detects resolver angle compare match and electrical angle compare match.
	Batch rectangle wave control function (batch rectangle IP)	Switches the U/V/W-phase rectangle wave output patterns all at once upon detection of an electrical angle compare match.
	Independent rectangle wave control function 1 (independent rectangle IP1)	Sets up three types each of compare values and rectangle wave output pattern values for each of the U/V/W phases and switches the rectangle wave output patterns at an independent timing of the phases.
	Independent rectangle wave control function 2 (independent rectangle IP2)	Generates rectangle waves of high degree of freedom with reinforcing the register compare function.
	Pulse period measurement timer / Resolver angle measurement timer	The timer that counts the number of cycles per Z-phase interval. It can be used as an item of information for calculating the speed.
	Checking buffer (fault detection function)	A buffer to be used by a CPU program to check the results of H/W accelerator processing.
	IIR filter	A quadratic IIR filter to be used filter out the current value obtained from an A/D converter.

Simultaneous operation of an input IP, PI control IP, PWM IP, and Rectangle IP module is not possible. An attempt to start multiple IP modules from among those in this set at the same time leads to starting of that with the highest priority,

according to the priority order rectangle IP > input IP > PI control IP > PWM IP. An attempt to start an IP module which is identical to one that is already operating is ignored.

25.2.2 Block Diagram

The major components of the H/W accelerators are shown below*[1].

Note 1. The H/W accelerator (EMU31) of channel 1 is not provided for the C1M-A1.

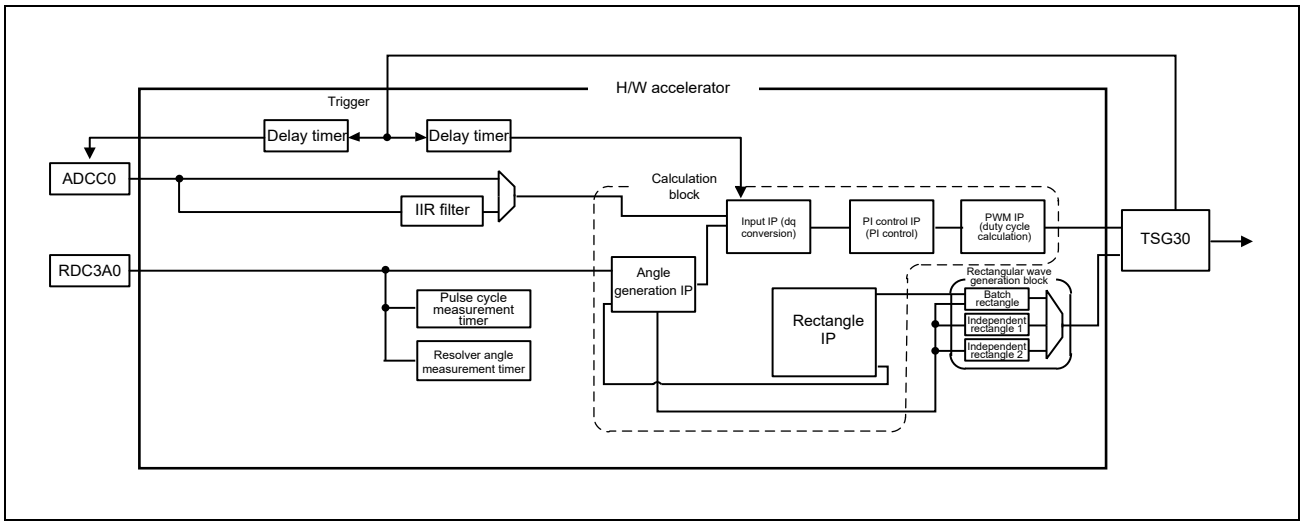
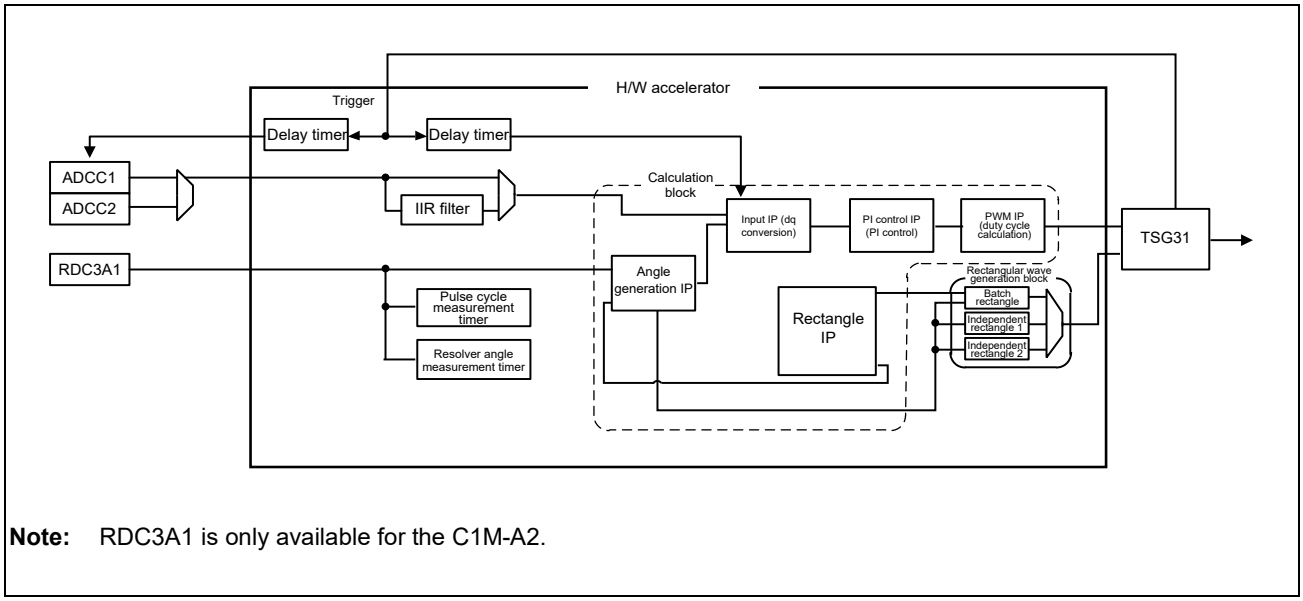


Figure 25.2 EMU30 Block Diagram



Note: RDC3A1 is only available for the C1M-A2.

Figure 25.3 EMU31 Block Diagram

The A/D converter data to be taken into the EMU31 can be selected from the ADC1 or ADC2. For details, see the register description of the ADC Select Register (EMU3ADCSEL).

25.3 Registers

25.3.1 List of Registers

The tables that follow list the registers of the EMU3. Blank entries in the Address column denote reserved areas. They must not be accessed.

Table 25.10 List of H/W Accelerator Registers (General/Common) (1/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n protect register	EMU3nPRT	<EMU3n_base> + 0000_H	8 R/W	00_H
EMU3n control register	EMU3nCTR	<EMU3n_base> + 0004_H	8 R/W	00_H
EMU3n register value reflection control register	EMU3nREFCTR	<EMU3n_base> + 0008_H	8 R/W	00_H
EMU3n IP startup trigger source select register	EMU3nIPTRG	<EMU3n_base> + 0010_H	8 R/W	00_H
EMU3n IP software startup register	EMU3nIPSFT	<EMU3n_base> + 0014_H	8 R/W	00_H
EMU3n A/D conversion completion timing select register	EMU3nADEND	<EMU3n_base> + 0016_H	8 R/W	00_H
EMU3n A/D conversion start trigger source select register	EMU3nADTRG	<EMU3n_base> + 0018_H	8/16/32 R/W	$0000\ 0000_H$
EMU3n A/D conversion start trigger source determination register	EMU3nADMON	<EMU3n_base> + $001C_H$	32 R	$0000\ 0000_H$
EMU3n A/D conversion start trigger source determination clear register	EMU3nADMONC	<EMU3n_base> + 0020_H	8/16/32 R/W	$0000\ 0000_H$
EMU3n data delay count value register	EMU3nDDCNT	<EMU3n_base> + 0024_H	32 R/W	$0000\ 0000_H$
EMU3n interrupt source select 0 register	EMU3nINT0	<EMU3n_base> + 0028_H	32 R/W	$0000\ 0000_H$
EMU3n interrupt source select 1 register	EMU3nINT1	<EMU3n_base> + $002C_H$	32 R/W	$0000\ 0000_H$
EMU3n interrupt source select 2 register	EMU3nINT2	<EMU3n_base> + 0030_H	32 R/W	$0000\ 0000_H$
EMU3n interrupt source select 3 register	EMU3nINT3	<EMU3n_base> + 0034_H	32 R/W	$0000\ 0000_H$
EMU3n interrupt source select 4 register	EMU3nINT4	<EMU3n_base> + 0038_H	32 R/W	$0000\ 0000_H$
EMU3n interrupt source select 5 register	EMU3nINT5	<EMU3n_base> + $003C_H$	32 R/W	$0000\ 0000_H$
EMU3n interrupt source select 6 register	EMU3nINT6	<EMU3n_base> + 0040_H	32 R/W	$0000\ 0000_H$
EMU3n interrupt source select 7 register	EMU3nINT7	<EMU3n_base> + 0044_H	32 R/W	$0000\ 0000_H$
EMU3n interrupt source determination register	EMU3nINTSD	<EMU3n_base> + 0048_H	32 R	$0000\ 0000_H$
EMU3n interrupt source determination clear register	EMU3nINTSDC	<EMU3n_base> + $004C_H$	32 R/W	$0000\ 0000_H$
EMU3n overflow detection result register	EMU3nOFMON	<EMU3n_base> + 0050_H	8 R	00_H
EMU3n zero division detection result register	EMU3nZDMON	<EMU3n_base> + 0051_H	8 R	00_H
EMU3n overflow detection result clear register	EMU3nOFMONC	<EMU3n_base> + 0052_H	8 R/W	00_H
EMU3n zero division detection result clear register	EMU3nZDMONC	<EMU3n_base> + 0053_H	8 R/W	00_H
EMU3n pulse period measurement timer control register	EMU3nPMTCTR	<EMU3n_base> + 0060_H	16 R/W	0000_H
EMU3n pulse period measurement timer counter register	EMU3nPMTCNT	<EMU3n_base> + 0064_H	32 R/W	$0000\ 0000_H$
EMU3n pulse period measurement timer capture register	EMU3nPMTCAP	<EMU3n_base> + 0068_H	32 R	$0000\ 0000_H$
EMU3n pulse period measurement timer overflow register	EMU3nPMTOF	<EMU3n_base> + $006C_H$	8 R/W	00_H

Table 25.10 List of H/W Accelerator Registers (General/Common) (2/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n resolver angle measurement timer control register	EMU3nPMT2CTR	<EMU3n_base> + 0070_H	8 R/W	00_H
EMU3n resolver angle measurement timer soft trigger register	EMU3nPMT2SFT	<EMU3n_base> + 0074_H	8 R/W	00_H
EMU3n resolver angle measurement timer counter register	EMU3nPMT2CNT	<EMU3n_base> + 0078_H	32 R/W	$0000\ 0000_H$
EMU3n resolver angle measurement timer capture register	EMU3nPMT2CAP	<EMU3n_base> + $007C_H$	32 R	$0000\ 0000_H$
EMU3n resolver angle measurement timer capture interval value register	EMU3nPMT2INVL	<EMU3n_base> + 0080_H	8 R/W	$0B_H$
EMU3n A/D conversion start soft trigger register	EMU3nADSFTTRG	<EMU3n_base> + 0084_H	8 R/W	00_H
EMU3n H/W arithmetic block IDLE time startup command A0 register	EMU3nFUNCIDLEGRPA0	<EMU3n_base> + 0090_H	16 R/W	0000_H
EMU3n H/W arithmetic block IDLE time startup command A1 register	EMU3nFUNCIDLEGRPA1	<EMU3n_base> + 0092_H	16 R/W	0000_H
EMU3n H/W arithmetic block IDLE time startup command A2 register	EMU3nFUNCIDLEGRPA2	<EMU3n_base> + 0094_H	16 R/W	0000_H
EMU3n H/W arithmetic block completion determination A register	EMU3nFUNCFINGRPA	<EMU3n_base> + 0096_H	16 R	0000_H
EMU3n H/W arithmetic block IDLE time startup command B register	EMU3nFUNCIDLEGRPB	<EMU3n_base> + 0098_H	16 R/W	0000_H
EMU3n H/W arithmetic block completion determination B register	EMU3nFUNCFINGRPB	<EMU3n_base> + $009A_H$	16 R	0000_H
EMU3n H/W arithmetic block WAIT time startup command A register	EMU3nFUNCWAITGRPA	<EMU3n_base> + $00A4_H$	16 R/W	0000_H
EMU3n H/W arithmetic block WAIT time startup command B register	EMU3nFUNCWAITGRPB	<EMU3n_base> + $00A8_H$	16 R/W	0000_H
EMU3n functional IP state determination A register	EMU3nFSMSTGRPA	<EMU3n_base> + $00D4_H$	32 R	$0000\ 0000_H$
EMU3n functional IP state determination B register	EMU3nFSMSTGRPB	<EMU3n_base> + $00D8_H$	32 R	$0000\ 0000_H$
EMU3n H/W arithmetic block post-completion transition control A0 register	EMU3nFUNCFLGRPA0	<EMU3n_base> + $00E4_H$	16 R/W	0000_H
EMU3n H/W arithmetic block post-completion transition control A1 register	EMU3nFUNCFLGRPA1	<EMU3n_base> + $00E8_H$	8 R/W	00_H
EMU3n H/W arithmetic block post-completion transition control A2 register	EMU3nFUNCFLGRPA2	<EMU3n_base> + $00EC_H$	32 R/W	$0000\ 0000_H$
EMU3n H/W arithmetic block post-completion transition control B register	EMU3nFUNCFLGRPB	<EMU3n_base> + $00F0_H$	32 R/W	$0000\ 0000_H$

Table 25.11 List of H/W Accelerator Registers (Angle Generation IP)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n angle generation IP control register	EMU3nANGCTR	<EMU3n_base> + 0180_H	8 R/W	00_H
EMU3n compare judgment correction register 0	EMU3nCPJUD0	<EMU3n_base> + 0182_H	8 R/W	FF_H
EMU3n compare judgment correction register 1	EMU3nCPJUD1	<EMU3n_base> + 0183_H	8 R/W	FF_H
EMU3n resolver angle software input register	EMU3nRESTHSFT	<EMU3n_base> + 0184_H	16 R/W	0000_H
EMU3n resolver angle offset value register	EMU3nANGOFS	<EMU3n_base> + 0186_H	16 R/W	0000_H
EMU3n electrical angle generation coefficient register	EMU3nPXR	<EMU3n_base> + 0188_H	16 R/W	0100_H
EMU3n resolver angle register	EMU3nRESTHETA	<EMU3n_base> + $018A_H$	16 R/W	0000_H
EMU3n electrical angle register	EMU3nTHTEFIX	<EMU3n_base> + $018C_H$	16 R/W	0000_H
EMU3n resolver pole number setting register	EMU3nRESRLD	<EMU3n_base> + $018E_H$	8 R/W	00_H
EMU3n resolver angle period count value register	EMU3nRESCNT	<EMU3n_base> + $018F_H$	8 R/W	00_H
EMU3n post error convolution resolver angle register	EMU3nTHTRESFIX	<EMU3n_base> + 0190_H	16 R/W	0000_H

Table 25.12 List of H/W Accelerator Registers (Input IP) (1/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n input IP control register	EMU3nCTRINMD	<EMU3n_base> + $01C0_H$	16 R/W	0000_H
EMU3n resolver angle monitor register	EMU3nTHTRESFIXIN	<EMU3n_base> + $01C4_H$	16 R	0000_H
EMU3n electrical angle retention register	EMU3nTHTE	<EMU3n_base> + $01C6_H$	16 R	0000_H
EMU3n input IP electrical angle software input register	EMU3nTHTESFT	<EMU3n_base> + $01C8_H$	16 R/W	0000_H
EMU3n electrical angle response delay correction variable register	EMU3nEARD	<EMU3n_base> + $01CA_H$	16 R/W	0000_H
EMU3n electrical angle input buffer register	EMU3nTHTEIBUF	<EMU3n_base> + $01CC_H$	16 R/W	0000_H
EMU3n input IP post correction electrical angle register	EMU3nTHTESEL	<EMU3n_base> + $01CE_H$	16 R/W	0000_H
EMU3n A/D data 0 register	EMU3nAD0	<EMU3n_base> + $01D0_H$	16 R/W	0000_H
EMU3n A/D data 0 input buffer register	EMU3nAD0IBUF	<EMU3n_base> + $01D2_H$	16 R/W	0000_H
EMU3n A/D data 1 register	EMU3nAD1	<EMU3n_base> + $01D4_H$	16 R/W	0000_H
EMU3n A/D data 1 input buffer register	EMU3nAD1IBUF	<EMU3n_base> + $01D6_H$	16 R/W	0000_H
EMU3n A/D data 2 register	EMU3nAD2	<EMU3n_base> + $01D8_H$	16 R/W	0000_H
EMU3n A/D data 2 input buffer register	EMU3nAD2IBUF	<EMU3n_base> + $01DA_H$	16 R/W	0000_H
EMU3n A/D data 0 conversion value register	EMU3nAD0FIX	<EMU3n_base> + $01DC_H$	16 R/W	0000_H
EMU3n A/D data 0 origin correction value register	EMU3nAD0OFS	<EMU3n_base> + $01DE_H$	16 R/W	0800_H
EMU3n A/D data 1 conversion value register	EMU3nAD1FIX	<EMU3n_base> + $01E0_H$	16 R/W	0000_H
EMU3n A/D data 1 origin correction value register	EMU3nAD1OFS	<EMU3n_base> + $01E2_H$	16 R/W	0800_H

Table 25.12 List of H/W Accelerator Registers (Input IP) (2/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n A/D data 2 conversion value register	EMU3nAD2FIX	<EMU3n_base> + $01E4_H$	16 R/W	0000_H
EMU3n A/D data 2 origin correction value register	EMU3nAD2OFS	<EMU3n_base> + $01E6_H$	16 R/W	0800_H
EMU3n dq-axis current conversion coefficient register	EMU3nSR2	<EMU3n_base> + $01E8_H$	32 R/W	0000 $D106_H$
EMU3n LSB adjustment register	EMU3nDIVLSB	<EMU3n_base> + $01EC_H$	32 R/W	0001 0000_H
EMU3n U-phase current value register	EMU3nIUFIX	<EMU3n_base> + $01F0_H$	32 R/W	0000 0000_H
EMU3n V-phase current value register	EMU3nIVFIX	<EMU3n_base> + $01F4_H$	32 R/W	0000 0000_H
EMU3n W-phase current value register	EMU3nIWFIX	<EMU3n_base> + $01F8_H$	32 R/W	0000 0000_H
EMU3n d-axis current value register	EMU3nIDFIX	<EMU3n_base> + $01FC_H$	32 R/W	0000 0000_H
EMU3n q-axis current value register	EMU3nIQFIX	<EMU3n_base> + 0200_H	32 R/W	0000 0000_H
EMU3n U-phase current value output buffer register	EMU3nIUFIXOBUF	<EMU3n_base> + 0204_H	32 R	0000 0000_H
EMU3n V-phase current value output buffer register	EMU3nIVFIXOBUF	<EMU3n_base> + 0208_H	32 R	0000 0000_H
EMU3n W-phase current value output buffer register	EMU3nIWFIXOBUF	<EMU3n_base> + $020C_H$	32 R	0000 0000_H
EMU3n d-axis current value output buffer register	EMU3nIDFIXOBUF	<EMU3n_base> + 0210_H	32 R/W	0000 0000_H
EMU3n q-axis current value output buffer register	EMU3nIQFIXOBUF	<EMU3n_base> + 0214_H	32 R/W	0000 0000_H
EMU3n Kirchhoff's current law threshold value register	EMU3nKCLJUD	<EMU3n_base> + 0218_H	32 R/W	0000 0000_H
EMU3n A/D data input buffer select register	EMU3nADBUFSEL	<EMU3n_base> + $021C_H$	8 R/W	00_H
EMU3n A/D data 0 conversion value output buffer register	EMU3nAD0FIXOBUF	<EMU3n_base> + 0220_H	16 R	0000_H
EMU3n A/D data 1 conversion value output buffer register	EMU3nAD1FIXOBUF	<EMU3n_base> + 0222_H	16 R	0000_H
EMU3n A/D data 2 conversion value output buffer register	EMU3nAD2FIXOBUF	<EMU3n_base> + 0224_H	16 R	0000_H

Table 25.13 List of H/W Accelerator Registers (PI Control IP)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n PI control IP control register	EMU3nPICTR	<EMU3n_base> + 0260_H	8 R/W	00_H
EMU3n d-axis directive current value register	EMU3nIDIN	<EMU3n_base> + 0268_H	32 R/W	$0000\ 0000_H$
EMU3n q-axis directive current value register	EMU3nIQIN	<EMU3n_base> + $026C_H$	32 R/W	$0000\ 0000_H$
EMU3n d-axis current value software input register	EMU3nID	<EMU3n_base> + 0270_H	32 R/W	$0000\ 0000_H$
EMU3n q-axis current value software input register	EMU3nIQ	<EMU3n_base> + 0274_H	32 R/W	$0000\ 0000_H$
EMU3n d-axis proportional gain 0 register	EMU3nGPD0	<EMU3n_base> + 0278_H	32 R/W	$0000\ 0000_H$
EMU3n q-axis proportional gain 0 register	EMU3nGPQ0	<EMU3n_base> + $027C_H$	32 R/W	$0000\ 0000_H$
EMU3n d-axis proportional gain register	EMU3nGPD	<EMU3n_base> + 0280_H	32 R/W	$0000\ 0000_H$
EMU3n q-axis proportional gain register	EMU3nGPQ	<EMU3n_base> + 0284_H	32 R/W	$0000\ 0000_H$
EMU3n d-axis integrated gain register	EMU3nGID	<EMU3n_base> + 0288_H	32 R/W	$0000\ 0000_H$
EMU3n q-axis integrated gain register	EMU3nGIQ	<EMU3n_base> + $028C_H$	32 R/W	$0000\ 0000_H$
EMU3n d-axis integrated maximum value register	EMU3nGIDMAX	<EMU3n_base> + 0290_H	32 R/W	$0000\ 0000_H$
EMU3n q-axis integrated maximum value register	EMU3nGIQMAX	<EMU3n_base> + 0294_H	32 R/W	$0000\ 0000_H$
EMU3n d-axis integrated value software input register	EMU3nSUMID	<EMU3n_base> + 0298_H	32 R/W	$0000\ 0000_H$
EMU3n q-axis integrated value software input register	EMU3nSUMIQ	<EMU3n_base> + $029C_H$	32 R/W	$0000\ 0000_H$
EMU3n d-axis integrated value monitor register	EMU3nSUMIDM	<EMU3n_base> + $02A0_H$	32 R	$0000\ 0000_H$
EMU3n q-axis integrated value monitor register	EMU3nSUMIQM	<EMU3n_base> + $02A4_H$	32 R	$0000\ 0000_H$
EMU3n d-axis voltage maximum value register	EMU3nVDMAX	<EMU3n_base> + $02A8_H$	32 R/W	$0000\ 0000_H$
EMU3n q-axis voltage maximum value register	EMU3nVQMAX	<EMU3n_base> + $02AC_H$	32 R/W	$0000\ 0000_H$
EMU3n d-axis voltage value register	EMU3nVD	<EMU3n_base> + $02B0_H$	32 R/W	$0000\ 0000_H$
EMU3n q-axis voltage value register	EMU3nVQ	<EMU3n_base> + $02B4_H$	32 R/W	$0000\ 0000_H$
EMU3n d-axis voltage value output buffer register	EMU3nVDOBUF	<EMU3n_base> + $02B8_H$	32 R	$0000\ 0000_H$
EMU3n q-axis voltage value output buffer register	EMU3nVQOBUF	<EMU3n_base> + $02BC_H$	32 R	$0000\ 0000_H$

Table 25.14 List of H/W Accelerator Registers (PWM IP) (1/3)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n PWM IP control register	EMU3nPWMCTR	<EMU3n_base> + 0300_H	32 R/W	$0001\ 0000_H$
EMU3n PWM data software transfer register	EMU3nPWMDT	<EMU3n_base> + 0304_H	8 R/W	00_H
EMU3n d-axis voltage correction value register	EMU3nVDCRCT	<EMU3n_base> + 0308_H	32 R/W	$0000\ 0000_H$
EMU3n q-axis voltage correction value register	EMU3nVQCRCT	<EMU3n_base> + $030C_H$	32 R/W	$0000\ 0000_H$
EMU3n angular velocity value register	EMU3nVEL	<EMU3n_base> + 0310_H	32 R	$0000\ 0000_H$
EMU3n angular velocity value software input register	EMU3nVELSFT	<EMU3n_base> + 0314_H	32 R/W	$0000\ 0000_H$
EMU3n non-interference control coefficient angular velocity value gain register	EMU3nDECVELG	<EMU3n_base> + 0318_H	32 R/W	$0000\ 0000_H$
EMU3n non-interference control coefficient magnetic flux value register	EMU3nDECFLUX	<EMU3n_base> + $031C_H$	32 R/W	$0000\ 0000_H$
EMU3n non-interference control coefficient Ld value register	EMU3nDECLD	<EMU3n_base> + 0320_H	32 R/W	$0000\ 0000_H$
EMU3n non-interference control coefficient Lq value register	EMU3nDECLQ	<EMU3n_base> + 0324_H	32 R/W	$0000\ 0000_H$
EMU3n non-interference control d-axis maximum value register	EMU3nVD2MAX	<EMU3n_base> + 0328_H	32 R/W	$0000\ 0000_H$
EMU3n non-interference control q-axis maximum value register	EMU3nVQ2MAX	<EMU3n_base> + $032C_H$	32 R/W	$0000\ 0000_H$
EMU3n post correction d-axis voltage value register	EMU3nVD2	<EMU3n_base> + 0330_H	32 R/W	$0000\ 0000_H$
EMU3n post correction q-axis voltage value register	EMU3nVQ2	<EMU3n_base> + 0334_H	32 R/W	$0000\ 0000_H$
EMU3n PWM IP electrical angle offset register	EMU3nPHI	<EMU3n_base> + 0338_H	16 R/W	0000_H
EMU3n PWM IP electrical angle adjustment coefficient register	EMU3nGTHT	<EMU3n_base> + $033C_H$	16 R/W	0100_H
EMU3n PWM IP electrical angle soft input register	EMU3nTHTFORESFT	<EMU3n_base> + $033E_H$	16 R/W	0000_H
EMU3n PWM IP post correction electrical angle register	EMU3nTHTEPWM	<EMU3n_base> + 0340_H	16 R/W	0000_H
EMU3n dq-axis voltage phase angle software input register	EMU3nTHTVSFT	<EMU3n_base> + 0342_H	16 R/W	0000_H
EMU3n dq-axis voltage value software input register	EMU3nVDQSFT	<EMU3n_base> + 0344_H	32 R/W	$0000\ 0000_H$
EMU3n 3-phase voltage conversion coefficient register	EMU3nSR23	<EMU3n_base> + 0348_H	32 R/W	$0000\ D106_H$
EMU3n post 3-phase voltage conversion U-phase voltage value register	EMU3nVU	<EMU3n_base> + $034C_H$	32 R/W	$0000\ 0000_H$
EMU3n post 3-phase voltage conversion V-phase voltage value register	EMU3nVV	<EMU3n_base> + 0350_H	32 R/W	$0000\ 0000_H$
EMU3n post 3-phase voltage conversion W-phase voltage value register	EMU3nVW	<EMU3n_base> + 0354_H	32 R/W	$0000\ 0000_H$
EMU3n PWM modulation peak value register	EMU3nTMAX	<EMU3n_base> + 0358_H	32 R/W	$0000\ 0000_H$
EMU3n PWM post modulation U-phase voltage value register	EMU3nVU0	<EMU3n_base> + $035C_H$	32 R/W	$0000\ 0000_H$

Table 25.14 List of H/W Accelerator Registers (PWM IP) (2/3)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n PWM post modulation V-phase voltage value register	EMU3nVV0	<EMU3n_base> + 0360_H	32 R/W	$0000\ 0000_H$
EMU3n PWM post modulation W-phase voltage value register	EMU3nVW0	<EMU3n_base> + 0364_H	32 R/W	$0000\ 0000_H$
EMU3n digit position alignment 1 register	EMU3nPWMK1	<EMU3n_base> + 0368_H	32 R/W	$0000\ 0000_H$
EMU3n input voltage register	EMU3n VOLV	<EMU3n_base> + $036C_H$	16 R/W	0000_H
EMU3n post duty factor calculation U-phase voltage value register	EMU3nVU1	<EMU3n_base> + 0370_H	32 R/W	$0000\ 0000_H$
EMU3n post duty factor calculation V-phase voltage value register	EMU3nVV1	<EMU3n_base> + 0374_H	32 R/W	$0000\ 0000_H$
EMU3n post duty factor calculation W-phase voltage value register	EMU3nVW1	<EMU3n_base> + 0378_H	32 R/W	$0000\ 0000_H$
EMU3n U-phase voltage correction value register	EMU3nVUOFS	<EMU3n_base> + $037C_H$	16 R/W	0000_H
EMU3n V-phase voltage correction value register	EMU3nVVOFS	<EMU3n_base> + $037E_H$	16 R/W	0000_H
EMU3n W-phase voltage correction value register	EMU3nVWOFS	<EMU3n_base> + 0380_H	16 R/W	0000_H
EMU3n post offset addition U-phase voltage value register	EMU3nVU2	<EMU3n_base> + 0384_H	32 R/W	$0000\ 0000_H$
EMU3n post offset addition V-phase voltage value register	EMU3nVV2	<EMU3n_base> + 0388_H	32 R/W	$0000\ 0000_H$
EMU3n post offset addition W-phase voltage value register	EMU3nVW2	<EMU3n_base> + $038C_H$	32 R/W	$0000\ 0000_H$
EMU3n duty factor upper-limit value register	EMU3nDTUL	<EMU3n_base> + 0390_H	32 R/W	$0000\ 0000_H$
EMU3n duty factor lower-limit value register	EMU3nDTLL	<EMU3n_base> + 0394_H	32 R/W	$0000\ 0000_H$
EMU3n post limit processing U-phase voltage value register	EMU3nVUFIX	<EMU3n_base> + 0398_H	32 R/W	$0000\ 0000_H$
EMU3n post limit processing V-phase voltage value register	EMU3nVVFIX	<EMU3n_base> + $039C_H$	32 R/W	$0000\ 0000_H$
EMU3n post limit processing W-phase voltage value register	EMU3nVWFIX	<EMU3n_base> + $03A0_H$	32 R/W	$0000\ 0000_H$
EMU3n digit position alignment 2 register	EMU3nPWMK2	<EMU3n_base> + $03A4_H$	16 R/W	0000_H
EMU3n dead time setting register	EMU3nDTT	<EMU3n_base> + $03A6_H$	16 R/W	$0FFF_H$
EMU3n carrier period register	EMU3nCARR	<EMU3n_base> + $03A8_H$	16 R/W	$7FFF_H$
EMU3n carrier period buffer register	EMU3nCARRBUF	<EMU3n_base> + $03AA_H$	16 R/W	$FFFF_H$
EMU3n U-phase PWM value register	EMU3nPWMU0	<EMU3n_base> + $03AC_H$	32 R/W	$0000\ 0000_H$
EMU3n V-phase PWM value register	EMU3nPWMV0	<EMU3n_base> + $03B0_H$	32 R/W	$0000\ 0000_H$
EMU3n W-phase PWM value register	EMU3nPWMW0	<EMU3n_base> + $03B4_H$	32 R/W	$0000\ 0000_H$
EMU3n dead time compensation threshold value register	EMU3nDTOTH	<EMU3n_base> + $03B8_H$	32 R/W	$7FFF\ FFFF_H$
EMU3n dead time compensation addend for positive current register	EMU3nDTOPV	<EMU3n_base> + $03BC_H$	16 R/W	0000_H
EMU3n dead time compensation addend for negative current register	EMU3nDTONV	<EMU3n_base> + $03BE_H$	16 R/W	0000_H
EMU3n post dead time compensation U-phase PWM value register	EMU3nPWMUDT	<EMU3n_base> + $03C0_H$	32 R/W	$0000\ 0000_H$

Table 25.14 List of H/W Accelerator Registers (PWM IP) (3/3)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n post dead time compensation V-phase PWM value register	EMU3nPWMVDT	<EMU3n_base> + $03C4_H$	32 R/W	$0000\ 0000_H$
EMU3n post dead time compensation W-phase PWM value register	EMU3nPWMWDT	<EMU3n_base> + $03C8_H$	32 R/W	$0000\ 0000_H$
EMU3n PWM upper-limit value register	EMU3nPWMUL	<EMU3n_base> + $03CC_H$	16 R/W	0000_H
EMU3n PWM lower-limit value register	EMU3nPWMLL	<EMU3n_base> + $03CE_H$	16 R/W	0000_H
EMU3n U-phase PWM compare value register	EMU3nPWMUIP	<EMU3n_base> + $03D0_H$	16 R/W	0000_H
EMU3n V-phase PWM compare value register	EMU3nPWMVIP	<EMU3n_base> + $03D2_H$	16 R/W	0000_H
EMU3n W-phase PWM compare value register	EMU3nPWMWIP	<EMU3n_base> + $03D4_H$	16 R/W	0000_H
EMU3n U-phase PWM compare value software input register	EMU3nPWMU	<EMU3n_base> + $03D8_H$	16 R/W	$47FF_H$
EMU3n V-phase PWM compare value software input register	EMU3nPWMV	<EMU3n_base> + $03DA_H$	16 R/W	$47FF_H$
EMU3n W-phase PWM compare value software input register	EMU3nPWMW	<EMU3n_base> + $03DC_H$	16 R/W	$47FF_H$

Table 25.15 List of H/W Accelerator Registers (Rectangle IP)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n rectangle IP control register	EMU3nRECCTR	<EMU3n_base> + 0480_H	8 R/W	04_H
EMU3n rectangle output software control pattern register	EMU3nPTNN	<EMU3n_base> + 0484_H	8 R/W	00_H
EMU3n rectangle output pattern AB register	EMU3nPTNAB	<EMU3n_base> + 0485_H	8 R/W	00_H
EMU3n rectangle output pattern CD register	EMU3nPTNCD	<EMU3n_base> + 0486_H	8 R/W	00_H
EMU3n rectangle output pattern EF register	EMU3nPTNEF	<EMU3n_base> + 0487_H	8 R/W	00_H
EMU3n angle compare 0 comparison value software input register	EMU3nCMP0	<EMU3n_base> + 0488_H	16 R/W	0000_H
EMU3n angle compare 1 comparison value software input register	EMU3nCMP1	<EMU3n_base> + $048A_H$	16 R/W	0000_H
EMU3n q-axis reference voltage phase software input register	EMU3nPHQSFT	<EMU3n_base> + $048C_H$	16 R/W	0000_H
EMU3n switching instruction software input register	EMU3nPSWSFT	<EMU3n_base> + $048E_H$	8 R/W	00_H
EMU3n switching instruction register	EMU3nPSW	<EMU3n_base> + $048F_H$	8 R	00_H
EMU3n angle compare 0 comparison value IP output register	EMU3nIPCMP0	<EMU3n_base> + 0490_H	16 R	0000_H

Table 25.16 List of H/W Accelerator Registers (Independent Rectangle IP1, Rectangle Wave Generation Block)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n independent rectangle IP 1 control register	EMU3nIRECCTR	<EMU3n_base> + $04C0_H$	8 R/W	00_H
EMU3n independent rectangle IP 1 output pattern update register	EMU3nIRPTN	<EMU3n_base> + $04C4_H$	8 R/W	00_H
EMU3n independent rectangle IP 1 flag select signal initialization register	EMU3nIRCTRST	<EMU3n_base> + $04C6_H$	8 R/W	00_H
EMU3n independent rectangle IP 1 U-phase angle compare 0 match detection comparison value/pattern setting 0 register	EMU3nIRUCPPN0	<EMU3n_base> + $04C8_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 U-phase angle compare 0 match detection comparison value/pattern setting 1 register	EMU3nIRUCPPN1	<EMU3n_base> + $04CC_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 U-phase angle compare 0 match detection comparison value/pattern setting 2 register	EMU3nIRUCPPN2	<EMU3n_base> + $04D0_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 V-phase angle compare 0 match detection comparison value/pattern setting 0 register	EMU3nIRVCPPN0	<EMU3n_base> + $04D4_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 V-phase angle compare 0 match detection comparison value/pattern setting 1 register	EMU3nIRVCPPN1	<EMU3n_base> + $04D8_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 V-phase angle compare 0 match detection comparison value/pattern setting 2 register	EMU3nIRVCPPN2	<EMU3n_base> + $04DC_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 W-phase angle compare 0 match detection comparison value/pattern setting 0 register	EMU3nIRWCPPN0	<EMU3n_base> + $04E0_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 W-phase angle compare 0 match detection comparison value/pattern setting 1 register	EMU3nIRWCPPN1	<EMU3n_base> + $04E4_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 W-phase angle compare 0 match detection comparison value/pattern setting 2 register	EMU3nIRWCPPN2	<EMU3n_base> + $04E8_H$	16/32 R/W	$0000\ 0000_H$
EMU3n independent rectangle IP 1 flag monitor register	EMU3nIRFLGM	<EMU3n_base> + $04EC_H$	16 R	0000_H
EMU3n independent rectangle IP 1 select signal monitor register	EMU3nIRSELM	<EMU3n_base> + $04EE_H$	16 R	0049_H

Table 25.17 List of H/W Accelerator Registers (Independent Rectangle IP2) (1/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n independent rectangle IP2 control register	EMU3nNRECCTR	<EMU3n_base> + 0500_H	8 R/W	00_H
EMU3n independent rectangle IP2 3-phase common angle correction value register	EMU3nNRECOFSALL	<EMU3n_base> + 0508_H	16 R/W	0000_H
EMU3n independent rectangle IP2 U-phase angle correction value register	EMU3nNRECOFSU	<EMU3n_base> + $050A_H$	16 R/W	0000_H
EMU3n independent rectangle IP2 V-phase angle correction value register	EMU3nNRECOFSV	<EMU3n_base> + $050C_H$	16 R/W	0000_H
EMU3n independent rectangle IP2 W-phase angle correction value register	EMU3nNRECOFSW	<EMU3n_base> + $050E_H$	16 R/W	0000_H
EMU3n independent rectangle IP2 U-phase compare control 0 register	EMU3nNRECU0	<EMU3n_base> + 0510_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 U-phase compare control 1 register	EMU3nNRECU1	<EMU3n_base> + 0514_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 U-phase compare control 2 register	EMU3nNRECU2	<EMU3n_base> + 0518_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 U-phase compare control 3 register	EMU3nNRECU3	<EMU3n_base> + $051C_H$	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 U-phase compare control 4 register	EMU3nNRECU4	<EMU3n_base> + 0520_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 U-phase compare control 5 register	EMU3nNRECU5	<EMU3n_base> + 0524_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 U-phase compare control 6 register	EMU3nNRECU6	<EMU3n_base> + 0528_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 U-phase compare control 7 register	EMU3nNRECU7	<EMU3n_base> + $052C_H$	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 V-phase compare control 0 register	EMU3nNRECV0	<EMU3n_base> + 0530_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 V-phase compare control 1 register	EMU3nNRECV1	<EMU3n_base> + 0534_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 V-phase compare control 2 register	EMU3nNRECV2	<EMU3n_base> + 0538_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 V-phase compare control 3 register	EMU3nNRECV3	<EMU3n_base> + $053C_H$	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 V-phase compare control 4 register	EMU3nNRECV4	<EMU3n_base> + 0540_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 V-phase compare control 5 register	EMU3nNRECV5	<EMU3n_base> + 0544_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 V-phase compare control 6 register	EMU3nNRECV6	<EMU3n_base> + 0548_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 V-phase compare control 7 register	EMU3nNRECV7	<EMU3n_base> + $054C_H$	8/16/32 R/W*1	$0000\ 0000_H$

Table 25.17 List of H/W Accelerator Registers (Independent Rectangle IP2) (2/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n independent rectangle IP2 W-phase compare control 0 register	EMU3nNRECW0	<EMU3n_base> + 0550_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 W-phase compare control 1 register	EMU3nNRECW1	<EMU3n_base> + 0554_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 W-phase compare control 2 register	EMU3nNRECW2	<EMU3n_base> + 0558_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 W-phase compare control 3 register	EMU3nNRECW3	<EMU3n_base> + $055C_H$	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 W-phase compare control 4 register	EMU3nNRECW4	<EMU3n_base> + 0560_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 W-phase compare control 5 register	EMU3nNRECW5	<EMU3n_base> + 0564_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 W-phase compare control 6 register	EMU3nNRECW6	<EMU3n_base> + 0568_H	8/16/32 R/W*1	$0000\ 0000_H$
EMU3n independent rectangle IP2 W-phase compare control 7 register	EMU3nNRECW7	<EMU3n_base> + $056C_H$	8/16/32 R/W*1	$0000\ 0000_H$

Note 1. The lower-order 16 bits cannot be accessed in 8-bit units.

Table 25.18 List of H/W Accelerator Registers (IIR Filter) (1/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n IIR filter channel 0 control register	EMU3nIIRCTR0	<EMU3n_base> + $05C0_H$	8 R/W	00_H
EMU3n IIR filter channel 1 control register	EMU3nIIRCTR1	<EMU3n_base> + $05C1_H$	8 R/W	00_H
EMU3n IIR filter channel 2 control register	EMU3nIIRCTR2	<EMU3n_base> + $05C2_H$	8 R/W	00_H
EMU3n IIR filter initialization register	EMU3nIIRINIT	<EMU3n_base> + $05C4_H$	8 R/W	00_H
EMU3n IIR filter software startup register	EMU3nIIRSFT	<EMU3n_base> + $05C8_H$	8 R/W	00_H
EMU3n IIR filter coefficient shift amount reload register	EMU3nIIRRLD	<EMU3n_base> + $05CC_H$	8 R/W	00_H
EMU3n IIR filter completion flag register	EMU3nIIRSTAT	<EMU3n_base> + $05CE_H$	8 R	00_H
EMU3n IIR filter completion flag clear register	EMU3nIIRSTATC	<EMU3n_base> + $05CF_H$	8 R/W	00_H
EMU3n IIR filter coefficient 0 value register	EMU3nIIRCOEFF0	<EMU3n_base> + $05D0_H$	16 R/W	0000_H
EMU3n IIR filter coefficient 1 value register	EMU3nIIRCOEFF1	<EMU3n_base> + $05D2_H$	16 R/W	0000_H
EMU3n IIR filter coefficient 2 value register	EMU3nIIRCOEFF2	<EMU3n_base> + $05D4_H$	16 R/W	0000_H
EMU3n IIR filter coefficient 3 value register	EMU3nIIRCOEFF3	<EMU3n_base> + $05D6_H$	16 R/W	0000_H
EMU3n IIR filter coefficient 4 value register	EMU3nIIRCOEFF4	<EMU3n_base> + $05D8_H$	16 R/W	0000_H
EMU3n IIR filter coefficient 5 value register	EMU3nIIRCOEFF5	<EMU3n_base> + $05DA_H$	16 R/W	0000_H
EMU3n IIR filter shift amount value register	EMU3nIIRSHIFT	<EMU3n_base> + $05DC_H$	8 R/W	00_H
EMU3n IIR filter channel 0 coefficient 0 monitor register	EMU3nIIRCOEFFM00	<EMU3n_base> + $05E0_H$	16 R	0000_H
EMU3n IIR filter channel 0 coefficient 1 monitor register	EMU3nIIRCOEFFM10	<EMU3n_base> + $05E2_H$	16 R	0000_H
EMU3n IIR filter channel 0 coefficient 2 monitor register	EMU3nIIRCOEFFM20	<EMU3n_base> + $05E4_H$	16 R	0000_H
EMU3n IIR filter channel 0 coefficient 3 monitor register	EMU3nIIRCOEFFM30	<EMU3n_base> + $05E6_H$	16 R	0000_H
EMU3n IIR filter channel 0 coefficient 4 monitor register	EMU3nIIRCOEFFM40	<EMU3n_base> + $05E8_H$	16 R	0000_H
EMU3n IIR filter channel 0 coefficient 5 monitor register	EMU3nIIRCOEFFM50	<EMU3n_base> + $05EA_H$	16 R	0000_H
EMU3n IIR filter channel 0 shift amount monitor register	EMU3nIIRSHIFTM0	<EMU3n_base> + $05EC_H$	8 R	00_H
EMU3n IIR filter channel 1 coefficient 0 monitor register	EMU3nIIRCOEFFM01	<EMU3n_base> + $05F0_H$	16 R	0000_H
EMU3n IIR filter channel 1 coefficient 1 monitor register	EMU3nIIRCOEFFM11	<EMU3n_base> + $05F2_H$	16 R	0000_H
EMU3n IIR filter channel 1 coefficient 2 monitor register	EMU3nIIRCOEFFM21	<EMU3n_base> + $05F4_H$	16 R	0000_H
EMU3n IIR filter channel 1 coefficient 3 monitor register	EMU3nIIRCOEFFM31	<EMU3n_base> + $05F6_H$	16 R	0000_H
EMU3n IIR filter channel 1 coefficient 4 monitor register	EMU3nIIRCOEFFM41	<EMU3n_base> + $05F8_H$	16 R	0000_H

Table 25.18 List of H/W Accelerator Registers (IIR Filter) (2/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n IIR filter channel 1 coefficient 5 monitor register	EMU3nIIRCOEFFM51	<EMU3n_base> + $05FA_H$	16 R	0000_H
EMU3n IIR filter channel 1 shift amount monitor register	EMU3nIIRSHIFTM1	<EMU3n_base> + $05FC_H$	8 R	00_H
EMU3n IIR filter channel 2 coefficient 0 monitor register	EMU3nIIRCOEFFM02	<EMU3n_base> + 0600_H	16 R	0000_H
EMU3n IIR filter channel 2 coefficient 1 monitor register	EMU3nIIRCOEFFM12	<EMU3n_base> + 0602_H	16 R	0000_H
EMU3n IIR filter channel 2 coefficient 2 monitor register	EMU3nIIRCOEFFM22	<EMU3n_base> + 0604_H	16 R	0000_H
EMU3n IIR filter channel 2 coefficient 3 monitor register	EMU3nIIRCOEFFM32	<EMU3n_base> + 0606_H	16 R	0000_H
EMU3n IIR filter channel 2 coefficient 4 monitor register	EMU3nIIRCOEFFM42	<EMU3n_base> + 0608_H	16 R	0000_H
EMU3n IIR filter channel 2 coefficient 5 monitor register	EMU3nIIRCOEFFM52	<EMU3n_base> + $060A_H$	16 R	0000_H
EMU3n IIR filter channel 2 shift amount monitor register	EMU3nIIRSHIFTM2	<EMU3n_base> + $060C_H$	8 R	00_H
EMU3n IIR filter channel 0 data software input register	EMU3nIIRSFTDAT0	<EMU3n_base> + 0620_H	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 1 data software input register	EMU3nIIRSFTDAT1	<EMU3n_base> + 0624_H	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 2 data software input register	EMU3nIIRSFTDAT2	<EMU3n_base> + 0628_H	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 0 delay 1 data register	EMU3nIIRZN1DAT0	<EMU3n_base> + 0630_H	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 0 delay 2 data register	EMU3nIIRZN2DAT0	<EMU3n_base> + 0634_H	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 1 delay 1 data register	EMU3nIIRZN1DAT1	<EMU3n_base> + 0638_H	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 1 delay 2 data register	EMU3nIIRZN2DAT1	<EMU3n_base> + $063C_H$	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 2 delay 1 data register	EMU3nIIRZN1DAT2	<EMU3n_base> + 0640_H	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 2 delay 2 data register	EMU3nIIRZN2DAT2	<EMU3n_base> + 0644_H	32 R/W	$0000\ 0000_H$
EMU3n IIR filter channel 0 output data register	EMU3nIIROUTDAT0	<EMU3n_base> + 0650_H	32 R	$0000\ 0000_H$
EMU3n IIR filter channel 1 output data register	EMU3nIIROUTDAT1	<EMU3n_base> + 0654_H	32 R	$0000\ 0000_H$
EMU3n IIR filter channel 2 output data register	EMU3nIIROUTDAT2	<EMU3n_base> + 0658_H	32 R	$0000\ 0000_H$

Table 25.19 List of H/W Accelerator Registers (Checking Buffer)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n checking buffer control register	EMU3nCBCTR0	<EMU3n_base> + $06C4_H$	8 R/W	00_H
EMU3n checking buffer timing select register	EMU3nCBTIM	<EMU3n_base> + $06C6_H$	16 R/W	0000_H
EMU3n A/D data 0 checking buffer register	EMU3nCBAD0	<EMU3n_base> + $06C8_H$	16 R	0000_H
EMU3n A/D data 1 checking buffer register	EMU3nCBAD1	<EMU3n_base> + $06CA_H$	16 R	0000_H
EMU3n A/D data 2 checking buffer register	EMU3nCBAD2	<EMU3n_base> + $06CC_H$	16 R	0000_H
EMU3n resolver angle checking buffer register	EMU3nCBTHTRESFIXIN	<EMU3n_base> + $06CE_H$	16 R	0000_H
EMU3n d-axis current value checking buffer register	EMU3nCBIDFIX	<EMU3n_base> + $06D0_H$	32 R	$0000\ 0000_H$
EMU3n q-axis current value checking buffer register	EMU3nCBIQFIX	<EMU3n_base> + $06D4_H$	32 R	$0000\ 0000_H$
EMU3n U-phase PWM compare value checking buffer register	EMU3nCBPWMUIP	<EMU3n_base> + $06D8_H$	32 R	$0000\ 0000_H$
EMU3n V-phase PWM compare value checking buffer register	EMU3nCBPWMVIP	<EMU3n_base> + $06DC_H$	32 R	$0000\ 0000_H$
EMU3n W-phase PWM compare value checking buffer register	EMU3nCBPWMWIP	<EMU3n_base> + $06E0_H$	32 R	$0000\ 0000_H$
EMU3n rectangle pattern value checking buffer register	EMU3nCBBREC	<EMU3n_base> + $06E4_H$	8 R	00_H
EMU3n independent rectangle IP1 pattern value checking buffer register	EMU3nCBIREC	<EMU3n_base> + $06E5_H$	8 R	00_H

Table 25.20 List of H/W Accelerator Registers (Handshaking) (1/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n data set WB0 transfer trigger register	EMU3nDATSETWB0	<EMU3n_base> + 0700_H	8 R/W	00_H
EMU3n data set WB1 transfer trigger register	EMU3nDATSETWB1	<EMU3n_base> + 0704_H	8 R/W	00_H
EMU3n data set WB2 transfer trigger register	EMU3nDATSETWB2	<EMU3n_base> + 0708_H	8 R/W	00_H
EMU3n data set BR0 transfer trigger register	EMU3nDATSETBR0	<EMU3n_base> + 0710_H	8 R/W	00_H
EMU3n data set BR1 transfer trigger register	EMU3nDATSETBR1	<EMU3n_base> + 0714_H	8 R/W	00_H
EMU3n data set BR2 transfer trigger register	EMU3nDATSETBR2	<EMU3n_base> + 0718_H	8 R/W	00_H
EMU3n data set 0 WRITE0 register	EMU3nDATSETW00	<EMU3n_base> + 0720_H	16/32 R/W*1	0000_H
EMU3n data set 0 WRITE1 register	EMU3nDATSETW01	<EMU3n_base> + 0722_H	16/32 R/W*2	0000_H
EMU3n data set 0 WRITE2 register	EMU3nDATSETW02	<EMU3n_base> + 0724_H	16/32 R/W*1	0000_H
EMU3n data set 0 WRITE3 register	EMU3nDATSETW03	<EMU3n_base> + 0726_H	16/32 R/W*2	0000_H
EMU3n data set 1 WRITE0 register	EMU3nDATSETW10	<EMU3n_base> + 0728_H	16/32 R/W*1	0000_H
EMU3n data set 1 WRITE1 register	EMU3nDATSETW11	<EMU3n_base> + $072A_H$	16/32 R/W*2	0000_H
EMU3n data set 1 WRITE2 register	EMU3nDATSETW12	<EMU3n_base> + $072C_H$	16/32 R/W*1	0000_H
EMU3n data set 1 WRITE3 register	EMU3nDATSETW13	<EMU3n_base> + $072E_H$	16/32 R/W*2	0000_H
EMU3n data set 2 WRITE0 register	EMU3nDATSETW20	<EMU3n_base> + 0730_H	16/32 R/W*1	0000_H
EMU3n data set 2 WRITE1 register	EMU3nDATSETW21	<EMU3n_base> + 0732_H	16/32 R/W*2	0000_H
EMU3n data set 2 WRITE2 register	EMU3nDATSETW22	<EMU3n_base> + 0734_H	16/32 R/W*1	0000_H
EMU3n data set 2 WRITE3 register	EMU3nDATSETW23	<EMU3n_base> + 0736_H	16/32 R/W*2	0000_H
EMU3n data set 3 WRITE0 register	EMU3nDATSETW30	<EMU3n_base> + 0738_H	16/32 R/W*1	0000_H
EMU3n data set 3 WRITE1 register	EMU3nDATSETW31	<EMU3n_base> + $073A_H$	16/32 R/W*2	0000_H
EMU3n data set 3 WRITE2 register	EMU3nDATSETW32	<EMU3n_base> + $073C_H$	16/32 R/W*1	0000_H

Table 25.20 List of H/W Accelerator Registers (Handshaking) (2/2)

Register Name	Symbol	Address	Access Width	Initial Value
EMU3n data set 3 WRITE3 register	EMU3nDATSETW33	<EMU3n_base> + 073E$_H$	16/32 R/W*2	0000$_H$
EMU3n data set 4 WRITE0 register	EMU3nDATSETW40	<EMU3n_base> + 0740$_H$	16/32 R/W*1	0000$_H$
EMU3n data set 4 WRITE1 register	EMU3nDATSETW41	<EMU3n_base> + 0742$_H$	16/32 R/W*2	0000$_H$
EMU3n data set 4 WRITE2 register	EMU3nDATSETW42	<EMU3n_base> + 0744$_H$	16/32 R/W*1	0000$_H$
EMU3n data set 4 WRITE3 register	EMU3nDATSETW43	<EMU3n_base> + 0746$_H$	16/32 R/W*2	0000$_H$
EMU3n data set 5 WRITE0 register	EMU3nDATSETW50	<EMU3n_base> + 0748$_H$	16/32 R/W*1	0000$_H$
EMU3n data set 5 WRITE1 register	EMU3nDATSETW51	<EMU3n_base> + 074A$_H$	16/32 R/W*2	0000$_H$
EMU3n data set 5 WRITE2 register	EMU3nDATSETW52	<EMU3n_base> + 074C$_H$	16/32 R/W*1	0000$_H$
EMU3n data set 5 WRITE3 register	EMU3nDATSETW53	<EMU3n_base> + 074E$_H$	16/32 R/W*2	0000$_H$
EMU3n data set 0 READ0 register	EMU3nDATSETR00	<EMU3n_base> + 0760$_H$	16/32 R *3	0000$_H$
EMU3n data set 0 READ1 register	EMU3nDATSETR01	<EMU3n_base> + 0762$_H$	16/32 R *4	0000$_H$
EMU3n data set 0 READ2 register	EMU3nDATSETR02	<EMU3n_base> + 0764$_H$	16/32 R *3	0000$_H$
EMU3n data set 0 READ3 register	EMU3nDATSETR03	<EMU3n_base> + 0766$_H$	16/32 R *4	0000$_H$
EMU3n data set 1 READ0 register	EMU3nDATSETR10	<EMU3n_base> + 0768$_H$	16/32 R *3	0000$_H$
EMU3n data set 1 READ1 register	EMU3nDATSETR11	<EMU3n_base> + 076A$_H$	16/32 R *4	0000$_H$
EMU3n data set 1 READ2 register	EMU3nDATSETR12	<EMU3n_base> + 076C$_H$	16/32 R *3	0000$_H$
EMU3n data set 1 READ3 register	EMU3nDATSETR13	<EMU3n_base> + 076E$_H$	16/32 R *4	0000$_H$
EMU3n data set 2 READ0 register	EMU3nDATSETR20	<EMU3n_base> + 0770$_H$	16/32 R*3	0000$_H$
EMU3n data set 2 READ1 register	EMU3nDATSETR21	<EMU3n_base> + 0772$_H$	16/32 R*4	0000$_H$
EMU3n data set 2 READ2 register	EMU3nDATSETR22	<EMU3n_base> + 0774$_H$	16/32 R*3	0000$_H$
EMU3n data set 2 READ3 register	EMU3nDATSETR23	<EMU3n_base> + 0776$_H$	16/32 R*4	0000$_H$
EMU3n data set 3 READ0 register	EMU3nDATSETR30	<EMU3n_base> + 0778$_H$	16/32 R*3	0000$_H$
EMU3n data set 3 READ1 register	EMU3nDATSETR31	<EMU3n_base> + 077A$_H$	16/32 R*4	0000$_H$
EMU3n data set 3 READ2 register	EMU3nDATSETR32	<EMU3n_base> + 077C$_H$	16/32 R*3	0000$_H$
EMU3n data set 3 READ3 register	EMU3nDATSETR33	<EMU3n_base> + 077E$_H$	16/32 R*4	0000$_H$
EMU3n data set 4 READ0 register	EMU3nDATSETR40	<EMU3n_base> + 0780$_H$	16/32 R*3	0000$_H$
EMU3n data set 4 READ1 register	EMU3nDATSETR41	<EMU3n_base> + 0782$_H$	16/32 R*4	0000$_H$
EMU3n data set 4 READ2 register	EMU3nDATSETR42	<EMU3n_base> + 0784$_H$	16/32 R*3	0000$_H$
EMU3n data set 4 READ3 register	EMU3nDATSETR43	<EMU3n_base> + 0786$_H$	16/32 R*4	0000$_H$
EMU3n data set 5 READ0 register	EMU3nDATSETR50	<EMU3n_base> + 0788$_H$	16/32 R*3	0000$_H$
EMU3n data set 5 READ1 register	EMU3nDATSETR51	<EMU3n_base> + 078A$_H$	16/32 R*4	0000$_H$
EMU3n data set 5 READ2 register	EMU3nDATSETR52	<EMU3n_base> + 078C$_H$	16/32 R*3	0000$_H$
EMU3n data set 5 READ3 register	EMU3nDATSETR53	<EMU3n_base> + 078E$_H$	16/32 R*4	0000$_H$

Note 1. Handles the lower-order 16 bits when using 32-bit data. See **Section 25.3.2.152, EMU3n Data Set kWRITEm Register (EMU3nDATSETWkm) (k = 0 to 5) (m = 0 to 3)** for details.

Note 2. Handles the higher-order 16 bits when using 32-bit data. See **Section 25.3.2.152, EMU3n Data Set kWRITEm Register (EMU3nDATSETWkm) (k = 0 to 5) (m = 0 to 3)** for details.

Note 3. Handles the lower-order 16 bits when using 32-bit data. See **Section 25.3.2.153, EMU3n Data Set kREADm Register (EMU3nDATSETRkm) (k = 0 to 5) (m = 0 to 3)** for details.

Note 4. Handles the higher-order 16 bits when using 32-bit data. See **Section 25.3.2.153, EMU3n Data Set kREADm Register (EMU3nDATSETRkm) (k = 0 to 5) (m = 0 to 3)** for details.

Table 25.21 List of EMUSS Registers

Register Name	Symbol	Address	Access Width	Initial Value
SubCPU startup register	EMU3CPUINIT	<EMU_base> + 0000_2000_H	8 R/W	00_H
ADC select register	EMU3ADCSEL	<EMU_base> + 0000_2040_H	8 R/W	00_H

In the register descriptions that follow, “Data (signed),” “Data bit (signed integer),” and “Data bit (signed fixed-point number)” are represented in 2's complement notation. The bit assigned to the sign part in each register description is represented by “(b**: Sign part).” For “(b15-11: Sign part),” for example, a 0 is written or read if b15-11 carry a positive number and a 1 is written or read if b15-11 carry a negative number.

25.3.2 Register Details

25.3.2.1 EMU3n Protect Register (EMU3nPRT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0000_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PRTCT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.22 EMU3nPRT Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	PRTCT	Sets or resets protection of a register with a write protection. 0: Protected state 1: Unprotect

The EMU3nPRT register sets the protect function to prevent the EMU3nCTR register from being easily rewritten.

PRTCT bit

To change the EMU3nCTR register, follow the procedure below.

(1) Write "01_H" to the EMU3nPRT (Writing to the register is enabled).

(2) Change the value of the target register or bit(s).

(3) Write "00_H" to the EMU3nPRT (Writing to the register is disabled).

25.3.2.2 EMU3n Control Register (EMU3nCTR)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0004_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	EMUST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.23 EMU3nCTR Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	EMUST	EMU operation bit 0: Resets the EMU. 1: Runs the EMU.

The EMU3n control registers are initialized when the EMUST bit is set to 0. The target registers include the EMU3n control register (EMU3nCTR) and the registers other than the EMU3n protect register (EMU3nPRT), and the internal registers of the H/W accelerator.

The EMU3nCTR registers are in the protected state after a hardware reset. When writing to the EMU3nCTR register, use the EMU3nPRT register to release protection.

If writing to the EMU3nCTR register is inadvertently attempted while it is protected, set up a wait for at least 20 cycles of the CLK_EMU_L clock to elapse, then start again after releasing protection.

25.3.2.3 EMU3n Register Value Reflection Control Register (EMU3nREFCTR)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0008_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	FPWMREFPER
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.24 EMU3nREFCTR Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	FPWMREFPER	PWM IP register value reflection control bit Controls how the register value is to be reflected in the interior of the EMU circuitry. 0: Disables the value to be reflected. 1: Enables the value to be reflected.

The EMU3nREFCTR register selects the way in which the values loaded in the registers listed in **Table 25.25** are reflected in the interior of the H/W accelerator.

FPWMREFPER bit

The FPWMREFPER bit can be used to select the way in which the values of the registers listed in **Table 25.25** are to be reflected in the circuit interior when the PWM IP is started

The state of the signals in the interior of the circuits established after a reset is the “value of the corresponding register established after a reset”.

Table 25.25 Registers the Reflection of which Value can be Enabled or Disabled by the FPWMREFPER Bit

Register Name	Symbol
EMU3n d-axis voltage correction value register	EMU3nVDCRCT
EMU3n q-axis voltage correction value register	EMU3nVQCRCT
EMU3n PWM IP electrical angle offset register	EMU3nPHI
EMU3n non-interference control coefficient angular velocity value gain register	EMU3nDECVELG
EMU3n non-interference control coefficient Ld value register	EMU3nDECLD
EMU3n non-interference control coefficient Lq value register	EMU3nDECLQ
EMU3n non-interference control coefficient magnetic flux value register	EMU3nDECFLUX
EMU3n dq-axis voltage phase angle software input register	EMU3nTHTVSFT
EMU3n dq-axis voltage value software input register	EMU3nVDQSFT
EMU3n PWM modulation peak value register	EMU3nTMAX

25.3.2.4 EMU3n IP Startup Trigger Source Select Register (EMU3nIPTRG)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0010_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	RECIPTRG	PWMIPTRG	PIIPTRG	INIPTRG[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 25.26 EMU3nIPTRG Register Contents

Bit Position	Bit	Function
7 to 5	—	These bits are read as 0. The write value should be 0.
4	RECIPTRG	Rectangle IP startup trigger select bit 0: Software trigger (RECIPSFT bit of the EMU3nIPSFT register) 1: Angle compare 0 match
3	PWMIPTRG	PWM IP startup trigger select bit 0: Software trigger (PWMIPSFT bit of the EMU3nIPSFT register) 1: PI control IP complete
2	PIIPTRG	PI control P startup trigger select bit 0: Software trigger (PIIPSFT bit of the EMU3nIPSFT register) 1: Input IP complete
1, 0	INIPTRG[1:0]	Input IP startup trigger select bit 00: Software trigger (INIPSFT bit of the EMU3nIPSFT register) 01: Compare 0 match detected 10: A/D conversion complete*1 11: Compare 0 match detected or A/D conversion complete*1

Note 1. Selected by the combination of the EMU3nCTRINMD register, bits INSTCTR[1:0] and EMU3nADEND register, bits ADEND[1:0].

For details on the angle compare 0, see **Section 25.4.7.1, Batch Rectangle IP**.

25.3.2.5 EMU3n IP Software Startup Register (EMU3nIPSFT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0014_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	RECIPSFT	PWMIPSFT	PIIPSFT	INIPSFT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.27 EMU3nIPSFT Register Contents

Bit Position	Bit	Function
7 to 4	—	These bits are read as 0. The write value should be 0.
3	RECIPSFT	Rectangle IP software startup bit Setting this bit to 1 starts the rectangle IP. The RECIPTRG bit of the EMU3nIPTRG register must be set to 0 in advance.
2	PWMIPSFT	PWM IP software startup bit Setting this bit to 1 starts the PWM IP. The PWMIPTRG bit of the EMU3nIPTRG register must be set to 0 in advance.
1	PIIPSFT	PI control IP software startup bit Setting this bit to 1 starts the PI control IP. The PIIPTRG bit of the EMU3nIPTRG register must be set to 0 in advance.
0	INIPSFT	Input IP software startup bit Setting this bit to 1 starts the input IP. The INIPTRG bit of the EMU3nIPTRG register must be set to 0 in advance.

This register is set up when starting the IPs under program control. The bits, after set to 1, are reset to 0 when the processing of the corresponding IPs is completed.

Before starting an IP under program control, clear the necessary bit of the EMU3nINTSD register associated with that IP. The bits of the EMU3nINTSD register are cleared through the EMU3nINTSDC register.

25.3.2.6 EMU3n A/D Conversion Completion Timing Select Register (EMU3nADEND)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0016_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ADEND	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 25.28 EMU3nADEND Register Contents

Bit Position	Bit	Function
7 to 2	—	These bits are read as 0. The write value should be 0.
1, 0	ADEND	A/D conversion completion timing select bit 00: A/D conversion complete 01: IIR filter channel 0 IIR filter processing complete 10: IIR filter channel 1 IIR filter processing complete 11: IIR filter channel 2 IIR filter processing complete

Note: For details, see **Section 25.4.8, A/D Conversion Control and Angle Value Latching Control.**

The value of the ADEND bit must be changed when the A/D converter is in the stopped state.

25.3.2.7 EMU3n A/D Conversion Start Trigger Source Select Register (EMU3nADTRG)

Access: Readable/writable in 8, 16, and 32 bit units.

Address: <EMU3n_base> + 0018_H

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NW7AD	NW6AD	NW5AD	NW4AD	NW3AD	NW2AD	NW1AD	NW0AD	NV7AD	NV6AD	NV5AD	NV4AD	NV3AD	NV2AD	NV1AD	NV0AD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NU7AD	NU6AD	NU5AD	NU4AD	NU3AD	NU2AD	NU1AD	NU0AD	—	—	—	—	—	CMPAD	CAVALAD	CAMOUAD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.29 EMU3nADTRG Register Contents (1/3)

Bit Position	Bit	Function
31	NW7AD	A/D conversion start trigger source enable bit (independent rectangle IP2 W-phase compare 7 match detected) 0: Disable 1: Enable
30	NW6AD	A/D conversion start trigger source enable bit (independent rectangle IP2 W-phase compare 6 match detected) 0: Disable 1: Enable
29	NW5AD	A/D conversion start trigger source enable bit (independent rectangle IP2 W-phase compare 5 match detected) 0: Disable 1: Enable
28	NW4AD	A/D conversion start trigger source enable bit (independent rectangle IP2 W-phase compare 4 match detected) 0: Disable 1: Enable
27	NW3AD	A/D conversion start trigger source enable bit (independent rectangle IP2 W-phase compare 3 match detected) 0: Disable 1: Enable
26	NW2AD	A/D conversion start trigger source enable bit (independent rectangle IP2 W-phase compare 2 match detected) 0: Disable 1: Enable
25	NW1AD	A/D conversion start trigger source enable bit (independent rectangle IP2 W-phase compare 1 match detected) 0: Disable 1: Enable
24	NW0AD	A/D conversion start trigger source enable bit (independent rectangle IP2 W-phase compare 0 match detected) 0: Disable 1: Enable

Table 25.29 EMU3nADTRG Register Contents (2/3)

Bit Position	Bit	Function
23	NV7AD	A/D conversion start trigger source enable bit (independent rectangle IP2 V-phase compare 7 match detected) 0: Disable 1: Enable
22	NV6AD	A/D conversion start trigger source enable bit (independent rectangle IP2 V-phase compare 6 match detected) 0: Disable 1: Enable
21	NV5AD	A/D conversion start trigger source enable bit (independent rectangle IP2 V-phase compare 5 match detected) 0: Disable 1: Enable
20	NV4AD	A/D conversion start trigger source enable bit (independent rectangle IP2 V-phase compare 4 match detected) 0: Disable 1: Enable
19	NV3AD	A/D conversion start trigger source enable bit (independent rectangle IP2 V-phase compare 3 match detected) 0: Disable 1: Enable
18	NV2AD	A/D conversion start trigger source enable bit (independent rectangle IP2 V-phase compare 2 match detected) 0: Disable 1: Enable
17	NV1AD	A/D conversion start trigger source enable bit (independent rectangle IP2 V-phase compare 1 match detected) 0: Disable 1: Enable
16	NV0AD	A/D conversion start trigger source enable bit (independent rectangle IP2 V-phase compare 0 match detected) 0: Disable 1: Enable
15	NU7AD	A/D conversion start trigger source enable bit (independent rectangle IP2 U-phase compare 7 match detected) 0: Disable 1: Enable
14	NU6AD	A/D conversion start trigger source enable bit (independent rectangle IP2 U-phase compare 6 match detected) 0: Disable 1: Enable
13	NU5AD	A/D conversion start trigger source enable bit (independent rectangle IP2 U-phase compare 5 match detected) 0: Disable 1: Enable
12	NU4AD	A/D conversion start trigger source enable bit (independent rectangle IP2 U-phase compare 4 match detected) 0: Disable 1: Enable
11	NU3AD	A/D conversion start trigger source enable bit (independent rectangle IP2 U-phase compare 3 match detected) 0: Disable 1: Enable

Table 25.29 EMU3nADTRG Register Contents (3/3)

Bit Position	Bit	Function
10	NU2AD	A/D conversion start trigger source enable bit (independent rectangle IP2 U-phase compare 2 match detected) 0: Disable 1: Enable
9	NU1AD	A/D conversion start trigger source enable bit (independent rectangle IP2 U-phase compare 1 match detected) 0: Disable 1: Enable
8	NU0AD	A/D conversion start trigger source enable bit (independent rectangle IP2 U-phase compare 0 match detected) 0: Disable 1: Enable
7 to 3	—	These bits are read as 0. The write value should be 0.
2	CMPAD	A/D conversion start trigger source enable bit (angle compare 0 match detected) 0: Disable 1: Enable
1	CAVALAD	A/D conversion start trigger source enable bit (carrier signal trough timing) 0: Disable 1: Enable
0	CAMOUAD	A/D conversion start trigger source enable bit ((carrier signal peak timing) 0: Disable 1: Enable

Note: For details, see **Section 25.4.8, A/D Conversion Control and Angle Value Latching Control**.

25.3.2.8 EMU3n A/D Conversion Start Trigger Source Determination Register (EMU3nADMON)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $001C_H$

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NW7ADM	NW6ADM	NW5ADM	NW4ADM	NW3ADM	NW2ADM	NW1ADM	NW0ADM	NV7ADM	NV6ADM	NV5ADM	NV4ADM	NV3ADM	NV2ADM	NV1ADM	NV0ADM
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NU7ADM	NU6ADM	NU5ADM	NU4ADM	NU3ADM	NU2ADM	NU1ADM	NU0ADM	—	—	—	—	SFTADM	CMPADM	CAVALADM	CAMOUADM
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.30 EMU3nADMON Register Contents (1/3)

Bit Position	Bit	Function
31	NW7ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 W-phase compare 7 match detected) 0: No source. 1: Source occurred.
30	NW6ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 W-phase compare 6 match detected) 0: No source. 1: Source occurred.
29	NW5ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 W-phase compare 5 match detected) 0: No source. 1: Source occurred.
28	NW4ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 W-phase compare 4 match detected) 0: No source. 1: Source occurred.
27	NW3ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 W-phase compare 3 match detected) 0: No source. 1: Source occurred.
26	NW2ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 W-phase compare 2 match detected) 0: No source. 1: Source occurred.
25	NW1ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 W-phase compare 1 match detected) 0: No source. 1: Source occurred.
24	NW0ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 W-phase compare 0 match detected) 0: No source. 1: Source occurred.

Table 25.30 EMU3nADMON Register Contents (2/3)

Bit Position	Bit	Function
23	NV7ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 V-phase compare 7 match detected) 0: No source. 1: Source occurred.
22	NV6ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 V-phase compare 6 match detected) 0: No source. 1: Source occurred.
21	NV5ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 V-phase compare 5 match detected) 0: No source. 1: Source occurred.
20	NV4ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 V-phase compare 4 match detected) 0: No source. 1: Source occurred.
19	NV3ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 V-phase compare 3 match detected) 0: No source. 1: Source occurred.
18	NV2ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 V-phase compare 2 match detected) 0: No source. 1: Source occurred.
17	NV1ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 V-phase compare 1 match detected) 0: No source. 1: Source occurred.
16	NV0ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 V-phase compare 0 match detected) 0: No source. 1: Source occurred.
15	NU7ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 U-phase compare 7 match detected) 0: No source. 1: Source occurred.
14	NU6ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 U-phase compare 6 match detected) 0: No source. 1: Source occurred.
13	NU5ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 U-phase compare 5 match detected) 0: No source. 1: Source occurred.
12	NU4ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 U-phase compare 4 match detected) 0: No source. 1: Source occurred.
11	NU3ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 U-phase compare 3 match detected) 0: No source. 1: Source occurred.

Table 25.30 EMU3nADMON Register Contents (3/3)

Bit Position	Bit	Function
10	NU2ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 U-phase compare 2 match detected) 0: No source. 1: Source occurred.
9	NU1ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 U-phase compare 1 match detected) 0: No source. 1: Source occurred.
8	NU0ADM	A/D conversion start trigger source determination bit (independent rectangle IP2 U-phase compare 0 match detected) 0: No source. 1: Source occurred.
7 to 4	—	These bits are read as 0. The write value should be 0.
3	SFTADM	A/D conversion start trigger source determination bit (software trigger) 0: No source. 1: Source occurred.
2	CMPADM	A/D conversion start trigger source determination bit (angle compare 0 match detected) 0: No source. 1: Source occurred.
1	CAVALADM	A/D conversion start trigger source determination bit (carrier signal trough timing) 0: No source. 1: Source occurred.
0	CAMOUADM	A/D conversion start trigger source determination bit (carrier signal peak timing) 0: No source. 1: Source occurred.

Note: For details, see **Section 25.4.8, A/D Conversion Control and Angle Value Latching Control**.

This register is used to monitor the source of the A/D conversion start triggers that occurred. Each bit of the register is set to 1 when the source of the corresponding A/D conversion start trigger occurred. The bit is cleared by setting the corresponding bit of the EMU3nADMONC register.

25.3.2.9 EMU3n A/D Conversion Start Trigger Source Determination Clear Register (EMU3nADMONC)

Access: Readable/writable in 8, 16, and 32 bit units.

Address: <EMU3n_base> + 0020_H

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NW7 ADMC	NW6 ADMC	NW5 ADMC	NW4 ADMC	NW3 ADMC	NW2 ADMC	NW1 ADMC	NW0 ADMC	NV7 ADMC	NV6 ADMC	NV5 ADMC	NV4 ADMC	NV3 ADMC	NV2 ADMC	NV1 ADMC	NV0 ADMC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NU7 ADMC	NU6 ADMC	NU5 ADMC	NU4 ADMC	NU3 ADMC	NU2 ADMC	NU1 ADMC	NU0 ADMC	—	—	—	—	SFT ADMC	CMP ADMC	CAVAL ADMC	CAMOU ADMC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.31 EMU3nADMONC Register Contents (1/2)

Bit Position	Bit	Function
31	NW7ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 W-phase compare 7 match detected) Setting this bit to 1 resets the NW7ADM bit of the EMU3nADMON register to 0.
30	NW6ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 W-phase compare 6 match detected) Setting this bit to 1 resets the NW6ADM bit of the EMU3nADMON register to 0.
29	NW5ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 W-phase compare 5 match detected) Setting this bit to 1 resets the NW5ADM bit of the EMU3nADMON register to 0.
28	NW4ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 W-phase compare 4 match detected) Setting this bit to 1 resets the NW4ADM bit of the EMU3nADMON register to 0.
27	NW3ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 W-phase compare 3 match detected) Setting this bit to 1 resets the NW3ADM bit of the EMU3nADMON register to 0.
26	NW2ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 W-phase compare 2 match detected) Setting this bit to 1 resets the NW2ADM bit of the EMU3nADMON register to 0.
25	NW1ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 W-phase compare 1 match detected) Setting this bit to 1 resets the NW1ADM bit of the EMU3nADMON register to 0.
24	NW0ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 W-phase compare 0 match detected) Setting this bit to 1 resets the NW0ADM bit of the EMU3nADMON register to 0.
23	NV7ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 V-phase compare 7 match detected) Setting this bit to 1 resets the NV7ADM bit of the EMU3nADMON register to 0.
22	NV6ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 V-phase compare 6 match detected) Setting this bit to 1 resets the NV6ADM bit of the EMU3nADMON register to 0.

Table 25.31 EMU3nADMONC Register Contents (2/2)

Bit Position	Bit	Function
21	NV5ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 V-phase compare 5 match detected) Setting this bit to 1 resets the NV5ADM bit of the EMU3nADMON register to 0.
20	NV4ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 V-phase compare 4 match detected) Setting this bit to 1 resets the NV4ADM bit of the EMU3nADMON register to 0.
19	NV3ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 V-phase compare 3 match detected) Setting this bit to 1 resets the NV3ADM bit of the EMU3nADMON register to 0.
18	NV2ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 V-phase compare 2 match detected) Setting this bit to 1 resets the NV2ADM bit of the EMU3nADMON register to 0.
17	NV1ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 V-phase compare 1 match detected) Setting this bit to 1 resets the NV1ADM bit of the EMU3nADMON register to 0.
16	NV0ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 V-phase compare 0 match detected) Setting this bit to 1 resets the NV0ADM bit of the EMU3nADMON register to 0.
15	NU7ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 U-phase compare 7 match detected) Setting this bit to 1 resets the NU7ADM bit of the EMU3nADMON register to 0.
14	NU6ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 U-phase compare 6 match detected) Setting this bit to 1 resets the NU6ADM bit of the EMU3nADMON register to 0.
13	NU5ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 U-phase compare 5 match detected) Setting this bit to 1 resets the NU5ADM bit of the EMU3nADMON register to 0.
12	NU4ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 U-phase compare 4 match detected) Setting this bit to 1 resets the NU4ADM bit of the EMU3nADMON register to 0.
11	NU3ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 U-phase compare 3 match detected) Setting this bit to 1 resets the NU3ADM bit of the EMU3nADMON register to 0.
10	NU2ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 U-phase compare 2 match detected) Setting this bit to 1 resets the NU2ADM bit of the EMU3nADMON register to 0.
9	NU1ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 U-phase compare 1 match detected) Setting this bit to 1 resets the NU1ADM bit of the EMU3nADMON register to 0.
8	NU0ADMC	A/D conversion start trigger source determination register clear bit (independent rectangle IP2 U-phase compare 0 match detected) Setting this bit to 1 resets the NU0ADM bit of the EMU3nADMON register to 0.
7 to 4	—	These bits are read as 0. The write value should be 0.
3	SFTADMC	A/D conversion start trigger source determination register clear bit (software trigger) Setting this bit to 1 resets the SFTADM bit of the EMU3nADMON register to 0.
2	CMPADMC	A/D conversion start trigger source determination register clear bit (angle compare 0 match detected) Setting this bit to 1 resets the CMPADM bit of the EMU3nADMON register to 0.
1	CAVALADMC	A/D conversion start trigger source determination register clear bit (carrier counter trough) Setting this bit to 1 resets the CAVALADM bit of the EMU3nADMON register to 0.
0	CAMOUADMC	A/D conversion start trigger source determination register clear bit (carrier counter peak) Setting this bit to 1 resets the CAMOUADM bit of the EMU3nADMON register to 0.

Note: For details, see **Section25.4.8, A/D Conversion Control and Angle Value Latching Control**.

The bits of these register are automatically reset to 0 after they are set to 1. It is invalid to write a 0 to these bits. They are always read as 0.

25.3.2.10 EMU3n Data Delay Count Value Register (EMU3nDDCNT)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0024_H

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDDATA(unsigned)[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADDATA(unsigned)[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.32 EMU3nDDCNT Register Contents

Bit Position	Bit	Function
31 to 16	RDDATA[15:0]	Angle value latch delay setting bit The input IP latches the resolver angle and electrical angle value which are outputs from the angle generation to the input IP after the cycles specified in this register if an A/D conversion startup source specified by the IP EMU3nADTRG register occurs.
15 to 0	ADDATA[15:0]	A/D conversion trigger delay setting bit Output an A/D conversion start trigger to the A/D converter after the cycles specified in this register if an A/D conversion startup source specified by the EMU3nADTRG register occurs. Any new A/D conversion startup source occurring while the delay time processing is in progress is invalid.

Note: For details, see **Section 25.4.8, A/D Conversion Control and Angle Value Latching Control**.

This register defines the delay time of the A/D conversion trigger outputs and R/D conversion outputs. The conversions may be started with this register kept at their initial value of 0000_H. Any A/D conversion start trigger occurring while the delay counter is running is invalid.

25.3.2.11 EMU3n Interrupt Source Select k Register (EMU3nINTk) (k = 0 to 7)

Access: Readable/writable in 32-bit units.

Address: EMU3nINT0: <EMU3n_base> + 0028_H

EMU3nINT1: <EMU3n_base> + $002C_H$

EMU3nINT2: <EMU3n_base> + 0030_H

EMU3nINT3: <EMU3n_base> + 0034_H

EMU3nINT4: <EMU3n_base> + 0038_H

EMU3nINT5: <EMU3n_base> + $003C_H$

EMU3nINT6: <EMU3n_base> + 0040_H

EMU3nINT7: <EMU3n_base> + 0044_H

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FWGBINT	FWGAINT	KCLINT	NRECWINT	NRECVINT	NRECUINT	IIR2INT	IIR1INT	IIR0INT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADINT	PMT2CPINT	PMT2OFINT	PMTOFINT	IRECWINT	IRECVINT	IRECUINT	CMP1INT	CMP0INT	CARMINT	CARVINT	CBUFINT	RECINT	PWMINT	PIINT	ININT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.33 EMU3nINTk Register Contents (1/3)

Bit Position	Bit	Function
31 to 25	—	These bits are read as 0. The write value should be 0.
24	FWGBINT	Interrupt source value bit k (angle generation IP WAIT transition detected) 0: Disables the interrupt 1: Enables the interrupt
23	FWGAINT	Interrupt source value bit k (input, PI control, PWM IP WAIT transition detected) 0: Disables the interrupt 1: Enables the interrupt
22	KCLINT	Interrupt source value bit k (Kirchhoff's current law violation detected) 0: Disables the interrupt 1: Enables the interrupt
21	NRECWINT	Interrupt source value bit k (independent rectangle IP 2 W-phase angle compare 0 match detected) 0: Disables the interrupt 1: Enables the interrupt
20	NRECVINT	Interrupt source value bit k (independent rectangle IP2 V-phase angle compare 0 match detected) 0: Disables the interrupt 1: Enables the interrupt
19	NRECUINT	Interrupt source value bit k (independent rectangle IP2 U-phase angle compare 0 match detected) 0: Disables the interrupt 1: Enables the interrupt

Table 25.33 EMU3nINTk Register Contents (2/3)

Bit Position	Bit	Function
18	IIR2INT	Interrupt source value bit k (IIR filter channel 2) 0: Disables the interrupt 1: Enables the interrupt
17	IIR1INT	Interrupt source value bit k (IIR filter channel 1) 0: Disables the interrupt 1: Enables the interrupt
16	IIR0INT	Interrupt source value bit k (IIR filter channel 0) 0: Disables the interrupt 1: Enables the interrupt
15	ADINT	Interrupt source value bit k (A/D conversion complete) 0: Disables the interrupt 1: Enables the interrupt
14	PMT2CPINT	Interrupt source value bit k (resolver angle measurement timer capture) 0: Disables the interrupt 1: Enables the interrupt
13	PMT2OFINT	Interrupt source value bit k (resolver angle measurement timer overflow) 0: Disables the interrupt 1: Enables the interrupt
12	PMTOFINT	Interrupt source value bit k (pulse period measurement timer overflow) 0: Disables the interrupt 1: Enables the interrupt
11	IRECWINT	Interrupt source value bit k (independent rectangle IP1W-phase angle compare 0 match detected) 0: Disables the interrupt 1: Enables the interrupt
10	IRECVINT	Interrupt source value bit k (independent rectangle IP1V-phase angle compare 0 match detected) 0: Disables the interrupt 1: Enables the interrupt
9	IRECUINT	Interrupt source value bit k (independent rectangle IP1U-phase angle compare 0 match detected) 0: Disables the interrupt 1: Enables the interrupt
8	CMP1INT	Interrupt source value bit k (angle compare 1 match detected) 0: Disables the interrupt 1: Enables the interrupt
7	CMP0INT	Interrupt source value bit k (angle compare 0 match detected) 0: Disables the interrupt 1: Enables the interrupt
6	CARMINT	Interrupt source value bit k (carrier counter peak) 0: Disables the interrupt 1: Enables the interrupt
5	CARVINT	Interrupt source value bit k (carrier counter trough) 0: Disables the interrupt 1: Enables the interrupt
4	CBUFINT	Interrupt source value bit k (checking buffering complete) 0: Disables the interrupt 1: Enables the interrupt
3	RECINT	Interrupt source value bit k (rectangle IP complete) 0: Disables the interrupt 1: Enables the interrupt

Table 25.33 EMU3nINTk Register Contents (3/3)

Bit Position	Bit	Function
2	PWMINT	Interrupt source value bit k (PWM IP complete) 0: Disables the interrupt 1: Enables the interrupt
1	PIINT	Interrupt source value bit k (PI control IP complete) 0: Disables the interrupt 1: Enables the interrupt
0	ININT	Interrupt source value bit k (input IP complete) 0: Disables the interrupt 1: Enables the interrupt

Note: For details, see **Section 25.4.15, Interrupt Control**.

25.3.2.12 EMU3n Interrupt Source Determination Register (EMU3nINTSD)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + 0048_H

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FWGBIF	FWGAIF	KCLIF	NRECWIF	NRECVIF	NRECUIF	IIR2IF	IIR1IF	IIR0IF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADIF	PMT2CPIF	PMT2OFIF	PMTOFIF	IRECWIF	IRECVIF	IRECUIF	CMP1IF	CMP0IF	CARMIF	CARVIF	CBUFIF	RECIF	PWMIF	PIIF	INIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.34 EMU3nINTSD Register Contents (1/3)

Bit Position	Bit	Function
31 to 25	—	When read, the value read is undefined.
24	FWGBIF	Interrupt source determination flag bit (angle generation IP WAIT transition detected) 0: Angle generation IP WAIT transition not occurred. 1: Angle generation IP WAIT transition occurred.
23	FWGAIF	Interrupt source determination flag bit (input, PI control, PWM IPWAIT transition detected) 0: Input, PI control, PWM IP WAIT transition not occurred. 1: Input, PI control, PWM IP WAIT transition occurred.
22	KCLIF	Interrupt source determination flag bit (Kirchhoff's current law violation detected) 0: Kirchhoff's current law violation not detected. 1: Kirchhoff's current law violation detected.
21	NRECWIF	Interrupt source determination flag bit (independent rectangle IP2 W-phase angle compare 0 match detected) 0: Independent rectangle IP2 W-phase angle compare 0 match not detected. 1: Independent rectangle IP2 W-phase angle compare 0 match detected.
20	NRECVIF	Interrupt source determination flag bit (independent rectangle IP2 V-phase angle compare 0 match detected.) 0: Independent rectangle IP2 V-phase angle compare 0 match not detected. 1: Independent rectangle IP2 V-phase angle compare 0 match detected.
19	NRECUIF	Interrupt source determination flag bit (independent rectangle IP2 U-phase angle compare 0 match detected.) 0: Independent rectangle IP2 U-phase angle compare 0 match not detected. 1: Independent rectangle IP2 U-phase angle compare 0 match detected.
18	IIR2IF	Interrupt source determination flag bit (IIR filter channel 2 complete) 0: IIR filter channel 2 completion not occurred. 1: IIR filter channel 2 completion occurred.

Table 25.34 EMU3nINTSD Register Contents (2/3)

Bit Position	Bit	Function
17	IIR1IF	Interrupt source determination flag bit (IIR filter channel 1 complete) 0: IIR filter channel 1 completion not occurred. 1: IIR filter channel 1 completion occurred.
16	IIR0IF	Interrupt source determination flag bit (IIR filter channel 0 complete) 0: IIR filter channel 0 completion not occurred. 1: IIR filter channel 0 completion occurred.
15	ADIF	Interrupt source determination flag bit (A/D conversion complete) 0: A/D conversion not completed. 1: A/D conversion completed.
14	PMT2CPIF	Interrupt source determination flag bit (resolver angle measurement timer capture) 0: Resolver angle measurement timer capture not occurred. 1: Resolver angle measurement timer capture occurred.
13	PMT2OFIF	Interrupt source determination flag bit (resolver angle measurement timer overflow) 0: Resolver angle measurement timer overflow not occurred. 1: Resolver angle measurement timer overflow occurred.
12	PMTOFIF	Interrupt source determination flag bit (pulse period measurement timer overflow) 0: Pulse period measurement timer overflow not occurred. 1: Pulse period measurement timer overflow occurred.
11	IRECWIF	Interrupt source determination flag bit (independent rectangle IP1W-phase angle compare 0 match detected) 0: Independent rectangle IP1W-phase angle compare 0 match not detected. 1: Independent rectangle IP1W-phase angle compare 0 match detected.
10	IRECVIF	Interrupt source determination flag bit (independent rectangle IP1V-phase angle compare 0 match detected.) 0: Independent rectangle IP1V-phase angle compare 0 match not detected. 1: Independent rectangle IP1V-phase angle compare 0 match detected.
9	IRECUIF	Interrupt source determination flag bit (independent rectangle IP1U-phase angle compare 0 match detected.) 0: Independent rectangle IP1U-phase angle compare 0 match not detected. 1: Independent rectangle IP1U-phase angle compare 0 match detected.
8	CMP1IF	Interrupt source determination flag bit (angle compare 1 match detected) 0: Angle compare 1 match not detected. 1: Angle compare 1 match detected.
7	CMP0IF	Interrupt source determination flag bit (angle compare 0 match detected.) 0: Angle compare 0 match not detected. 1: Angle compare 0 match detected.
6	CARMIF	Interrupt source determination flag bit (carrier counter peak) 0: Carrier counter peak not occurred. 1: Carrier counter peak occurred.
5	CARVIF	Interrupt source determination flag bit (carrier counter trough) 0: Carrier counter peak not occurred. 1: Carrier counter peak occurred.
4	CBUFIF	Interrupt source determination flag bit (checking buffering complete) 0: Checking register buffering not occurred. 1: Checking register buffering occurred.

Table 25.34 EMU3nINTSD Register Contents (3/3)

Bit Position	Bit	Function
3	RECIF	Interrupt source determination flag bit (rectangle IP complete) 0: rectangle IP completion not occurred. 1: rectangle IP completion occurred.
2	PWMIF	Interrupt source determination flag bit (PWM IP complete) 0: PWM IP completion not occurred. 1: PWM IP completion occurred.
1	PIIF	Interrupt source determination flag bit (PI control IP complete) 0: PI control IP completion not occurred. 1: PI control IP completion occurred.
0	INIF	Interrupt source determination flag bit (input IP complete) 0: Input IP completion not occurred. 1: Input IP completion occurred.

Note: For details, see **Section 25.4.15, Interrupt Control**.

The bits of this register are set to 1 when the operation initiated by the corresponding sources is completed. They can be used to determine the sources of interrupts that occurred.

25.3.2.13 EMU3n Interrupt Source Determination Clear Register (EMU3nINTSDC)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + $004C_H$

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	FWGBIFC	FWGAIFC	KCLIFC	NRECWIFC	NRECVIFC	NRECUIFC	IIR2IFC	IIR1IFC	IIR0IFC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADIFC	PMT2CPIFC	PMT2OFIFC	PMTOFIFC	IRECWIFC	IRECVIFC	IRECUIFC	CMP1IFC	CMP0IFC	CARMIFC	CARVIFC	CBUFIFC	RECIFC	PWMIFC	PIIFC	INIFC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.35 EMU3nINTSDC Register Contents (1/2)

Bit Position	Bit	Function
31 to 25	—	These bits are read as 0. The write value should be 0.
24	FWGBIFC	Interrupt source determination flag clear bit (angle generation IP WAIT transition detected) Setting this bit to 1 resets the FWGBIF bit of the EMU3nINTSD register to 0.
23	FWGAIFC	Interrupt source determination flag clear bit (input, PI control, PWM IPWAIT transition detected) Setting this bit to 1 resets the FWGAIF bit of the EMU3nINTSD register to 0.
22	KCLIFC	Interrupt source determination flag clear bit (Kirchhoff's current law violation detected) Setting this bit to 1 resets the KCLIF bit of the EMU3nINTSD register to 0.
21	NRECWIFC	Interrupt source determination flag clear bit (independent rectangle IP2 W-phase angle compare 0 match detected) Setting this bit to 1 resets the NRECWIF bit of the EMU3nINTSD register to 0.
20	NRECVIFC	Interrupt source determination flag clear bit (independent rectangle IP2 V-phase angle compare 0 match detected) Setting this bit to 1 resets the NRECVIF bit of the EMU3nINTSD register to 0.
19	NRECUIFC	Interrupt source determination flag clear bit (independent rectangle IP2 U-phase angle compare 0 match detected) Setting this bit to 1 resets the NRECUIF bit of the EMU3nINTSD register to 0.
18	IIR2IFC	Interrupt source determination flag clear bit (IIR filter channel 2 complete) Setting this bit to 1 resets the IIR2IF bit of the EMU3nINTSD register to 0.
17	IIR1IFC	Interrupt source determination flag clear bit (IIR filter channel 1 complete) Setting this bit to 1 resets the IIR1IF bit of the EMU3nINTSD register to 0.
16	IIR0IFC	Interrupt source determination flag clear bit (IIR filter channel 0 complete) Setting this bit to 1 resets the IIR0IF bit of the EMU3nINTSD register to 0.
15	ADIFC	Interrupt source determination flag clear bit (A/D conversion complete) Setting this bit to 1 resets the ADIF bit of the EMU3nINTSD register to 0.
14	PMT2CPIFC	Interrupt source determination flag clear bit (resolver angle measurement timer capture) Setting this bit to 1 resets the PMT2CPIF bit of the EMU3nINTSD register to 0.
13	PMT2OFIFC	Interrupt source determination flag clear bit (resolver angle measurement timer overflow) Setting this bit to 1 resets the PMT2OFIF bit of the EMU3nINTSD register to 0.
12	PMTOFIFC	Interrupt source determination flag clear bit (pulse period measurement timer overflow) Setting this bit to 1 resets the PMTOFIF bit of the EMU3nINTSD register to 0.

Table 25.35 EMU3nINTSDC Register Contents (2/2)

Bit Position	Bit	Function
11	IRECWIFC	Interrupt source determination flag clear bit (independent rectangle IP1 W-phase angle compare 0 match detected) Setting this bit to 1 resets the IRECWIF bit of the EMU3nINTSD register to 0.
10	IRECVIFC	Interrupt source determination flag clear bit (independent rectangle IP1 V-phase angle compare 0 match detected.) Setting this bit to 1 resets the IRECVIF bit of the EMU3nINTSD register to 0.
9	IRECUIFC	Interrupt source determination flag clear bit (independent rectangle IP1 U-phase angle compare 0 match detected.) Setting this bit to 1 resets the IRECUIF bit of the EMU3nINTSD register to 0.
8	CMP1IFC	Interrupt source determination flag clear bit (angle compare 1 match detected) Setting this bit to 1 resets the CMP1IF bit of the EMU3nINTSD register to 0.
7	CMP0IFC	Interrupt source determination flag clear bit (angle compare 0 match detected.) Setting this bit to 1 resets the CMP0IF bit of the EMU3nINTSD register to 0.
6	CARMIFC	Interrupt source determination flag clear bit (carrier counter peak) Setting this bit to 1 resets the CARMIF bit of the EMU3nINTSD register to 0.
5	CARVIFC	Interrupt source determination flag clear bit (carrier counter trough) Setting this bit to 1 resets the CARVIF bit of the EMU3nINTSD register to 0.
4	CBUFIFC	Interrupt source determination flag clear bit (checking buffering complete) Setting this bit to 1 resets the CBUFIF bit of the EMU3nINTSD register to 0.
3	RECIFC	Interrupt source determination flag clear bit (rectangle IP complete) Setting this bit to 1 resets the RECIF bit of the EMU3nINTSD register to 0.
2	PWMIFC	Interrupt source determination flag clear bit (PWM IP complete) Setting this bit to 1 resets the PWMIF bit of the EMU3nINTSD register to 0.
1	PIIFC	Interrupt source determination flag clear bit (PI control IP complete) Setting this bit to 1 resets the PIIF bit of the EMU3nINTSD register to 0.
0	INIFC	Interrupt source determination flag clear bit (input IP complete) Setting this bit to 1 resets the INIF bit of the EMU3nINTSD register to 0.

Note: For details, see **Section 25.4.15, Interrupt Control**.

The bits of these registers are automatically reset to 0 after they are set to 1. It is invalid to write a 0 to these bits. They are always read as 0.

25.3.2.14 EMU3n Overflow Detection Result Register (EMU3nOFMON)

Access: Readable in 8-bit units.

Address: <EMU3n_base> + 0050_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PWMIPOF	PIIPOF	INIPOF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 25.36 EMU3nOFMON Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMIPOF	PWM IP overflow detection flag 0: No overflow detected. 1: Overflow detected.
1	PIIPOF	PI control IP overflow detection flag 0: No overflow detected. 1: Overflow detected.
0	INIPOF	Input IP overflow detection flag 0: No overflow detected. 1: Overflow detected.

25.3.2.15 EMU3n Zero Division Detection Result Register (EMU3nZDMON)

Access: Readable in 8-bit units.

Address: <EMU3n_base> + 0051_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PWMIPZD	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 25.37 EMU3nZDMON Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMIPZD	PWM IP zero division detection flag 0: No zero division detected. 1: Zero division detected.
1, 0	—	These bits are read as 0. The write value should be 0.

25.3.2.16 EMU3n Overflow Detection Result Clear Register (EMU3nOFMONC)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0052_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PWMIPOFC	PIIPOFC	INIPOFC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.38 EMU3nOFMONC Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMIPOFC	PWM IP overflow detection flag clear bit Setting this bit to 1 resets the PWMIPOF bit of the EMU3nOFMON register to 0.
1	PIIPOFC	PI control IP overflow detection flag clear bit Setting this bit to 1 resets the PIIPOF bit of the EMU3nOFMON register to 0.
0	INIPOFC	Input IP overflow detection flag clear bit Setting this bit to 1 resets the INIPOF bit of the EMU3nOFMON register to 0.

The bits of these registers are automatically reset to 0 after they are set to 1. It is invalid to write a 0 to these bits. They are always read as 0.

25.3.2.17 EMU3n Zero Division Detection Result Clear Register (EMU3nZDMONC)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0053_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PWM IPZDC	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R

Table 25.39 EMU3nZDMONC Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMIPZDC	PWM IP zero division detection flag clear bit Setting this bit to 1 resets the PWMIPZD bit of the EMU3nZDMON register to 0.
1, 0	—	These bits are read as 0. The write value should be 0.

The bits of these registers are automatically reset to 0 after they are set to 1. It is invalid to write a 0 to these bits. They are always read as 0.

25.3.2.18 EMU3n Pulse Period Measurement Timer Control Register (EMU3nPMTCTR)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0060_H

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STTRG	—	—	—	—	—	—	—	—	—	—	—	—	—	OVFSW	STR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 25.40 EMU3nPMTCTR Register Contents

Bit Position	Bit	Function
15	STTRG	Software trigger bit Setting this bit to 1 generates a software trigger. This bit is automatically reset to 0 after it is set to 1. It is invalid to write a 0. The bit is always read as 0.
14 to 2	—	These bits are read as 0. The write value should be 0.
1	OVFSW	Overflow output destination switching bit 0: EMU3nPMTOF register 1: Interrupt request This bit allows the destination to be notified when an EMU3nPMTCNT counter overflow occurs to be selected.
0	STR	Count start bit 0: Stops counter count operation. 1: Starts counter count operation.

Note: For details, see **Section 25.4.10, Pulse Cycle Measurement Timer**.

25.3.2.19 EMU3n Pulse Period Measurement Timer Counter Register (EMU3nPMTCNT)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0064_H

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	DATA (unsigned)								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.41 EMU3nPMTCNT Register Contents

Bit Position	Bit	Function
31 to 25	—	These bits are read as 0. The write value should be 0.
24 to 0	DATA	Data bits (unsigned integer) These bits are loaded with a 25-bit up-counter value which is readable. The counter value can be updated by writing it in these bits.

Note: For details, see **Section 25.4.10, Pulse Cycle Measurement Timer**.

An EMU3nPMTCNT counter is a 25-bit counter that counts up on CLKC_HSB when the STR bit of the corresponding EMU3nPMTCTR register is set to 1. The value of this register can be rewritten regardless of whether the counter (capture operation) is running or stopped. Reading this register yields a counter value.

The EMU3nPMTCNT counter resets to $0000\ 0000_H$ when its value is loaded into the EMU3nPMTCAP register, that is capture operation, or when an overflow occurs in the EMU3nPMTCNT counter. The counter subsequently continues counting.

25.3.2.20 EMU3n Pulse Period Measurement Timer Capture Register (EMU3nPMTCAP)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + 0068_H

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	DATA (unsigned)								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.42 EMU3nPMTCAP Register Contents

Bit Position	Bit	Function
31 to 25	—	These bits are read as 0. The write value should be 0.
24 to 0	DATA	Data bits (unsigned integer) These bits store a 25-bit capture value which is readable.

Note: For details, see **Section 25.4.10, Pulse Cycle Measurement Timer**.

An EMU3nPMTCAP register is a read-only register. It is loaded with the counter value from the EMU3nPMTCNT counter on a rising edge of the Z-phase signal or a software trigger (setting the STTRG bit of the EMU3nPMTCTR register to 1).

25.3.2.21 EMU3n Pulse Period Measurement Timer Overflow Register (EMU3nPMTOF)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $006C_H$

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.43 EMU3nPMTOF Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	OVF	Overflow flag bit 0: No overflow occurred. 1: Overflow occurred. The bit can be set to 1 manually if it is set to 0.

Note: For details, see **Section 25.4.10, Pulse Cycle Measurement Timer**.

The EMU3nPMTOF register is enabled when the OVFSW bit of the corresponding EMU3nPMTCTR register is set to 0.

OVF flag

[Set to 0 when:]

- A 0 is written to the OVF flag when the flag is set to 1.

[Set to 1 when]

- An overflow (01FF $FFFF_H \rightarrow 0000\ 0000_H$) occurs in the EMU3nPMTCNT counter when the OVFSW bit of the EMU3nPMTCTR register is set to 0 (EMU3nPMTOF register selected).
- A 1 is written to the OVF flag when the flag is set to 0.

25.3.2.22 EMU3n Resolver Angle Measurement Timer Control Register (EMU3nPMT2CTR)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0070_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	EN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.44 EMU3nPMT2CTR Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	EN	Counter enable bit 0: Disables the counter for counting. 1: Enables the counter for counting.

Note: For details, see **Section 25.4.11, Resolver Angle Measurement timer**.

25.3.2.23 EMU3n Resolver Angle Measurement Timer Soft Trigger Register (EMU3nPMT2SFT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0074_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SCAPTRG
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.45 EMU3nPMT2SFT Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	SCAPTRG	Software capture trigger bit Setting this bit to 1, loads the value of the EMU3nPMT2CNT register into the EMU3nPMT2CAP register and a 0 value into the EMU3nPMT2CNT register (capture operation). The bit is automatically reset to 0 after it is set to 1. It is invalid to write a 0. The bit is always read as 0.

Note: For details, see **Section 25.4.11, Resolver Angle Measurement timer**.

25.3.2.24 EMU3n Resolver Angle Measurement Timer Counter Register (EMU3nPMT2CNT)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0078_H

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.46 EMU3nPMT2CNT Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (unsigned integer) These bits store a 32-bit up-counter count value. The count value is updated every time a value is written into these bit positions.

Note: For details, see **Section 25.4.11, Resolver Angle Measurement timer**.

A resolver angle measurement timer is a 32-bit up-counter that counts up on each cycle in synchronization with CLKC_HSB. Its value can be rewritten regardless of whether the timer is running or stopped. Reading this register yields the count value of the timer.

The EMU3nPMT2CNT register is reset to 0 when the value of the EMU3nPMT2CNT register is loaded into the EMU3nPMT2CAP register as a capture operation or when a counter overflow occurs in the EMU3nPMT2CNT register. Subsequently, the timer continues count-up processing.

25.3.2.25 EMU3n Resolver Angle Measurement Timer Capture Register (EMU3nPMT2CAP)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $007C_H$

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.47 EMU3nPMT2CAP Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (unsigned integer) These bits are loaded with the value of the EMU3nPMT2CNT register as a capture operation.

Note: For details, see **Section 25.4.11, Resolver Angle Measurement timer**.

The value of the EMU3nPMT2CNT register is loaded into the EMU3nPMT2CAP register on a “rising edge of the Z-phase signal”, which is selected by the EMU3nPMT2INVL register, or a software trigger (setting the STTRG bit of the EMU3nPMTCTR register to 1).

25.3.2.26 EMU3n Resolver Angle Measurement Timer Capture Interval Value Register (EMU3nPMT2INVL)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0080_H

Value after reset: $0B_H$

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	DATA (unsigned)				
Value after reset	0	0	0	0	1	0	1	1
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 25.48 EMU3nPMT2INVL Register Contents

Bit Position	Bit	Function
7 to 5	—	These bits are read as 0. The write value should be 0.
4 to 0	DATA	Data bits (unsigned integer) Select the capture interval according to the Z-phase signal or resolver angle as follows: $1F_H$: Capture on each rising edge of the Z-phase signal. $1E_H$ to 00_H: Inhibited

Note: For details, see **Section 25.4.11, Resolver Angle Measurement timer**.

The value this register must be changed after stopping the resolver angle measurement timer by setting the EN bit of the EMU3nPMT2CTR register to 0. It is inhibited to change the value of this register while the resolver angle measurement timer is running.

25.3.2.27 EMU3n A/D Conversion Start Soft Trigger Register (EMU3nADSFTTRG)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0084_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SFTAD
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.49 EMU3nADSFTTRG Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	SFTAD	A/D startup bit (software trigger) 0: Disable 1: Enable

Note: For details, see **Section 25.4.8, A/D Conversion Control and Angle Value Latching Control**.

25.3.2.28 EMU3n H/W Arithmetic Block IDLE Time Startup Command A0 Register (EMU3nFUNCIDLEGRPA0)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0090_H

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SFTEN	—	—	—	IP	—	—	—	—	—	FUNC[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.50 EMU3nFUNCIDLEGRPA0 Register Contents

Bit Position	Bit	Function
15 to 13	—	These bits are read as 0. The write value should be 0.
12	SFTEN	Software startup enable bit This bit enables software startup of the input IP at a write access timing. If software startup is disabled, the input IP is started according to a startup source of the input IP. 0: Software startup disabled. 1: Software startup enabled. **CAUTION:** Any startup request issued when the IP bit of the EMU3nFUNCIDLEGRPA0 register is set to 0 is invalid.
11 to 9	—	These bits are read as 0. The write value should be 0.
8	IP	Startup IP specification enable bit This bit enables or disables the specification of starting H/W arithmetic block to be used when the input IP is started. 0: Disable. 1: Enable.
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	FUNC[2:0]	Startup H/W arithmetic block select bits These bits select the H/W arithmetic block to be initiated when starting the input IP.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

- When starting the input IP on an input IP startup source

Register Value	Corresponding IP and H/W Arithmetic Block
01_00_H	Input IP's Func(input0) “No Operation”
01_01_H	Input IP's Func(input1) “Electrical angle source select”
01_04_H	Input IP's Func(input4) “Motor current value calculation”
01_05_H	Input IP's Func(input5) “dq-axis current conversion”

- When software-starting the input IP at timing of EMU3nFUNCIDLEGRPA0 register write access

Register Value	Corresponding IP and H/W Arithmetic Block
11_00_H	Input IP's Func(input0) “No Operation”
11_01_H	Input IP's Func(input1) “Electrical angle source select”
11_04_H	Input IP's Func(input4) “Motor current value calculation”
11_05_H	Input IP's Func(input5) “dq-axis current conversion”

25.3.2.29 EMU3n H/W Arithmetic Block IDLE Time Startup Command A1 Register (EMU3nFUNCIDLEGRPA1)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0092_H

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SFTEN	—	—	—	IP	—	—	—	—	—	—	—	FUNC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Table 25.51 EMU3nFUNCIDLEGRPA1 Register Contents

Bit Position	Bit	Function
15 to 13	—	These bits are read as 0. The write value should be 0.
12	SFTEN	Software startup enable bit This bit enables software startup of the PI control IP at a write access timing. If software startup is disabled, the PI control IP is started according to a startup source of the PI control IP. 0: Software startup disabled. 1: Software startup enabled. **CAUTION:** Any startup request issued when the IP bit of the EMU3nFUNCIDLEGRPA1 register is set to 0 is invalid.
11 to 9	—	These bits are read as 0. The write value should be 0.
8	IP	Startup IP specification enable bit This bit enables or disables the specification of starting H/W arithmetic block to be used when the PI control IP is started. 0: Disable. 1: Enable.
7 to 1	—	These bits are read as 0. The write value should be 0.
0	FUNC	Startup H/W arithmetic block select bit These bits select the H/W arithmetic block to be initiated when starting the PI control IP.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

- When not software-starting the PI control IP but starting it on a PI control IP startup source

Register Value	Corresponding IP and H/W Arithmetic Block
01_00_H	PI control IP Func(pi0) “No Operation”
01_01_H	PI control IP Func(pi1) “pi control”

- When software-starting the PI control IP at timing of EMU3nFUNCIDLEGRPA1 register write access

Register Value	Corresponding IP and H/W Arithmetic Block
11_00_H	PI control IP Func(pi0) “No Operation”
11_01_H	PI control IP Func(pi1) “pi control”

25.3.2.30 EMU3n H/W Arithmetic Block IDLE Time Startup Command A2 Register (EMU3nFUNCIDLEGRPA2)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0094_H

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SFTEN	—	—	—	IP	—	—	—	—	FUNC[2:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.52 EMU3nFUNCIDLEGRPA2 Register Contents

Bit Position	Bit	Function
15 to 13	—	These bits are read as 0. The write value should be 0.
12	SFTEN	Software startup enable bit This bit enables software startup of the PWM IP at a write access timing. If software startup is disabled, the PWM IP is started according to a startup source of the PWM IP. 0: Software startup disabled. 1: Software startup enabled. **CAUTION:** Any startup request issued when the IP bit of the EMU3nFUNCIDLEGRPA2 register is set to 0 is invalid.
11 to 9	—	These bits are read as 0. The write value should be 0.
8	IP	Startup IP specification enable bit This bit enables or disables the specification of starting H/W arithmetic block to be used when the PWM IP is started. 0: Disable. 1: Enable.
7 to 4	—	These bits are read as 0. The write value should be 0.
3 to 0	FUNC[2:0]	Startup H/W arithmetic block select bit These bits select the H/W arithmetic block to be initiated when starting the PWM IP.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

- When not software-starting the PWM IP but starting it on a PWM IP startup source

Register Value	Corresponding IP and H/W arithmetic block
01_00_H	PWM IP's Func(pwm0) “No Operation”
01_01_H	PWM IP's Func(pwm1) “dq-axis voltage correction (non-interference control)”
01_02_H	PWM IP's Func(pwm2) “Electrical angle offset processing”
01_03_H	PWM IP's Func(pwm3) “2-phase-to-3-phase conversion”
01_04_H	PWM IP's Func(pwm4) “PWM modulation”
01_05_H	PWM IP's Func(pwm5) “Duty factor calculation”
01_06_H	PWM IP's Func(pwm6) “Offset addition”
01_08_H	PWM IP's Func(pwm8) “Output voltage selection and limit processing”
01_09_H	PWM IP's Func(pwm9) “PWM value calculation”
01_0A_H	PWM IP's Func(pwm10) “Dead time compensation”
01_0B_H	PWM IP's Func(pwm11) “PWM value limit processing”

- When software-starting the PWM IP at timing of EMU3nFUNCIDLEGRPA2 register write access

Register Value	Corresponding IP and H/W Arithmetic Block
11_00_H	PWM IP's Func(pwm0) “No Operation”
11_01_H	PWM IP's Func(pwm1) “dq-axis voltage correction (non-interference control)”
11_02_H	PWM IP's Func(pwm2) “Electrical angle offset processing”
11_03_H	PWM IP's Func(pwm3) “2-phase-to-3-phase conversion”
11_04_H	PWM IP's Func(pwm4) “PWM modulation”
11_05_H	PWM IP's Func(pwm5) “Duty factor calculation”
11_06_H	PWM IP's Func(pwm6) “Offset addition”
11_08_H	PWM IP's Func(pwm8) “Output voltage selection and limit processing”
11_09_H	PWM IP's Func(pwm9) “PWM value calculation”
11_0A_H	PWM IP's Func(pwm10) “Dead time compensation”
11_0B_H	PWM IP's Func(pwm11) “PWM value limit processing”

25.3.2.31 EMU3n H/W Arithmetic Block Completion Determination A Register (EMU3nFUNCFINGRPA)

Access: Readable in 16-bit units.

Address: <EMU3n_base> + 0096_H

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IP[1:0]		—	—	—	—	FUNC[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.53 EMU3nFUNCFINGRPA Register Contents

Bit Position	Bit	Function
15 to 10	—	These bits are read as 0. The write value should be 0.
9, 8	IP[1:0]	Completed IP determination bits Indicate the latest IP whose state has transited to IDEL or WAIT after completion of an H/W arithmetic block. 00: None 01: Input IP 10: PI control IP 11: PWM IP
7 to 4	—	These bits are read as 0. The write value should be 0.
3 to 0	FUNC[3:0]	Completed H/W arithmetic block determination bits Indicate the latest H/W arithmetic block whose state has transited to IDLE or WAIT after completion of a H/W arithmetic block

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis).**

NOTE

The register value is not updated when transiting to IDLE (input IP complete) with the INPUT5 bits of the EMU3nFUNCFLGRPA0 register indicating the last H/W arithmetic block of the input IP set to 00.

The register value is not updated when transiting to IDLE (PI control IP complete) with the PI1 bits of the EMU3nFUNCFLGRPA1 register indicating the last H/W arithmetic block of the PI control IP set to 00.

The register value is not updated when transiting to IDLE (PWM IP complete) with the PWM11 bits of the EMU3nFUNCFLGRPA2 register indicating the last H/W arithmetic block of the PWM IP set to 00.

Read Value	Corresponding IP and H/W Arithmetic Block
01_00_H	Input IP's Func(input0) “No Operation”
01_01_H	Input IP's Func(input1) “Electrical angle source select”
01_04_H	Input IP's Func(input4) “Motor current value calculation”
01_05_H	Input IP's Func(input5) “dq-axis current conversion”
02_00_H	PI control IP'S Func(pi0) “No Operation”
02_01_H	PI control IP's Func(pi1) “pi control”
03_00_H	PWM IP's Func(pwm0) “No Operation”
03_01_H	PWM IP's Func(pwm1) “dq-axis voltage correction (non-interference control)”
03_02_H	PWM IP's Func(pwm2) “Electrical angle offset processing”
03_03_H	PWM IP's Func(pwm3) “2-phase-to-3-phase conversion”
03_04_H	PWM IP's Func(pwm4) “PWM modulation”
03_05_H	PWM IP's Func(pwm5) “Duty factor calculation”
03_06_H	PWM IP's Func(pwm6) “Offset addition”
03_08_H	PWM IP's Func(pwm8) “Output voltage selection and limit processing”
03_09_H	PWM IP's Func(pwm9) “PWM value calculation”
03_0A_H	PWM IP's Func(pwm10) “Dead time compensation”
03_0B_H	PWM IP's Func(pwm11) “PWM value limit processing”

25.3.2.32 EMU3n H/W Arithmetic Block IDLE Time Startup Command B Register (EMU3nFUNCIDLEGRPB)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0098_H

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	IP	—	—	—	—	FUNC[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.54 EMU3nFUNCIDLEGRPB Register Contents

Bit Position	Bit	Function
15 to 9	—	These bits are read as 0. The write value should be 0.
8	IP	Startup IP specification enable bit Enables or disables the specification of starting H/W arithmetic block when starting the angle generation IP. 0: Disable. 1: Enable.
7 to 4	—	These bits are read as 0. The write value should be 0.
3 to 0	FUNC[3:0]	Startup H/W arithmetic block specification bits These bits specify the H/W arithmetic block to be started when starting the angle generation IP.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

Register Value	Corresponding IP and H/W Arithmetic Block
01_00_H	Angle generation IP Func(ang0) “No Operation”
01_06_H	Angle generation IP Func(ang6) “Offset error convolution”
01_07_H	Angle generation IP Func(ang7) “Electrical angle calculation”
01_08_H	Angle generation IP Func(ang8) “Angle compare match”

25.3.2.33 EMU3n H/W Arithmetic Block Completion Determination B Register (EMU3nFUNCFINGRPB)

Access: Readable in 16-bit units.

Address: <EMU3n_base> + $009A_H$

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	IP	—	—	—	—	FUNC[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.55 EMU3nFUNCFINGRPB Register Contents

Bit Position	Bit	Function
15 to 9	—	These bits are read as 0. The write value should be 0.
8	IP	Completed IP determination bit Indicates the latest IP whose state has transited to IDLE or WAIT after completion of a H/W arithmetic block. 0: None 1: Angle generation IP
7 to 4	—	These bits are read as 0. The write value should be 0.
3 to 0	FUNC[3:0]	Completed H/W arithmetic block determination bits Indicate the latest H/W arithmetic block whose state has transited to IDLE or WAIT after completion of a H/W arithmetic block.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

NOTE

The register value is not updated when transiting to IDLE (angle generation IP complete) with the ANG8 bits of the EMU3nFUNCFLGRPB register indicating the last H/W arithmetic block of the angle generation IP set to 00.

Read Value	Corresponding IP and H/W Arithmetic Block
01_00_H	Angle generation IP Func(ang0) “No Operation”
01_06_H	Angle generation IP Func(ang6) “Offset error convolution”
01_07_H	Angle generation IP Func(ang7) “Electrical angle calculation”
01_08_H	Angle generation IP Func(ang8) “Angle compare match”

25.3.2.34 EMU3n H/W Arithmetic Block WAIT Time Startup Command A Register (EMU3nFUNCWAITGRPA)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $00A4_H$

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IP[1:0]		—	—	—	—	FUNC[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.56 EMU3nFUNCWAITGRPA Register Contents

Bit Position	Bit	Function
15 to 10	—	These bits are read as 0. The write value should be 0.
9, 8	IP[1:0]	WAIT time startup IP specification enable bits Specify the IP of the next H/W arithmetic block to be transited to while the input IP, PI control IP, or PWM IP is in WAIT state. The IP transits its state from WAIT and runs the specified H/W arithmetic block upon receipt of a register write access. 00: Disable 01: Input IP 10: PI control IP 11: PWM IP
7 to 4	—	These bits are read as 0. The write value should be 0.
3 to 0	FUNC[3:0]	WAIT time startup H/W arithmetic block specification bits Specify the next state to be transit, H/W arithmetic block or an IP completion (transition to IDLE), while the input IP, PI control IP, or PWM IP is in WAIT state.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

Register Value	Corresponding IP and Arithmetic Block
01_00_H	Input IP's Func(input0) “No Operation”
01_01_H	Input IP's Func(input1) “Electrical angle source select”
01_04_H	Input IP's Func(input4) “Motor current value calculation”
01_05_H	Input IP's Func(input5) “dq-axis current conversion”
01_07_H	Input IP complete
02_00_H	PI control IP's Func(pi0) “No Operation”
02_01_H	PI control IP's Func(pi1) “pi control”
02_02_H	PI control IP complete
03_00_H	PWM IP's Func(pwm0) “No Operation”
03_01_H	PWM IP's Func(pwm1) “dq-axis voltage correction (non-interference control)”
03_02_H	PWM IP's Func(pwm2) “Electrical angle offset processing”
03_03_H	PWM IP's Func(pwm3) “2-phase-to-3-phase conversion”
03_04_H	PWM IP's Func(pwm4) “PWM modulation”
03_05_H	PWM IP's Func(pwm5) “Duty factor calculation”
03_06_H	PWM IP's Func(pwm6) “Offset addition”
03_08_H	PWM IP's Func(pwm8) “Output voltage selection and limit processing”
03_09_H	PWM IP's Func(pwm9) “PWM value calculation”
03_0A_H	PWM IP's Func(pwm10) “Dead time compensation”
03_0B_H	PWM IP's Func(pwm11) “PWM value limit processing”
03_0C_H	PWM IP complete

25.3.2.35 EMU3n H/W Arithmetic Block WAIT Time Startup Command B Register (EMU3nFUNCWAITGRPB)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $00A8_H$

Value after reset: 0000_H

Category: General/common

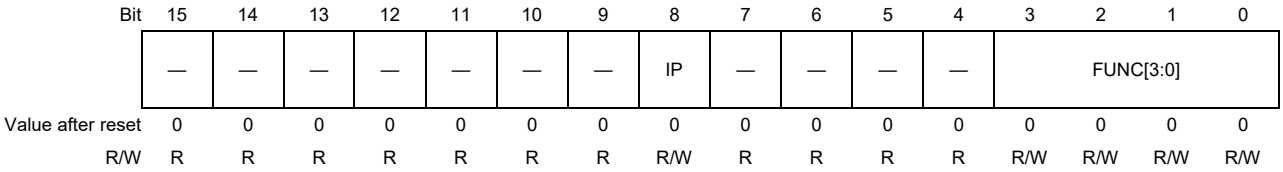


Table 25.57 EMU3nFUNCWAITGRPB Register Contents

Bit Position	Bit	Function
15 to 9	—	These bits are read as 0. The write value should be 0.
8	IP	WAIT time startup IP specification enable bits Specifies the next H/W arithmetic block to be transited to while the angle generation IP is in WAIT state. The IP transits its state from WAIT and runs the specified H/W arithmetic block upon receipt of a register write access. 0: Disable 1: Angle generation IP
7 to 4	—	These bits are read as 0. The write value should be 0.
3 to 0	FUNC[3:0]	WAIT time startup H/W arithmetic block specification bits Specify the next state to be transit, H/W arithmetic block or angle generation IP completion (transition to IDLE), while the angle generation IP is in WAIT state.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

Register Value	Corresponding IP and Arithmetic Block
01_00_H	Angle generation IP Func(ang0) “No Operation”
01_06_H	Angle generation IP Func(ang6) “Offset error convolution”
01_07_H	Angle generation IP Func(ang7) “Electrical angle calculation”
01_08_H	Angle generation IP Func(ang8) “Angle compare match”
01_09_H	Angle generation IP complete

25.3.2.36 EMU3n Functional IP State Determination A Register (EMU3nFSMSTGRPA)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $00D4_H$

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IP[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WAIT	—	—	—	—	—	—	—	BUSY
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.58 EMU3nFSMSTGRPA Register Contents

Bit Position	Bit	Function
31 to 18	—	These bits are read as 0. The write value should be 0.
17, 16	IP[1:0]	Active IP determination bits Reading this bit indicates the function IP that is running. 00: All of the input IP, PI control IP, and PWM IP are in IDLE state. 01: The input IP is running (not in idle state). 10: The PI control IP is running (not in idle state). 11: The PWM IP is running (not in idle state).
15 to 9	—	These bits are read as 0. The write value should be 0.
8	WAIT	Wait state determination bit Reading this bit indicates the wait state of the function IP designated by the IP bits. 0: The function IP is not in wait state (idle or busy state). 1: The function IP is in wait state.
7 to 1	—	These bits are read as 0. The write value should be 0.
0	BUSY	Busy state determination bit Reading this bit indicates the busy state of the function IP designated by the IP bits. 0: The function IP is not in busy state (idle or wait state). 1: The function IP is in busy state.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

25.3.2.37 EMU3n Functional IP State Determination B Register (EMU3nFSMSTGRPB)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $00D8_H$

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WAIT	—	—	—	—	—	—	—	BUSY
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.59 EMU3nFSMSTGRPB Register Contents

Bit Position	Bit	Function
31 to 17	—	These bits are read as 0. The write value should be 0.
16	IP	Active IP determination bits Reading this bit indicates the function IP that is running. 0: The angle generation IP is in idle state. 1: The angle generation IP is running (not in idle state).
15 to 9	—	These bits are read as 0. The write value should be 0.
8	WAIT	Wait state determination bit Reading this bit indicates the wait state of the function IP designated by the IP bits. 0: The function IP is not in wait state (idle or busy state). 1: The function IP is in wait state.
7 to 1	—	These bits are read as 0. The write value should be 0.
0	BUSY	Busy state determination bit Reading this bit indicates the busy state of the function IP designated by the IP bits. 0: The function IP is not in busy state (idle or wait state). 1: The function IP is in busy state.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

25.3.2.38 EMU3n H/W Arithmetic Block Post-Completion Transition Control A0 Register (EMU3nFUNCFLGRPA0)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $00E4_H$

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—		—		INPUT5[1:0]		INPUT4[1:0]		—		—		INPUT1[1:0]		INPUT0[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.60 EMU3nFUNCFLGRPA0 Register Contents

Bit Position	Bit	Function
15 to 12, 7 to 4	—	These bits are read as 0. The write value should be 0.
11, 10, 9, 8, 3, 2, 1, 0	INPUTn[1:0]	Input IP arithmetic block post completion transition control bits 00: Transit to the next block for execution upon completion of the execution of the H/W arithmetic block. 01: Transit to the idle state upon completion of the execution of the H/W arithmetic block. 10: Transit to the wait state upon completion of the execution of the H/W arithmetic block. 11: Inhibited **CAUTION:** The execution sequence is input 0→1→4→5.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

25.3.2.39 EMU3n H/W Arithmetic Block Post-Completion Transition Control A1 Register (EMU3nFUNCFLGRPA1)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $00E8_H$

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	PI1[1:0]		PI0[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.61 EMU3nFUNCFLGRPA1 Register Contents

Bit Position	Bit	Function
7 to 4	—	These bits are read as 0. The write value should be 0.
3 to 0	PIn[1:0]	PI control IP arithmetic block post completion transition control bits 00: Transit to the next block for execution upon completion of the execution of the H/W arithmetic block. 01: Transit to the idle state upon completion of the execution of the H/W arithmetic block. 10: Transit to the wait state upon completion of the execution of the H/W arithmetic block. 11: Inhibited **CAUTION:** The execution sequence is PI0→PI1.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

25.3.2.40 EMU3n H/W Arithmetic Block Post-Completion Transition Control A2 Register (EMU3nFUNCFLGRPA2)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + $00EC_H$

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	PWM11[1:0]		PWM10[1:0]		PWM9[1:0]		PWM8[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—		PWM6[1:0]		PWM5[1:0]		PWM4[1:0]		PWM3[1:0]		PWM2[1:0]		PWM1[1:0]		PWM0[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.62 EMU3nFUNCFLGRPA2 Register Contents

Bit Position	Bit	Function
31 to 24, 15, 14	—	These bits are read as 0. The write value should be 0.
23, 22, 21, 20, 19, 18, 17, 16, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0	PWMn[1:0]	PWM IP arithmetic block post completion transition control bits 00: Transit to the next block for execution upon completion of the execution of the H/W arithmetic block. 01: Transit to the idle state upon completion of the execution of the H/W arithmetic block. 10: Transit to the wait state upon completion of the execution of the H/W arithmetic block. 11: Inhibited **CAUTION:** The execution sequence is PWM0→PWM1→PWM2→PWM3→ … →PWM11.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)**.

25.3.2.41 EMU3n H/W Arithmetic Block Post-Completion Transition Control B Register (EMU3nFUNCFLGRPB)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + $00F0_H$

Value after reset: $0000\ 0000_H$

Category: General/common

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ANG8[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ANG7[1:0]		ANG6[1:0]		—		—		—		—		—		ANG0[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 25.63 EMU3nFUNCFLGRPB Register Contents

Bit Position	Bit	Function
31 to 18, 11 to 2	—	These bits are read as 0. The write value should be 0.
17, 16, 15, 14, 13, 12, 1, 0	ANGn[1:0]	Angle generation IP arithmetic block post completion transition control bits 00: Transit to the next block for execution upon completion of the execution of the H/W arithmetic block. 01: Transit to the idle state upon completion of the execution of the H/W arithmetic block. 10: Transit to the wait state upon completion of the execution of the H/W arithmetic block. 11: Inhibited **CAUTION:** The execution sequence is ANG0→ … →ANG8.

Note: For details, see **Section 25.5.3, Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis).**

25.3.2.42 EMU3n Angle Generation IP Control Register (EMU3nANGCTR)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0180_H

Value after reset: 00_H

Category: Angle generation IP

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ANGSEL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.64 EMU3nANGCTR Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	ANGSEL	Angle information select bit 0: Uses the resolver angle from the R/D converter. 1: Uses the value of the EMU3nRESTHSFT register.

Note: For details, see **Section 25.4.2, Angle Generation IP**. RDC3A1 is only available for the C1M-A2.

25.3.2.43 EMU3n Compare Judgment Correction Register 0 (EMU3nCPJUD0)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0182_H

Value after reset: FF_H

Category: Angle generation IP

Bit	7	6	5	4	3	2	1	0
	DATA (unsigned)							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.65 EMU3nCPJUD0 Register Contents

Bit Position	Bit	Function
7 to 0	DATA	Data bits (unsigned integer) Store the correction value to be used to detect angle transitions.

Note: For details, see **Section 25.4.2(3), Func(ang8) angle compare match**.

This register must be rewritten during the initialization which is to be started before starting motor control. It is inhibited to rewrite this register after motor control is started.

25.3.2.44 EMU3n Compare Judgment Correction Register 1 (EMU3nCPJUD1)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0183_H

Value after reset: FF_H

Category: Angle generation IP

Bit	7	6	5	4	3	2	1	0
	DATA (unsigned)							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.66 EMU3nCPJUD1 Register Contents

Bit Position	Bit	Function
7 to 0	DATA	Data bits (unsigned integer) Store the correction value to be used to detect resolver angle transitions.

Note: For details, see **Section 25.4.2(3), Func(ang8) angle compare match**.

This register must be rewritten during the initialization which is to be started before starting motor control. It is inhibited to rewrite this register after motor control is started.

25.3.2.45 EMU3n Resolver Angle Software Input Register (EMU3nRESTHSFT)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0184_H

Value after reset: 0000_H

Category: Angle generation IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.67 EMU3nRESTHSFT Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Store the software set resolver angle that is applied to the angle generation IP.

Note: For details, see **Section 25.4.2(1) Func(ang6) Convolve offset error**.

25.3.2.46 EMU3n Resolver Angle Offset Value Register (EMU3nANGOFS)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0186_H

Value after reset: 0000_H

Category: Angle generation IP

	Sign part															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.68 EMU3nANGOFS Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15-11: Sign part) Store the offset value for the resolver angle that is applied to the angle generation IP.

Note: For details, see **Section 25.4.2(1) Func(ang6) Convolve offset error**.

25.3.2.47 EMU3n Electrical Angle Generation Coefficient Register (EMU3nPXR)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0188_H

Value after reset: 0100_H

Category: Angle generation IP

	Integer part								Fraction part							
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.69 EMU3nPXR Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed fixed-point numbers) (b15: Sign part) Store the multiplier coefficient value to be used for electrical angle generation. **CAUTION:** b15 must always be set to 0.

Note: For details, see **Section 25.4.2(2), Func(ang7) Electrical angle**.

This register is used to store fixed-point numbers consisting of an 8-bit integer part and an 8-bit fractional part. After a multiplication is performed using the value of this register as the multiplication coefficient, the results of the multiplication are shifted 8 bits to the right.

25.3.2.48 EMU3n Resolver Angle Register (EMU3nRESTHETA)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $018A_H$

Value after reset: 0000_H

Category: Angle generation IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.70 EMU3nRESTHETA Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Store the value of the resolver angle that is applied to the angle generation IP.

Note: For details, see **Section 25.4.2(1) Func(ang6) Convolve offset error, Section 25.4.2(2) Func(ang7) Electrical angle calculation, Section 25.4.2(3) Func(ang8) angle compare match**.

25.3.2.49 EMU3n Electrical Angle Register (EMU3nTHTEFIX)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $018C_H$

Value after reset: 0000_H

Category: Angle generation IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.71 EMU3nTHTEFIX Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Store the value of the electrical angle that is generated by the angle generation IP.

Note: For details, see **Section 25.4.2(1) Func(ang6) Convolve offset error, Section 25.4.2(2) Func(ang7) Electrical angle, Section 25.4.2(3) Func(ang8) angle compare match**.

25.3.2.50 EMU3n Resolver Angle Poles Value Register (EMU3nRESRLD)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $018E_H$

Value after reset: 00_H

Category: Angle generation IP

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DATA (unsigned)		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.72 EMU3nRESRLD Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	DATA	Data bits (unsigned integer) These bits must be set with the number of resolver angle poles minus 1.

Note: For details, see **Section 25.4.2(2)**, **Func(ang7) Electrical angle**.

25.3.2.51 EMU3n Resolver Angle Period Count Value Register (EMU3nRESCNT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $018F_H$

Value after reset: 00_H

Category: Angle generation IP

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DATA (unsigned)		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.73 EMU3nRESCNT Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	DATA	Data bits (unsigned integer) Count value that increments or decrements as the position of the resolver angle pole transits. Writing a value in these bit positions updates the count value.

Note: For details, see **Section 25.4.2(2), Func(ang7) Electrical angle**.

25.3.2.52 EMU3n Post Error Convolution Resolver Angle Register (EMU3nTHTRESFIX)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0190_H

Value after reset: 0000_H

Category: Angle generation IP,

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.74 EMU3nTHTRESFIX Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer)

Note: For details, see **Section 25.4.2(1) Func(ang6) Convolve offset error**.

25.3.2.53 EMU3n Input IP Control Register (EMU3nCTRINMD)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $01C0_H$

Value after reset: 0000_H

Category: Input IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	INSTCTR[1:0]		CMUVW[1:0]			CMES	KCL	FREGIN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.75 EMU3nCTRINMD Register Contents

Bit Position	Bit	Function
15 to 8	—	These bits are read as 0. The write value should be 0.
7, 6	INSTCTR[1:0]	A/D conversion completion timing (input IP startup timing) select bits*1 00: All A/D conversions within A/D channel group are completed. (A/D converter scan group 4 scan ended) 01: A/D channel 0 A/D conversion completed (A/D converter virtual channel 0 conversion completed) 10: A/D channel 1 A/D conversion completed (A/D converter virtual channel 1 conversion completed) 11: A/D channel 2 A/D conversion completed (A/D converter virtual channel 2 conversion completed)
5 to 3	CMUVW[1:0]	These bits select the target of current value measurement when the CMES bit is set to 1. 000: 3 phases (U phase, V phase, and W phase) are measured. 001: 2 phases (V phase and W phase) are measured. 010: 2 phases (U phase and W phase) are measured. 100: 2 phases (U phase and V phase) are measured. Other than above are inhibited.
2	CMES	Selects the target of current measurement. 0: 2 phases (V phase and W phase) are measured. 1: Selected by the CMUVW bits.
1	KCL	Kirchhoff's current law violation detection processing setting bit 0: Disables Kirchhoff's current law violation detection processing. 1: Enables Kirchhoff's current law violation detection processing.
0	FREGIN	Angle data select bit This bit selects the electrical angle to be used by the input IP. 0: Uses the value of the EMU3nTHTESFT register. 1: Uses the electrical input buffer (EMU3nTHTEIBUF). (The above-mentioned buffer is the one that is loaded with the electrical angle (EMU3nTHTE register) generated by the angle generation IP when the input IP is started.

Note 1. The bits INSTCTR[1:0] must be changed only when the A/D converter is stopped.

This register is used to control the input IP. For control details of the bits, see **Section 25.4.3, Input IP**.

CAUTION

The connection relationship between the A/D converter's virtual channel registers (VCRk (k = 0, 1, 2)) and data registers (DRk (k = 0, 1, 2)) and the EMU3's A/D data channels (k = 0, 1, 2) is summarized below. Set up the registers in the A/D converter so that the A/D converter's virtual channels 0-2 can output the following current values that the EMU3 expects.

A/D Converter Side	EMU3 Side	Current Value that the EMU3 Expects
Virtual channel 0 (VCR0, DR0)	A/D data channel 0	V-phase current value
Virtual channel 1 (VCR1, DR1)	A/D data channel 1	W-phase current value
Virtual channel 2 (VCR2, DR2)	A/D data channel 2	U-phase current value

25.3.2.54 EMU3n Resolver Angle Monitor Register (EMU3nTHTRESFIXIN)
EMU3n Electrical Angle Retention Register (EMU3nTHTE)
EMU3n Input IP Electrical Angle Software Input Register (EMU3nTHTESFT)
EMU3n Electrical Angle Response Delay Correction Variable Register (EMU3nEARD)
EMU3n Electrical Angle Input Buffer Register (EMU3nTHTEIBUF)
EMU3n Input IP Post Correction Electrical Angle Register (EMU3nTHTESEL)

Access: EMU3nTHTRESFIXIN: Readable in 16-bit units.
EMU3nTHTE: Readable in 16-bit units.
EMU3nTHTESFT: Readable/writable in 16-bit units.
EMU3nEARD: Readable/writable in 16-bit units.
EMU3nTHTEIBUF: Readable/writable in 16-bit units.
EMU3nTHTESEL: Readable/writable in 16-bit units.

Address: EMU3nTHTRESFIXIN: <EMU3n_base> + $01C4_H$
EMU3nTHTE: <EMU3n_base> + $01C6_H$
EMU3nTHTESFT: <EMU3n_base> + $01C8_H$
EMU3nEARD: <EMU3n_base> + $01CA_H$
EMU3nTHTEIBUF: <EMU3n_base> + $01CC_H$
EMU3nTHTESEL: <EMU3n_base> + $01CE_H$

Value after reset: 0000_H

Category: Input IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	*	*	*	*	*	*	*	*	*	*	*	*

Table 25.76 EMU3nTHTRESFIXIN, EMU3nTHTE, EMU3nTHTESFT, EMU3nEARD, EMU3nTHTEIBUF, EMU3nTHTESEL Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer)

Note: For details, see **Section 25.4.3(1), Func(input1) Electrical angle source**.

25.3.2.55 EMU3n A/D Data k Register (EMU3nADk) (k = 0, 1, 2)

Access: Readable/writable in 16-bit units.

Address: EMU3nAD0: <EMU3n_base> + $01D0_H$

EMU3nAD1: <EMU3n_base> + $01D4_H$

EMU3nAD2: <EMU3n_base> + $01D8_H$

Value after reset: 0000_H

Category: Input IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.77 EMU3nADk Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Loaded with the A/D conversion value from the A/D converter upon completion of A/D conversion. The A/D conversion value can be changed by writing this register with a new value.

Note: For details, see Section **25.4.3(2), Func(input4) Motor** current value calculation.

This register is loaded with the results of A/D conversion at the timing that is specified by the INSTCTR bit of the EMU3nCTRINMD register. The conversion value can also be rewritten from a CPU program. If the loading of the A/D conversion results and the writing by a CPU program occur at the same time, the loading of the A/D conversion results takes precedence.

25.3.2.56 EMU3n A/D Data k Input Buffer Register (EMU3nADkIBUF) (k = 0, 1, 2)

Access: Readable/writable in 16-bit units.

Address: EMU3nAD0IBUF: <EMU3n_base> + $01D2_H$

EMU3nAD1IBUF: <EMU3n_base> + $01D6_H$

EMU3nAD2IBUF: <EMU3n_base> + $01DA_H$

Value after reset: 0000_H

Category: Input IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.78 EMU3nADkIBUF Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Loaded with the A/D data or the data that is subjected to IIR filter processing according to the settings of the EMU3nADBUFSEL register when the input IP is started. The data is used for arithmetic processing.

Note: For details, see **Section 25.4.3(2), Func(input4) Motor current value calculation**.

25.3.2.57 EMU3n A/D Data k Conversion Value Register (EMU3nADkFIX) (k = 0, 1, 2)

Access: Readable/writable in 16-bit units.

Address: EMU3nAD0FIX: <EMU3n_base> + $01DC_H$

EMU3nAD1FIX: <EMU3n_base> + $01E0_H$

EMU3nAD2FIX: <EMU3n_base> + $01E4_H$

Value after reset: 0000_H

Category: Input IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.79 EMU3nADkFIX Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Loaded with the A/D conversion value to be used for arithmetic processing.

Note: For details, see **Section 25.4.3(2), Func(input4) Motor current value calculation**.

25.3.2.58 EMU3n A/D Data k Origin Correction Value Register (EMU3nADkOFS) (k = 0, 1, 2)

Access: Readable/writable in 16-bit units.

Address: EMU3nAD0OFS: <EMU3n_base> + $01DE_H$

EMU3nAD1OFS: <EMU3n_base> + $01E2_H$

EMU3nAD2OFS: <EMU3n_base> + $01E6_H$

Value after reset: 0800_H

Category: Input IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.80 EMU3nADkOFS Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Set the origin correction value that is used to convert the sign of the A/D conversion value stored in the EMU3nADkFIX register from unsigned to signed.

Note: For details, see **Section 25.4.3(2), Func(input4) Motor current value calculation**.

The signed integer A/D data is calculated by subtracting the value of this register from the value of the EMU3nADkFIX register.

(Signed integer value) = EMU3nADkFIX - EMU3nADkOFS

25.3.2.59 EMU3n dq-Axis Current Transformation Coefficient Register (EMU3nSR2)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + $01E8_H$

Value after reset: $0000\ D106_H$

Category: Input IP

Integer part

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Fraction part

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	1	1	0	1	0	0	0	1	0	0	0	0	0	1	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.81 EMU3nSR2 Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed fixed-point numbers) (b31: Sign part) Set the coefficient to be used to convert 3-phase current values to dq-axis current values. The value established after a reset is $\sqrt{(2/3)}$.

Note: For details, see **Section 25.4.3(3), Func(input5) dq-axis current** .

This register is used to store fixed-point numbers consisting of a 16-bit integer part and a 16-bit fractional part. After a multiplication is performed using the value of this register as the multiplication coefficient, the results of the multiplication are shifted 16 bits to the right.

This register is set a value of $0000\ D106_H$ after a reset. This is equivalent to the following calculation results:

$\sqrt{(2/3)} \times 65536 = 0000\ D106_H$ (rounded off the fractional part)

25.3.2.60 EMU3n LSB Adjustment Register (EMU3nDIVLSB)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + $01EC_H$

Value after reset: $0001\ 0000_H$

Category: Input IP

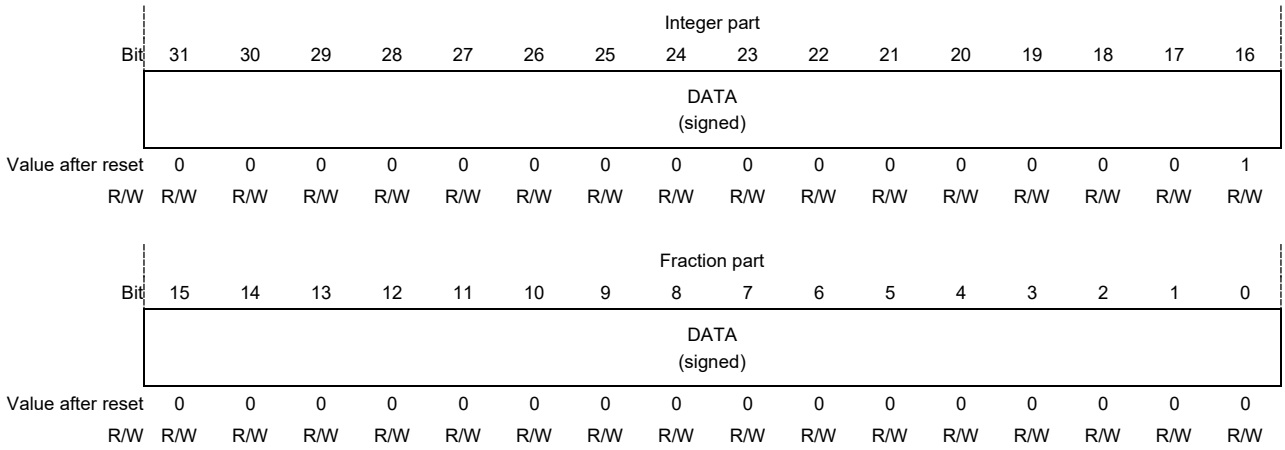


Table 25.82 EMU3nDIVLSB Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed fixed-point numbers) (b31: Sign part) Set the multiplication coefficient that is applied to each phase of the 3-phase current values.

Note: For details, see Section **25.4.3(2), Func(input4) Motor current value calculation**.

This register is used to store fixed-point numbers consisting of a 16-bit integer part and a 16-bit fractional part. After a multiplication is performed using the value of this register as the multiplication coefficient, the results of the multiplication are shifted 16 bits to the right.

25.3.2.61 EMU3n U-Phase Current Value Register (EMU3nIUFIX)
EMU3n V-Phase Current Value Register (EMU3nIVFIX)
EMU3n W-Phase Current Value Register (EMU3nIWFIX)
EMU3n d-Axis Current Value Register (EMU3nIDFIX)
EMU3n q-Axis Current Value Register (EMU3nIQFIX)

Access: Readable/writable in 32-bit units.

Address: EMU3nIUFIX: <EMU3n_base> + $01F0_H$

EMU3nIVFIX: <EMU3n_base> + $01F4_H$

EMU3nIWFIX: <EMU3n_base> + $01F8_H$

EMU3nIDFIX: <EMU3n_base> + $01FC_H$

EMU3nIQFIX: <EMU3n_base> + 0200_H

Value after reset: $0000\ 0000_H$

Category: Input IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.83 EMU3nIUFIX, EMU3nIVFIX, EMU3nIWFIX, EMU3nIDFIX, EMU3nIQFIX Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer) (b31: Sign part)

Note: For details, see **Section 25.4.3, Input IP**.

25.3.2.62 EMU3n U-Phase Current Value Output Buffer Register (EMU3nIUFIXOBUF)
EMU3n V-Phase Current Value Output Buffer Register (EMU3nIVFIXOBUF)
EMU3n W-Phase Current Value Output Buffer Register (EMU3nIWFIXOBUF)
EMU3n d-Axis Current Value Output Buffer Register (EMU3nIDFIXOBUF)
EMU3n q-Axis Current Value Output Buffer Register (EMU3nIQFIXOBUF)

Access: EMU3nIUFIXOBUF: Readable in 32-bit units.
EMU3nIVFIXOBUF: Readable in 32-bit units.
EMU3nIWFIXOBUF: Readable in 32-bit units.
EMU3nIDFIXOBUF: Readable/writable in 32-bit units.
EMU3nIQFIXOBUF: Readable/writable in 32-bit units.

Address: EMU3nIUFIXOBUF: <EMU3n_base> + 0204_H
EMU3nIVFIXOBUF: <EMU3n_base> + 0208_H
EMU3nIWFIXOBUF: <EMU3n_base> + $020C_H$
EMU3nIDFIXOBUF: <EMU3n_base> + 0210_H
EMU3nIQFIXOBUF: <EMU3n_base> + 0214_H

Value after reset: $0000\ 0000_H$

Category: Input IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Table 25.84 EMU3nIUFIXOBUF, EMU3nIVFIXOBUF, EMU3nIWFIXOBUF, EMU3nIDFIXOBUF, EMU3nIQFIXOBUF Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer) (b31: Sign part)

Note: For details, see **Section 25.4.3, Input IP**.

25.3.2.63 EMU3n Kirchhoff's Current Law Threshold Value Register (EMU3nKCLJUD)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0218_H

Value after reset: $0000\ 0000_H$

Category: Input IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	DATA (unsigned)														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.85 EMU3nKCLJUD Register Contents

Bit Position	Bit	Function
31	—	This bit is read as 0. The write value should be 0.
30 to 0	DATA	Data bits (unsigned integer) Set the threshold value that is used by the input IP to determine the detection of Kirchhoff's current law violations.

Note: For details, see Section **25.4.3(2), Func(input4) Motor current value calculation**.

25.3.2.64 EMU3n A/D Data Input Buffer Select Register (EMU3nADBUFSEL)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $021C_H$

Value after reset: 00_H

Category: Input IP

Bit	7	6	5	4	3	2	1	0
	—	—	AD2 BUFSEL[1:0]		AD1 BUFSEL[1:0]		AD0 BUFSEL[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.86 EMU3nADBUFSEL Register Contents

Bit Position	Bit	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5, 4	AD2BUFSEL[1:0]	A/D data 2-input buffer data select bits Select the data to be loaded in the EMU3nAD2IBUF register when the input IP is started. 00: A/D2 data 01: Data established after IIR filter channel 0 processing 10: Data established after IIR filter channel 1 processing 11: Data established after IIR filter channel 2 processing
3, 2	AD1BUFSEL[1:0]	A/D data 1-input buffer data select bits Select the data to be loaded in the EMU3nAD1IBUF register when the input IP is started. 00: A/D1 data 01: Data established after IIR filter channel 0 processing 10: Data established after IIR filter channel 1 processing 11: Data established after IIR filter channel 2 processing
1, 0	AD0BUFSEL[1:0]	A/D data 0-input buffer data select bits Select the data to be loaded in the EMU3nAD0IBUF register when the input IP is started. 00: A/D0 data 01: Data established after IIR filter channel 0 processing 10: Data established after IIR filter channel 1 processing 11: Data established after IIR filter channel 2 processing

25.3.2.65 EMU3n A/D Data k Converted Value Output Buffer Register (EMU3nADkFIXOBUF) (k = 0 to 2)

Access: Readable in 16-bit units.

Address: EMU3nAD0FIXOBUF : <EMU3n_base> + 0220_H

EMU3nAD1FIXOBUF : <EMU3n_base> + 0222_H

EMU3nAD2FIXOBUF : <EMU3n_base> + 0224_H

Value after reset: 0000_H

Category: Input IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.87 EMU3nADkFIXOBUF Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Signed part) Loaded with the value of the EMU3nADkFIX register on completion of input IP processing.

25.3.2.66 EMU3n PI Control IP Control Register (EMU3nPICTR)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0260_H

Value after reset: 00_H

Category: PI control IP

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FSUMIQ	FSUMID
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 25.88 EMU3nPICTR Register Contents

Bit Position	Bit	Function
7 to 2	—	These bits are read as 0. The write value should be 0.
1	FSUMIQ	q-axis integrated value select bit 0: Uses the value of the EMU3nSUMIQ register. 1: Uses the value calculated by the PI control IP.
0	FSUMID	d-axis integrated value select bit 0: Uses the value of the EMU3nSUMID register. 1: Uses the value calculated by the PI control IP

This register controls the PI control IP. For details on the control by each bit, see **Section 25.4.4, PI Control IP**.

25.3.2.67 EMU3n d-Axis Directive Current Value Register (EMU3nIDIN)
EMU3n q-Axis Directive Current Value Register (EMU3nIQIN)
EMU3n d-Axis Current Value Software Input Register (EMU3nID)
EMU3n q-Axis Current Value Software Input Register (EMU3nIQ)
EMU3n d-Axis Integrated Value Software Input Register (EMU3nSUMID)
EMU3n q-Axis Integrated Value Software Input Register (EMU3nSUMIQ)
EMU3n d-Axis Integrated Value Monitor Register (EMU3nSUMIDM)
EMU3n q-Axis Integrated Value Monitor Register (EMU3nSUMIQM)
EMU3n d-Axis Voltage Value Register (EMU3nVD)
EMU3n q-Axis Voltage Value Register (EMU3nVQ)
EMU3n d-Axis Voltage Value Output Buffer Register (EMU3nVDOBUF)
EMU3n q-Axis Voltage Value Output Buffer Register (EMU3nVQOBUF)

Access: EMU3nIDIN: Readable/writable in 32-bit units.

EMU3nIQIN: Readable/writable in 32-bit units.

EMU3nID: Readable/writable in 32-bit units.

EMU3nIQ: Readable/writable in 32-bit units.

EMU3nSUMID: Readable/writable in 32-bit units.

EMU3nSUMIQ: Readable/writable in 32-bit units.

EMU3nSUMIDM: Readable in 32-bit units.

EMU3nSUMIQM: Readable in 32-bit units.

EMU3nVD: Readable/writable in 32-bit units.

EMU3nVQ: Readable/writable in 32-bit units.

EMU3nVDOBUF: Readable in 32-bit units.

EMU3nVQOBUF: Readable in 32-bit units.

Address: EMU3nIDIN: <EMU3n_base> + 0268_H

EMU3nIQIN: <EMU3n_base> + $026C_H$

EMU3nID: <EMU3n_base> + 0270_H

EMU3nIQ: <EMU3n_base> + 0274_H

EMU3nSUMID: <EMU3n_base> + 0298_H

EMU3nSUMIQ: <EMU3n_base> + $029C_H$

EMU3nSUMIDM: <EMU3n_base> + $02A0_H$

EMU3nSUMIQM: <EMU3n_base> + $02A4_H$

EMU3nVD: <EMU3n_base> + $02B0_H$

EMU3nVQ: <EMU3n_base> + $02B4_H$

EMU3nVDOBUF: <EMU3n_base> + $02B8_H$

EMU3nVQOBUF: <EMU3n_base> + $02BC_H$

Value after reset: $0000\ 0000_H$

Category: PI control IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Table 25.89 EMU3nIDIN, EMU3nIQIN, EMU3nID, EMU3nIQ, EMU3nSUMID, EMU3nSUMIQ, EMU3nSUMIDM, EMU3nSUMIQM, EMU3nVD, EMU3nVQ, EMU3nVDOBUF, EMU3nVQOBUF Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer) (b31: Sign part)

Note: For details, see **Section 25.4.4, PI Control IP**.

25.3.2.68 EMU3n d-Axis Proportional Gain 0 Register (EMU3nGPD0)
EMU3n q-Axis Proportional Gain 0 Register (EMU3nGPQ0)
EMU3n d-Axis Proportional Gain Register (EMU3nGPD)
EMU3n q-Axis Proportional Gain Register (EMU3nGPQ)
EMU3n d-Axis Integral Gain Register (EMU3nGID)
EMU3n q-Axis Integral Gain Register (EMU3nGIQ)

Access: Readable/writable in 32-bit units.

Address: EMU3nGPD0: <EMU3n_base> + 0278_H

EMU3nGPQ0: <EMU3n_base> + $027C_H$

EMU3nGPD: <EMU3n_base> + 0280_H

EMU3nGPQ: <EMU3n_base> + 0284_H

EMU3nGID: <EMU3n_base> + 0288_H

EMU3nGIQ: <EMU3n_base> + $028C_H$

Value after reset: $0000\ 0000_H$

Category: PI control IP

Integer part

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Fractional part

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.90 EMU3nGPD0, EMU3nGPQ0, EMU3nGPD, EMU3nGPQ, EMU3nGID, EMU3nGIQ Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed fixed-point numbers) (b31: Sign part)

Note: For details, see **Section 25.4.4(1), Func(pi1) pi control**.

This register is used to store fixed-point numbers consisting of a 16-bit integer part and a 16-bit fractional part. After a multiplication is performed using the value of this register as the multiplication coefficient, the results of the multiplication are shifted 16 bits to the right.

25.3.2.69 EMU3n d-Axis Integrated Maximum Value Register (EMU3nGIDMAX)
EMU3n q-Axis Integrated Maximum Value Register (EMU3nGIQMAX)
EMU3n d-Axis Voltage Maximum Value Register (EMU3nVDMAX)
EMU3n q-Axis Voltage Maximum Value Register (EMU3nVQMAX)

Access: Readable/writable in 32-bit units.

Address: EMU3nGIDMAX: <EMU3n_base> + 0290_H

EMU3nGIQMAX: <EMU3n_base> + 0294_H

EMU3nVDMAX: <EMU3n_base> + $02A8_H$

EMU3nVQMAX: <EMU3n_base> + $02AC_H$

Value after reset: $0000\ 0000_H$

Category: PI control IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	DATA (unsigned)														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.91 EMU3nGIDMAX, EMU3nGIQMAX, EMU3nVDMAX, EMU3nVQMAX Register Contents

Bit Position	Bit	Function
31	—	This bits is read as 0. The write value should be 0.
30 to 0	DATA	Data bits (unsigned integer)

Note: For details, see **Section 25.4.4(1), Func(pi1) pi control**.

25.3.2.70 EMU3n PWM IP Control Register (EMU3nPWMCTR)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0300_H

Value after reset: $0001\ 0000_H$

Category: PWM IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	FPWMIBTH	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SETDTO	PWMOP[1:0]		SETDEC	DECIDQ[1:0]		SETVEL	VDQSEL	PWMSEL	SHIPWM	FLININIP[1:0]		SETHARM[2:0]			SETPWM
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.92 EMU3nPWMCTR Register Contents (1/2)

Bit Position	Bit	Function
31 to 21	—	These bits are read as 0. The write value should be 0.
20	FPWMIBTH	PWM IP internal input buffer function setting bit 0: Enables the PWM IP internal input buffer function. 1: Disables the PWM IP internal input buffer function.
19 to 17	—	These bits are read as 0. The write value should be 0.
16	—	These bits are read as 1. The write value should be 1.
15	SETDTO	Dead time compensation processing setting bit 0: Disables dead time compensation processing. 1: Enables dead time compensation processing.
14, 13	PWMOP[1:0]	PWM directive value operation select bits 00: PWM 50% duty + voltage directive value 01: PWM 50% duty – voltage directive value 10: Voltage directive value 11: Inhibited
12	SETDEC	Non-interference control setting bit 0: Disables non-interference control processing. 1: Enables non-interference control processing.
11, 10	DECIDQ[1:0]	Non-interference control dq-axis current value select bits 00: Uses the values of the EMU3nID and EMU3nIQ registers as the dq-axis current values. 01: Uses the values of the EMU3nIDFIXOBUFand EMU3nIQFIXOBUF registers as the dq-axis current values. 10: Uses the values of the EMU3nIDIN and EMU3nIQIN registers as the dq-axis current values. 11: Inhibited
9	SETVEL	Angle velocity select bit 0: Uses the value of the EMU3nVELSFT register. 1: Uses the angle velocity value from the RDC.

Table 25.92 EMU3nPWMCTR Register Contents (2/2)

Bit Position	Bit	Function
8	VDQSEL	dq-axis voltage input select bit 0: Designates the d-axis voltage (Vd) and q-axis voltage (Vq) as the inputs to the PWM IP. 1: Designates the dq-axis voltage (Vdq) and dq-axis phase difference (Vθ) as the inputs to the PWM IP.
7	PWMSEL	PWM arithmetic reference value select bit 0: Generates from the value specified in the EMU3nCARR register (carrier period). 1: Generates from the values of the EMU3nCARR register (carrier period) and EMU3nDTT register (short-circuit prevention time).
6	SHIPWM	Output carrier /PWM compare value shift bit 0: Outputs the carrier period and PWM compare values to the TSG3 without shifts. 1: Outputs the carrier period and PWM compare values that shifted 1 bit to the left (doubling) before outputting to the TSG3.
5, 4	FLININIP[1:0]	Electrical angle select bits 00: Use the value of the EMU3nTHTFORESFT register. 01: Use the value of the EMU3nTHTESEL register that is calculated by the input IP. 10: Use the value that is calculated by the angle generation IP. 11: Inhibited
3 to 1	SETHARM[2:0]	PWM modulation mode setting bits 000: Does not apply modulation. 001: Applies SVM. 010: Applies 2-phase modulation. 011: Applies 2-phase SVM (PWM0). 100: Applies 2-phase SVM (PWM1). 101: Applies 2-phase SVM (PWM2). 110: Applies 2-phase SVM (PWM3). 111: Applies third harmonic wave convolution. (Settings other than VDQSEL = 1 are inhibited.)
0	SETPWM	PWM setting bit Selects the output to the TSG3. 0: Uses the values of the EMU3nPWMU, EMU3nPWMV, and EMU3nPWMW registers. 1: Uses the values of the EMU3nPWMUIP, EMU3nPWMVIP, and EMU3nPWMWIP registers that are calculated by the PWM IP.

This register controls the PWM IP. For details on the control of each bit, see **Section 25.4.5, PWM IP**.

25.3.2.71 EMU3n PWM Data Software Transfer Register (EMU3nPWMDT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0304_H

Value after reset: 00_H

Category: PWM IP

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PWMDT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.93 EMU3nPWMDT Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	PWMDT	PWM data software transfer trigger bit Setting this bit to 1 causes the carrier period and PWM compare values to be transferred to the TSG3. This bit is automatically reset to 0 after being set to 1. It is inhibited to write a 0. This bit is always read as 0.

Note: For details, see **Section 25.4.5(11) Func(pwm_out) PWM value output processing**.

25.3.2.72 EMU3n d-Axis Voltage Correction Value Register (EMU3nVDCRCT) EMU3n q-Axis Voltage Correction Value Register (EMU3nVQCRCT)

Access: Readable/writable in 32-bit units.

Address: EMU3nVDCRCT: <EMU3n_base> + 0308_H

EMU3nVQCRCT: <EMU3n_base> + $030C_H$

Value after reset: $0000\ 0000_H$

Category: PWM IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.94 EMU3nVDCRCT, EMU3nVQCRCT Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer) (b31: Sign part) Set the correction value for the d-axis voltage value that is applied to the PWM IP. Set the correction value for the q-axis voltage value that is applied to the PWM IP.

Note: For details, see **Section 25.4.5(1) Func(pwm1) dq-axis voltage correction (non-interference control)**.

25.3.2.73 EMU3n Angular Velocity Value Register (EMU3nVEL)
EMU3n Angular Velocity Value Software Input Register (EMU3nVELSFT)

Access: EMU3nVEL: Readable in 32-bit units.
EMU3nVELSFT: Readable/writable in 32-bit units.

Address: EMU3nVEL: <EMU3n_base> + 0310_H
EMU3nVELSFT: <EMU3n_base> + 0314_H

Value after reset: $0000\ 0000_H$

Category: PWM IP

	Integer part															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

	Integer part				Fraction part											
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Table 25.95 EMU3nVEL, EMU3nVELSFT Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed fixed-point numbers) (b31: Sign part)

25.3.2.74 EMU3n Non-Interference Control Coefficient Angular Velocity Value Gain Register (EMU3nDECVELG)
EMU3n Non-Interference Control Coefficient Magnetic Flux Value Register (EMU3nDECFLUX)
EMU3n Non-Interference Control Coefficient Ld Value Register (EMU3nDECLD)
EMU3n Non-Interference Control Coefficient Lq Value Register (EMU3nDECLQ)

Access: Readable/writable in 32-bit units.

Address: EMU3nDECVELG: <EMU3n_base> + 0318_H

EMU3nDECFLUX: <EMU3n_base> + $031C_H$

EMU3nDECLD: <EMU3n_base> + 0320_H

EMU3nDECLQ: <EMU3n_base> + 0324_H

Value after reset: $0000\ 0000_H$

Category: PWM IP

Integer part

Bit	b31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	b16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Fraction part

Bit	b15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	b0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.96 EMU3nDECVELG, EMU3nDECFLUX, EMU3nDECLD, EMU3nDECLQ Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed fixed-point numbers) (b31: Sign part)

Note: For details, see **Section 25.4.5(1) Func(pwm1) dq-axis voltage correction (non-interference control)**.

This register is used to store fixed-point numbers consisting of a 16-bit integer part and a 16-bit fractional part. After a multiplication is performed using the value of this register as the multiplication coefficient, the results of the multiplication are shifted 16 bits to the right.

25.3.2.75 EMU3n Non-Interference Control d-Axis Maximum Value Register (EMU3nVD2MAX)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0328_H

Value after reset: $0000\ 0000_H$

Category: PWM IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	DATA (unsigned)														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.97 EMU3nVD2MAX Register Contents

Bit Position	Bit	Function
31	—	This bits is read as 0. The write value should be 0.
30 to 0	DATA	Data bits (unsigned integer) Set the maximum width of the d-axis voltage value for the results of the non-interference control.

Note: For details, see **Section 25.4.5(1) Func(pwm1) dq-axis voltage correction (non-interference control)**.

25.3.2.76 EMU3n Non-Interference Control q-Axis Maximum Value Register (EMU3nVQ2MAX)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + $032C_H$

Value after reset: $0000\ 0000_H$

Category: PWM IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	DATA (unsigned)														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.98 EMU3nVQ2MAX Register Contents

Bit Position	Bit	Function
31	—	This bits is read as 0. The write value should be 0.
30 to 0	DATA	Data bits (unsigned integer) Set the maximum width of the q-axis voltage value for the results of the non-interference control.

Note: For details, see **Section 25.4.5(1) Func(pwm1) dq-axis voltage correction (non-interference control)**.

25.3.2.77 EMU3n PWM IP Electrical Angle Offset Register (EMU3nPHI)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0338_H

Value after reset: 0000_H

Category: PWM IP

Bit	b15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	b0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.99 EMU3nPHI Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Set the offset value for the electrical angle to be applied to the PWM IP.

Note: For details, see **Section 25.4.5(2) Func(pwm2) Electrical angle offset processing**.

25.3.2.78 EMU3n PWM IP Electrical Angle Adjustment Coefficient Register (EMU3nGTHT)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $033C_H$

Value after reset: 0100_H

Category: PWM IP

Table 25.100 EMU3nGTHT Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed fixed-point numbers) (b15: Sign part) Set the coefficient of the multiplication to be executed for the electrical angle to be applied to the PWM IP.

Note: For details, see **Section 25.4.5(2) Func(pwm2) Electrical angle offset processing**.

This register is used to store fixed-point numbers consisting of an 8-bit integer part and an 8-bit fractional part. After a multiplication is performed using the value of this register as the multiplication coefficient, the results of the multiplication are shifted 8 bits to the right.

25.3.2.79 EMU3n PWM IP Electrical Angle Software Input Register (EMU3nTHTFORESFT)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $033E_H$

Value after reset: 0000_H

Category: PWM IP

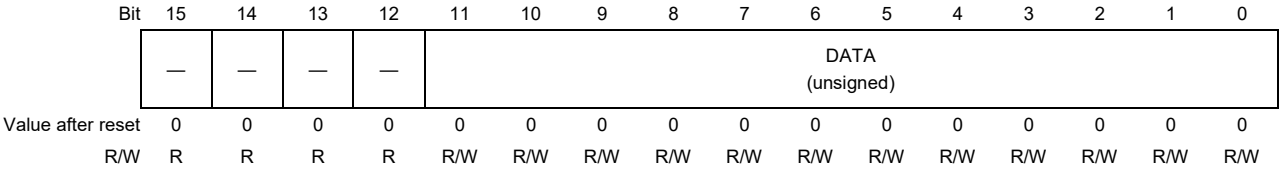


Table 25.101 EMU3nTHTFORESFT Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Set the software input value of the electrical angle to be applied to the PWM IP.

Note: For details, see **Section 25.4.5(2) Func(pwm2) Electrical angle offset processing**.

25.3.2.80 EMU3n PWM IP Post Correction Electrical Angle Register (EMU3nTHTEPWM)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0340_H

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.102 EMU3nTHTEPWM Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Set the corrected electrical angle value.

Note: For details, see **Section 25.4.5(2) Func(pwm2) Electrical angle offset processing**.

25.3.2.81 EMU3n dq-Axis Voltage Phase Angle Software Input Register (EMU3nTHTVSFT)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0342_H

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.103 EMU3nTHTVSFT Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Set the software input value of the dq-axis voltage phase angle to be applied to the PWM IP.

Note: For details, see **Section 25.4.5(3) Func(pwm3) 2-phase-to-3-phase conversion, Section 25.4.5(4), Func(pwm4) PWM modulation**.

25.3.2.82 EMU3n dq-Axis Voltage Value Software Input Register (EMU3nVDQSFT)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0344_H

Value after reset: $0000\ 0000_H$

Category: PWM IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.104 EMU3nVDQSFT Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer) (b31: Sign part) Set the software input value of the dq-axis voltage value to be applied to the PWM IP.

Note: For details, see **Section 25.4.5(3) Func(pwm3) 2-phase-to-3-phase conversion**, and **Section 25.4.5(4) Func(pwm4) PWM modulation**.

25.3.2.83 EMU3n 3-Phase Voltage Conversion Coefficient Register (EMU3nSR23)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0348_H

Value after reset: $0000\ D106_H$

Category: PWM IP

	Integer part															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	Fraction part															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	1	1	0	1	0	0	0	1	0	0	0	0	0	1	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.105 EMU3nSR23 Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed fixed-point numbers) (b31: Sign part) Set the coefficient to be used when converting the dq-axis voltage values to the 3-phase voltage values. The value established after a reset is $\sqrt{(2/3)}$.

Note: For details, see **Section 25.4.5(3) Func(pwm3) 2-phase-to-3-phase conversion**.

This register is used to store fixed-point numbers consisting of a 16-bit integer part and a 16-bit fractional part. After a multiplication is performed using the value of this register as the multiplication coefficient, the results of the multiplication are shifted 16 bits to the right.

This register is set a value of $0000\ D106_H$ after a reset. This is equivalent to the following calculation results:

$$\sqrt{(2/3)} \times 65536 = 0000\ D106_H \text{ (rounded off the fractional part)}$$

25.3.2.84 EMU3n Post 3-Phase Voltage Conversion U-Phase Voltage Value Register (EMU3nVU)
EMU3n Post 3-Phase Voltage Conversion V-Phase Voltage Value Register (EMU3nVV)
EMU3n Post 3-Phase Voltage Conversion W-Phase Voltage Value Register (EMU3nVW)
EMU3n PWM Post Modulation U-Phase Voltage Value Register (EMU3nVU0)
EMU3n PWM Post Modulation V-Phase Voltage Value Register (EMU3nVV0)
EMU3n PWM Post Modulation W-Phase Voltage Value Register (EMU3nVW0)
EMU3n Post Duty Factor Calculation U-Phase Voltage Value Register (EMU3nVU1)
EMU3n Post Duty Factor Calculation V-Phase Voltage Value Register (EMU3nVV1)
EMU3n Post Duty Factor Calculation W-Phase Voltage Value Register (EMU3nVW1)
EMU3n Post Offset Addition U-Phase Voltage Value Register (EMU3nVU2)
EMU3n Post Offset Addition V-Phase Voltage Value Register (EMU3nVV2)
EMU3n Post Offset Addition W-Phase Voltage Value Register (EMU3nVW2)
EMU3n Post Correction d-Axis Voltage Value Register (EMU3nVD2)
EMU3n Post Correction q-Axis Voltage Value Register (EMU3nVQ2)
EMU3n Post Limit Processing U-Phase Voltage Value Register (EMU3nVUFIX)
EMU3n Post Limit Processing V-Phase Voltage Value Register (EMU3nVVFIX)
EMU3n Post Limit Processing W-Phase Voltage Value Register (EMU3nVWFIX)
EMU3n U-Phase PWM Value Register (EMU3nPWMU0)
EMU3n V-Phase PWM Value Register (EMU3nPWMV0)
EMU3n W-Phase PWM Value Register (EMU3nPWMW0)
EMU3n PWM Modulation Peak Value Register (EMU3nTMAX)
EMU3n Duty Factor Upper-Limit Value Register (EMU3nDTUL)
EMU3n Duty Factor Lower-Limit Value Register (EMU3nDTLL)

Access: Readable/writable in 32-bit units.

Address: EMU3nVU: <EMU3n_base> + $034C_H$

EMU3nVV: <EMU3n_base> + 0350_H

EMU3nVW: <EMU3n_base> + 0354_H

EMU3nVU0: <EMU3n_base> + $035C_H$

EMU3nVV0: <EMU3n_base> + 0360_H

EMU3nVW0: <EMU3n_base> + 0364_H

EMU3nVU1: <EMU3n_base> + 0370_H

EMU3nVV1: <EMU3n_base> + 0374_H

EMU3nVW1: <EMU3n_base> + 0378_H

EMU3nVU2: <EMU3n_base> + 0384_H

EMU3nVV2: <EMU3n_base> + 0388_H

EMU3nVW2: <EMU3n_base> + $038C_H$

EMU3nVD2: <EMU3n_base> + 0330_H

EMU3nVQ2: <EMU3n_base> + 0334_H

EMU3nVUFIX: <EMU3n_base> + 0398_H

EMU3nVVFIX: <EMU3n_base> + $039C_H$

EMU3nVWFIX: <EMU3n_base> + $03A0_H$

EMU3nPWMU0: <EMU3n_base> + $03AC_H$

EMU3nPWMV0: <EMU3n_base> + $03B0_H$

EMU3nPWMW0: <EMU3n_base> + $03B4_H$

EMU3nTMAX: <EMU3n_base> + 0358_H

EMU3nDTUL: <EMU3n_base> + 0390_H

EMU3nDTLL: <EMU3n_base> + 0394_H

Value after reset: $0000\ 0000_H$

Category: PWM IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.106 EMU3nVU, EMU3nVV, EMU3nVW, EMU3nVU0, EMU3nVV0, EMU3nVW0, EMU3nVU1, EMU3nVV1, EMU3nVW1, EMU3nVU2, EMU3nVV2, EMU3nVW2, EMU3nVD2, EMU3nVQ2, EMU3nVUFIX, EMU3nVVFIX, EMU3nVWFIX, EMU3nPWMU0, EMU3nPWMV0, EMU3nPWMW0, EMU3nTMAX, EMU3nDTUL, EMU3nDTLL Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer) (b31: Sign part)

Note: For details, see **Section 25.4.5, PWM IP**.

25.3.2.85 EMU3n Digit Position Alignment 1 Register (EMU3nPWMK1)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + 0368_H

Value after reset: $0000\ 0000_H$

Category: PWM IP

	Integer part															
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	Fraction part															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.107 EMU3nPWMK1 Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed fixed-point numbers) (b31: Sign part) Set the coefficient value to be applied to 3-phase voltage scaling adjustment processing.

Note: For details, see **Section 25.4.5(5) Func(pwm5) Duty factor calculation**.

This register is used to store fixed-point numbers consisting of a 16-bit integer part and a 16-bit fractional part. After a multiplication is performed using the value of this register as the multiplication coefficient, the results of the multiplication are shifted 16 bits to the right.

25.3.2.86 EMU3n Input Voltage Register (EMU3nVOLV)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $036C_H$

Value after reset: 0000_H

Category: PWM IP

		Integer part															
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	DATA (signed)																
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Table 25.108 EMU3nVOLV Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Set the coefficient value to be applied to 3-phase voltage scaling adjustment processing.

Note: For details, see **Section25.4.5(5) Func(pwm5) Duty factor calculation**, **Section 25.4.5(10) Func(pwm11) PWM value limit processing**.

This register is for integers. After a division is performed on the number in this register as the denominator, the fractional part of the calculation results is truncated.

25.3.2.87 EMU3n U-Phase Voltage Correction Value Register (EMU3nVUOFS)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $037C_H$

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.109 EMU3nVUOFS

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Set the correction value for the U-phase voltage value established after the duty factor is calculated.

Note: For details, see **Section25.4.5(6) Func(pwm6) Offset addition**.

25.3.2.88 EMU3n V-Phase Voltage Correction Value Register (EMU3nVVOFS)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $037E_H$

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.110 EMU3nVVOFS Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Set the correction value for the V-phase voltage value established after the duty factor is calculated.

Note: For details, see **Section25.4.5(6) Func(pwm6) Offset addition**.

25.3.2.89 EMU3n W-Phase Voltage Correction Value Register (EMU3nVWOFS)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0380_H

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.111 EMU3nVWOFS Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Set the correction value for the W-phase voltage value established after the duty factor is calculated.

Note: For details, see **Section 25.4.5(6) Func(pwm6) Offset addition**.

25.3.2.90 EMU3n Digit Position Alignment 2 Register (EMU3nPWMK2)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $03A4_H$

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.112 EMU3nPWMK2 Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed) (b15: Sign part) Set the coefficient value for multiplications that is applied to PWM value calculation processing by the PWM IP.

Note: For details, see **Section 25.4.5(8) Func(pwm9) PWM value calculation processing**.

25.3.2.91 EMU3n Dead Time Setting Register (EMU3nDTT)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $03A6_H$

Value after reset: $0FFF_H$

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.113 EMU3nDTT Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer) Set the short-circuit prevention time to be applied to PWM value calculation processing by the PWM IP. This register must be set 0 or an integer from 2 to the value of the EMU3nCARR register divided by 2

Note: For details, see **Section 25.4.5(8) Func(pwm9) PWM value calculation processing** and **Section 25.4.5(10) Func(pwm11) PWM value limit processing**.

25.3.2.92 EMU3n Carrier Period Register (EMU3nCARR)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $03A8_H$

Value after reset: $7FFF_H$

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.114 EMU3nCARR Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer) Set the carrier period value to be applied to PWM value calculation processing by the PWM IP. The value of this register is also applied when software-inputting the PWM compare values and carrier period to be output to the TSG3.

Note: For details, see **Section 25.4.5(9) Func(pwm10) Dead time compensation**, **Section 25.4.5(10) Func(pwm11) PWM value limit processing**, and **Section 25.4.5(11) Func(pwm_out) PWM value output processing**.

Set up the register so that the sum of the values of the EMU3nCARR and EMU3nDTT registers is not greater than $FFFF_H$.

25.3.2.93 EMU3n Carrier Period Buffer Register (EMU3nCARRBUF)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $03AA_H$

Value after reset: $FFFF_H$

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.115 EMU3nCARRBUF Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer) The value of this register is applied as the carrier period value when outputting the PWM compare values from the PWM IP to the TSG3.

Note: For details, see **Section 25.4.5(10) Func(pwm11) PWM value limit processing**.

25.3.2.94 EMU3n Dead Time Compensation Threshold Value Register (EMU3nDTOTH)

Access: Readable/writable in 32-bit units.

Address: <EMU3n_base> + $03B8_H$

Value after reset: $7FFF\ FFFF_H$

Category: PWM IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	DATA (unsigned)														
Value after reset	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.116 EMU3nDTOTH Register Contents

Bit Position	Bit	Function
31	—	This bits is read as 0. The write value should be 0.
30 to 0	DATA	Data bits (unsigned integer) Set the threshold value to be applied to dead time compensation processing.

Note: For details, see **Section 25.4.5(9) Func(pwm10) Dead time compensation**.

25.3.2.95 EMU3n Dead Time Compensation Addend for Positive Current Register (EMU3nDTOPV)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $03BC_H$

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.117 EMU3nDTOPV Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Set the addend for positive 3-phase current values that is to be applied to dead time compensation processing.

Note: For details, see **Section 25.4.5(9) Func(pwm10) Dead time compensation**.

25.3.2.96 EMU3n Dead Time Compensation Addend for Negative Current Register (EMU3nDTONV)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $03BE_H$

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.118 EMU3nDTONV Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15: Sign part) Set the addend for negative 3-phase current values that is to be applied to dead time compensation processing.

Note: For details, see **Section 25.4.5(9) Func(pwm10) Dead time compensation**.

25.3.2.97 EMU3n Post Dead Time Compensation U-Phase PWM Value Register (EMU3nPWMUDT)
EMU3n Post Dead Time Compensation V-Phase PWM Value Register (EMU3nPWMVDT)
EMU3n Post Dead Time Compensation W-Phase PWM Value Register (EMU3nPWMWDT)

Access: Readable/writable in 32-bit units.

Address: EMU3nPWMUDT: <EMU3n_base> + $03C0_H$

EMU3nPWMVDT: <EMU3n_base> + $03C4_H$

EMU3nPWMWDT: <EMU3n_base> + $03C8_H$

Value after reset: $0000\ 0000_H$

Category: PWM IP

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.119 EMU3nPWMUDT, EMU3nPWMVDT, EMU3nPWMWDT Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer) (b31: Sign part)

Note: For details, see **Section 25.4.5(9) Func(pwm10) Dead time compensation**.

25.3.2.98 EMU3n PWM Upper-Limit Value Register (EMU3nPWMUL)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $03CC_H$

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.120 EMU3nPWMUL Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer) Set the upper-limit value of the PWM value to be applied to the PWM value limit processing by the PWM IP.

Note: For details, see **Section 25.4.5(10) Func(pwm11) PWM value limit processing**.

25.3.2.99 EMU3n PWM Lower-Limit Register (EMU3nPWMLL)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $03CE_H$

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.121 EMU3nPWMLL Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer) Set the lower-limit value of the PWM value to be applied to the PWM value limit processing by the PWM IP.

Note: For details, see **Section 25.4.5(10) Func(pwm11) PWM value limit processing**.

25.3.2.100 EMU3n U-Phase PWM Compare Value Register (EMU3nPWMUIP)
EMU3n V-Phase PWM Compare Value Register (EMU3nPWMVIP)
EMU3n W-Phase PWM Compare Value Register (EMU3nPWMWIP)

Access: Readable/writable in 16-bit units.

Address: EMU3nPWMUIP: <EMU3n_base> + $03D0_H$

EMU3nPWMVIP: <EMU3n_base> + $03D2_H$

EMU3nPWMWIP: <EMU3n_base> + $03D4_H$

Value after reset: 0000_H

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.122 EMU3nPWMUIP, EMU3nPWMVIP, EMU3nPWMWIP Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer)

Note: For details, see **Section 225.4.5(10) Func(pwm11) PWM value limit processing**.

25.3.2.101 EMU3n U-Phase PWM Compare Value Software Input Register (EMU3nPWMU)
EMU3n V-Phase PWM Compare Value Software Input Register (EMU3nPWMV)
EMU3n W-Phase PWM Compare Value Software Input Register (EMU3nPWMW)

Access: Readable/writable in 16-bit units.

Address: EMU3nPWMU: <EMU3n_base> + $03D8_H$

EMU3nPWMV: <EMU3n_base> + $03DA_H$

EMU3nPWMW: <EMU3n_base> + $03DC_H$

Value after reset: $47FF_H$

Category: PWM IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.123 EMU3nPWMU, EMU3nPWMV, EMU3nPWMW Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer)

Note: For details, see **Section 25.4.5(11) Func(pwm_out) PWM value output processing**.

This register must not be set a value that is greater than the sum of the values of the EMU3nCARR and EMU3nDTT registers.

25.3.2.102 EMU3n Rectangle IP Control Register (EMU3nRECCTR)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0480_H

Value after reset: 04_H

Category: Rectangle IP

Bit	7	6	5	4	3	2	1	0
	FDRCT	—	—	—	—	FIP POSI	SLCT CMP0	SET REC
Value after reset	0	0	0	0	0	1	0	0
R/W	R/W	R	R	R	R	R/W	R/W	R/W

Table 25.124 EMU3nRECCTR Register Contents

Bit Position	Bit	Function
7	FDRCT	Rotational direction select bit 0: Positive rotation 1: Reverse rotation
6 to 3	—	These bits are read as 0. The write value should be 0.
2	FIPPOSI	Switching instruction select bit 0: Uses the value calculated by the rectangle IP. 1: Uses the value of the EMU3nPSWSFT register.
1	SLCTCMP0	Angle compare 0 match detection compare value select bit 0: Uses the value of the EMU3nCMP0 register. 1: Uses the value calculated by the rectangle IP.
0	SETREC	rectangle setting bit 0: Uses the value of the EMU3nPTNN register. 1: Uses the value selected by the rectangle IP (selected from the values of the EMU3nPTNAB, EMU3nPTNCD, and EMU3nPTNEF registers).

This register is used to control the rectangle IP. For details on the control bits of this register, see **Section 25.4.7.1, Batch Rectangle IP**.

It is disallowed to set the combination of the SLCTCMP0 bit to "0" and the SETREC bit to "1".

25.3.2.103 EMU3n Rectangle Output Software Control Pattern Register (EMU3nPTNN)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0484_H

Value after reset: 00_H

Category: Rectangle IP

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WPTN	VPTN	UPTN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.125 EMU3nPTNN Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WPTN	W-phase rectangle wave pattern setting bit 0: Low level 1: High level
1	VPTN	V-phase rectangle wave pattern setting bit 0: Low level 1: High level
0	UPTN	U-phase rectangle wave pattern setting bit 0: Low level 1: High level

Note: For details, see **Section 25.4.7.1, Batch Rectangle IP**.

The value of this register is used as the rectangle wave pattern value when outputting the rectangle wave pattern in combination with the software processing by the CPU program.

25.3.2.104 EMU3n Rectangle Output Pattern AB Register (EMU3nPTNAB)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0485_H

Value after reset: 00_H

Category: Rectangle IP

Bit	7	6	5	4	3	2	1	0
	—	—	WPTNB	VPTNB	UPTNB	WPTNA	VPTNA	UPTNA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.126 EMU3nPTNAB Register Contents

Bit Position	Bit	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5	WPTNB	W-phase rectangle wave pattern B setting bit. 0: Low level 1: High level
4	VPTNB	V-phase rectangle wave pattern B setting bit. 0: Low level 1: High level
3	UPTNB	U-phase rectangle wave pattern B setting bit. 0: Low level 1: High level
2	WPTNA	W-phase rectangle wave pattern A setting bit. 0: Low level 1: High level
1	VPTNA	V-phase rectangle wave pattern A setting bit. 0: Low level 1: High level
0	UPTNA	U-phase rectangle wave pattern A setting bit. 0: Low level 1: High level

Note: For details, see **Section 25.4.7.1, Batch Rectangle IP**.

The value of this register is used as the rectangle wave pattern value when outputting the rectangle wave pattern by rectangle IP hardware.

25.3.2.105 EMU3n Rectangle Output Pattern CD Register (EMU3nPTNCD)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0486_H

Value after reset: 00_H

Category: Rectangle IP

Bit	7	6	5	4	3	2	1	0
	—	—	WPTND	VPTND	UPTND	WPTNC	VPTNC	UPTNC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.127 EMU3nPTNCD Register Contents

Bit Position	Bit	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5	WPTND	W-phase rectangle wave pattern D setting bit. 0: Low level 1: High level
4	VPTND	V-phase rectangle wave pattern D setting bit. 0: Low level 1: High level
3	UPTND	U-phase rectangle wave pattern D setting bit. 0: Low level 1: High level
2	WPTNC	W-phase rectangle wave pattern C setting bit. 0: Low level 1: High level
1	VPTNC	V-phase rectangle wave pattern C setting bit. 0: Low level 1: High level
0	UPTNC	U-phase rectangle wave pattern C setting bit. 0: Low level 1: High level

Note: For details, see **Section 25.4.7.1, Batch Rectangle IP**.

The value of this register is used as the rectangle wave pattern value when outputting the rectangle wave pattern by rectangle IP hardware.

25.3.2.106 EMU3n Rectangle Output Pattern EF Register (EMU3nPTNEF)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0487_H

Value after reset: 00_H

Category: Rectangle IP

Bit	7	6	5	4	3	2	1	0
	—	—	WPTNF	VPTNF	UPTNF	WPTNE	VPTNE	UPTNE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.128 EMU3nPTNEF Register Contents

Bit Position	Bit	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5	WPTNF	W-phase rectangle wave pattern F setting bit. 0: Low level 1: High level
4	VPTNF	V-phase rectangle wave pattern F setting bit. 0: Low level 1: High level
3	UPTNF	U-phase rectangle wave pattern F setting bit. 0: Low level 1: High level
2	WPTNE	W-phase rectangle wave pattern E setting bit. 0: Low level 1: High level
1	VPTNE	V-phase rectangle wave pattern E setting bit. 0: Low level 1: High level
0	UPTNE	U-phase rectangle wave pattern E setting bit. 0: Low level 1: High level

Note: For details, see **Section 25.4.7.1, Batch Rectangle IP**.

The value of this register is used as the rectangle wave pattern value when outputting the rectangle wave pattern by rectangle IP hardware.

25.3.2.107 EMU3n Angle Compare 0 Comparison Value Software Input Register (EMU3nCMP0)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0488_H

Value after reset: 0000_H

Category: Angle generation IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.129 EMU3nCMP0 Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Set the software -input comparison value for angle compare 0 (electrical angle).

Note: For details, see **Section 25.4.6, Rectangle IP,** and **Section 25.4.2(3) Func(ang8) angle compare match**.

The value of this register is used as the comparison value for angle compare 0 match detection when outputting the batch rectangle wave patterns in combination with software processing by the CPU program.

The value of this register is also used as the comparison value for the first angle compare 0 match detection when outputting the batch rectangle wave patterns by rectangle IP hardware.

25.3.2.108 EMU3n Angle Compare 1 Comparison Value Software Input Register (EMU3nCMP1)

Access: Readable/writable in 16 bit units.

Address: <EMU3n_base> + $048A_H$

Value after reset: 0000_H

Category: Angle generation IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.130 EMU3nCMP1 Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Set the software -input comparison value for angle compare 1 (resolver angle).

Note: For details, see **Section 25.4.2(3) Func(ang8) angle compare match**.

25.3.2.109 EMU3n q-Axis Reference Voltage Phase Software Input Register (EMU3nPHQSFT)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $048C_H$

Value after reset: 0000_H

Category: Rectangle IP

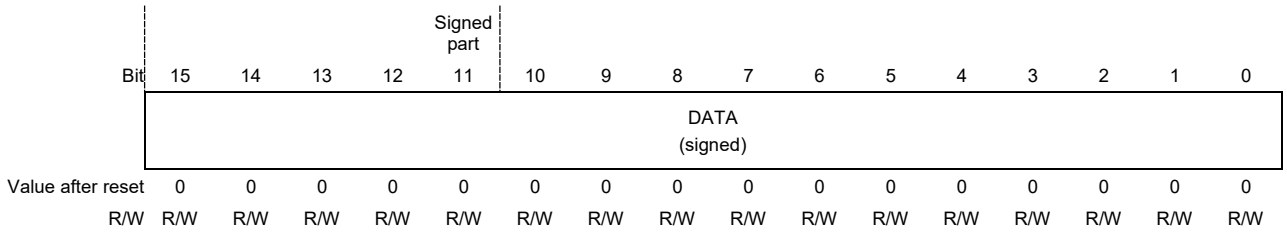


Table 25.131 EMU3nPHQSFT Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (signed integer) (b15-11: Sign part) Set the correction value for the “angle compare 0 match detection” by the rectangle IP.

Note: For details, see **Section 25.4.7.1, Batch Rectangle IP**.

25.3.2.110 EMU3n Switching Instruction Software-input Register (EMU3nPSWSFT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $048E_H$

Value after reset: 00_H

Category: Rectangle IP

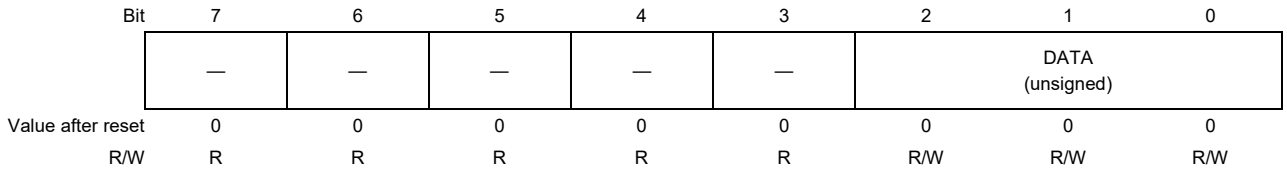


Table 25.132 EMU3nPSWSFT Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	DATA	Data bits (unsigned integer) Set the software -input switching instruction value to be applied to the rectangle IP. It is inhibited to write a value of 6 or 7 to this register. Set this register with a value from 0 to 5.

Note: For details, see **Section 25.4.7.1, Batch Rectangle IP**.

25.3.2.111 EMU3n Switching Instruction Register (EMU3nPSW)

Access: Readable in 8-bit units.

Address: <EMU3n_base> + 048F$_H$

Value after reset: 00$_H$

Category: Rectangle IP

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DATA (unsigned)		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 25.133 EMU3nPSW Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	DATA	Data bits (unsigned integer) Set the switching instruction value that is calculated by the rectangle IP.

Note: For details, see **Section 25.4.7.1, Batch Rectangle IP**.

25.3.2.112 EMU3n Angle Compare 0 Comparison Value IP Output Register (EMU3nIPCMP0)

Access: Readable in 16-bit units.

Address: <EMU3n_base> + 0490$_H$

Value after reset: 0000$_H$

Category: Rectangle IP

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.134 EMU3nIPCMP0 Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Loaded with the comparison value for angle compare 0 match detection that is calculated by the rectangle IP.

Note: For details, see **Section 25.4.7.1, Batch Rectangle IP**.

The value of this register is used as the comparison value for angle compare 0 match detection processing other than the first angle compare 0 match detection when outputting the rectangle wave patterns by rectangle IP hardware.

25.3.2.113 EMU3n Independent Rectangle IP 1 Control Register (EMU3nIRECCTR)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $04C0_H$

Value after reset: 00_H

Category: Rectangle wave generation block

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RECMD	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 25.135 EMU3nIRECCTR Register Contents

Bit Position	Bit	Function
7 to 2	—	These bits are read as 0. The write value should be 0.
1, 0	RECMD	Rectangle wave output mode switching bits 00: Outputs rectangle wave patterns to the TSG3. 01: Outputs independent rectangle IP1 wave patterns to the TSG3. 10: Outputs independent rectangle IP2 wave patterns to the TSG3. 11: inhibited

Note: For details, see **Section 25.4.7 Rectangle Wave Generation Block, Section 25.4.7.2 Independent Rectangle IP1**, and **Section 25.4.7.3, Independent Rectangle IP2.**

25.3.2.114 EMU3n Independent Rectangle IP 1 Output Pattern Update Register (EMU3nIRPTN)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $04C4_H$

Value after reset: 00_H

Category: Independent rectangle IP1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WINIPTN	VINIPTN	UINIPTN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.136 EMU3nIRPTN Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WINIPTN	W-phase rectangle wave pattern update bit. 0: Low level 1: High level
1	VINIPTN	V-phase rectangle wave pattern update bit. 0: Low level 1: High level
0	UINIPTN	U-phase rectangle wave pattern update bit. 0: Low level 1: High level

Note: For details, see **Section 25.4.7.2, Independent Rectangle IP1**.

25.3.2.115 EMU3n Independent Rectangle IP1 Flag Select Signal Initialization Register (EMU3nIRCTRST)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $04C6_H$

Value after reset: 00_H

Category: Independent rectangle IP1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WINIT	VINIT	UINIT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.137 EMU3nIRCTRST Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WINIT	W-phase flag/select signal initialization bit Setting this bit to 1 initializes the flag/select signals for W-phase processing by the independent rectangle IP1.
1	VINIT	V-phase flag/select signal initialization bit Setting this bit to 1 initializes the flag/select signals for V-phase processing by the independent rectangle IP1.
0	UINIT	U-phase flag/select signal initialization bit Setting this bit to 1 initializes the flag/select signals for U-phase processing by the independent rectangle IP1.

Note: For details, see **Section 25.4.7.2, Independent Rectangle IP1**.

The bits of these registers are automatically reset to 0 after they are set to 1. It is invalid to write a 0 to these bits. They are always read as 0.

25.3.2.116 EMU3n Independent Rectangle IP1 U-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 0 Register (EMU3nIRUCPPN0)
EMU3n Independent Rectangle IP1 U-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 1 Register (EMU3nIRUCPPN1)
EMU3n Independent Rectangle IP1 U-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 2 Register (EMU3nIRUCPPN2)
EMU3n Independent Rectangle IP1 V-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 0 Register (EMU3nIRVCPPN0)
EMU3n Independent Rectangle IP1 V-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 1 Register (EMU3nIRVCPPN1)
EMU3n Independent Rectangle IP1 V-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 2 Register (EMU3nIRVCPPN2)
EMU3n Independent Rectangle IP1 W-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 0 Register (EMU3nIRWCPPN0)
EMU3n Independent Rectangle IP1 W-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 1 Register (EMU3nIRWCPPN1)
EMU3n Independent Rectangle IP1 W-Phase Angle Compare 0 Match Detection Comparison Value/Pattern Setting 2 Register (EMU3nIRWCPPN2)

Access: Readable/writable in 16 and 32 bit units.

Address: EMU3nIRUCPPN0: <EMU3n_base> + $04C8_H$

EMU3nIRUCPPN1: <EMU3n_base> + $04CC_H$

EMU3nIRUCPPN2: <EMU3n_base> + $04D0_H$

EMU3nIRVCPPN0: <EMU3n_base> + $04D4_H$

EMU3nIRVCPPN1: <EMU3n_base> + $04D8_H$

EMU3nIRVCPPN2: <EMU3n_base> + $04DC_H$

EMU3nIRWCPPN0: <EMU3n_base> + $04E0_H$

EMU3nIRWCPPN1: <EMU3n_base> + $04E4_H$

EMU3nIRWCPPN2: <EMU3n_base> + $04E8_H$

Value after reset: $0000\ 0000_H$

Category: Independent rectangle IP1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	U/V/W PTN0/1/2
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	UCMP0,1,2[11:0]					VCMP0,1,2[11:0] (unsigned)			WCMP0,1,2[11:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.138 EMU3nIRUCPPN0, EMU3nIRUCPPN1, EMU3nIRUCPPN2, EMU3nIRVCPPN0, EMU3nIRVCPPN1, EMU3nIRVCPPN2, EMU3nIRWCPPN0, EMU3nIRWCPPN1, EMU3nIRWCPPN2 Register Contents

Bit Position	Bit	Function
31 to 17	—	These bits are read as 0. The write value should be 0.
16	UPTN0,1,2 VPTN0,1,2 WPTN0,1,2	U-phase pattern values 0,1,2 setting bit V-phase pattern values 0,1,2 setting bit W-phase pattern values 0,1,2 setting bit 0: Low level 1: High level
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	UCMP0, 1, 2[11:0] VCMP0, 1, 2[11:0] WCMP0, 1, 2[11:0]	U-phase angle compare 0 match detection comparison values 0,1,2 setting bits (unsigned integer) V-phase angle compare 0 match detection comparison values 0,1,2 setting bits (unsigned integer) W-phase angle compare 0 match detection comparison values 0,1,2 setting bits (unsigned integer) Set comparison values for angle compare 0 match detection.

Note: For details, see **Section 25.4.7.2, Independent Rectangle IP1**.

25.3.2.117 EMU3n Independent Rectangle IP 1 Flag Monitor Register (EMU3nIRFLGM)

Access: Readable in 16-bit units.

Address: <EMU3n_base> + $04EC_H$

Value after reset: 0000_H

Category: Independent rectangle IP1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WFLG2	WFLG1	WFLG0	VFLG2	VFLG1	VFLG0	UFLG2	UFLG1	UFLG0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.139 EMU3nIRFLGM Register Contents (1/2)

Bit Position	Bit	Function
15 to 9	—	These bits are read as 0. The write value should be 0.
8	WFLG2	W-phase flag 2 status bit 0: Before the write from CPU to EMU3nIRWCPPN2 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRWCPPN2 register and no angle compare 0 match is detected.
7	WFLG1	W-phase flag 1 status bit 0: Before the write from CPU to EMU3nIRWCPPN1 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRWCPPN1 register and no angle compare 0 match is detected.
6	WFLG0	W-phase flag 0 status bit 0: Before the write from CPU to EMU3nIRWCPPN0 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRWCPPN0 register and no angle compare 0 match is detected.
5	VFLG2	V-phase flag 2 status bit 0: Before the write from CPU to EMU3nIRVCPPN2 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRVCPPN2 register and no angle compare 0 match is detected.
4	VFLG1	V-phase flag 1 status bit 0: Before the write from CPU to EMU3nIRVCPPN1 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRVCPPN1 register and no angle compare 0 match is detected.
3	VFLG0	V-phase flag 0 status bit 0: Before the write from CPU to EMU3nIRVCPPN0 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRVCPPN0 register and no angle compare 0 match is detected.
2	UFLG2	U-phase flag 2 status bit 0: Before the write from CPU to EMU3nIRUCPPN2 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRUCPPN2 register and no angle compare 0 match is detected.

Table 25.139 EMU3nIRFLGM Register Contents (2/2)

Bit Position	Bit	Function
1	UFLG1	U-phase flag 1 status bit 0: Before the write from CPU to EMU3nIRUCPPN1 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRUCPPN1 register and no angle compare 0 match is detected.
0	UFLG0	U-phase flag 0 status bit 0: Before the write from CPU to EMU3nIRUCPPN0 register or after detection of an angle compare 0 match 1: After the write from CPU to EMU3nIRUCPPN0 register and no angle compare 0 match is detected.

Note: For details, see **Section 25.4.7.2, Independent Rectangle IP1.**

25.3.2.118 EMU3n Independent Rectangle IP 1 Select Signal Monitor Register (EMU3nIRSELM)

Access: Readable in 16-bit units.

Address: <EMU3n_base> + $04EE_H$

Value after reset: 0049_H

Category: Independent rectangle IP1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WSEL[2:0]			VSEL[2:0]			USEL[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.140 EMU3nIRSELM Register Contents

Bit Position	Bit	Function
15 to 9	—	These bits are read as 0. The write value should be 0.
8 to 6	WSEL[2:0]	W-phase selection status bit 001: EMU3nIRWCPPN0 register is selected. 010: EMU3nIRWCPPN1 register is selected. 100: EMU3nIRWCPPN2 register is selected.
5 to 3	VSEL[2:0]	V-phase selection status bit 001: EMU3nIRVCPPN0 register is selected. 010: EMU3nIRVCPPN1 register is selected. 100: EMU3nIRVCPPN2 register is selected.
2 to 0	USEL[2:0]	U-phase selection status bit 001: EMU3nIRUCPPN0 register is selected. 010: EMU3nIRUCPPN1 register is selected. 100: EMU3nIRUCPPN2 register is selected.

Note: For details, see **Section 25.4.7.2, Independent Rectangle IP1**.

25.3.2.119 EMU3n Independent Rectangle IP 2 Control Register (EMU3nNRECCTR)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + 0500_H

Value after reset: 00_H

Category: Independent rectangle IP2

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	EN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.141 EMU3nNRECCTR Register Contents

Bit Position	Bit	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	EN	Independent rectangle IP2 operation bit. 0: Stopped (U-phase /V-phase /W-phase pattern values are fixed at low level.) 1: Running

Note: For details, see **Section 25.4.7.3, Independent Rectangle IP2**.

25.3.2.120 EMU3n Independent Rectangle IP2 3-Phase Common Angle Correction Value Register (EMU3nNRECOFSALL)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + 0508_H

Value after reset: 0000_H

Category: Independent rectangle IP2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (signed)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.142 EMU3nNRECOFSALL Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA (signed)	Data bits (signed integer) (b11: Signed part) Set the U-/V-/W-phase offset values for the electrical angle to be applied to the independent rectangle IP2 all at once.

Note: For details, see **Section 25.4.7.3, Independent Rectangle IP2**.

The angle correction value of the U phase is the EMU3nNRECOFSALL register value + the EMU3nNRECOFSU value.

The angle correction value of the V phase is the EMU3nNRECOFSALL register value + the EMU3nNRECOFSV value.

The angle correction value of the W phase is the EMU3nNRECOFSALL register value + the EMU3nNRECOFSW value.

25.3.2.121 EMU3n Independent Rectangle IP2 U-Phase Angle Correction Value Register (EMU3nNRECOFSU)
EMU3n Independent Rectangle IP2 V-Phase Angle Correction Value Register (EMU3nNRECOFSV)
EMU3n Independent Rectangle IP2 W-Phase Angle Correction Value Register (EMU3nNRECOFSW)

Access: Readable/writable in 16-bit units.

Address: EMU3nNRECOFSU: <EMU3n_base> + $050A_H$

EMU3nNRECOFSV: <EMU3n_base> + $050C_H$

EMU3nNRECOFSW: <EMU3n_base> + $050E_H$

Value after reset: 0000_H

Category: Independent rectangle IP2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (signed)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.143 EMU3nNRECOFSU, EMU3nNRECOFSV, EMU3nNRECOFSW Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA (signed)	Data bits (signed integer) (b11: Signed part)

Note: For details, see **Section 25.4.7.3, Independent Rectangle IP2**.

The angle correction value of the U phase is the EMU3nNRECOFSALL register value + the EMU3nNRECOFSU value.

The angle correction value of the V phase is the EMU3nNRECOFSALL register value + the EMU3nNRECOFSV value.

The angle correction value of the W phase is the EMU3nNRECOFSALL register value + the EMU3nNRECOFSW value.

25.3.2.122 EMU3n Independent Rectangle IP2 U-Phase Compare Control k Register (EMU3nNRECUk) (k = 0 to 7)

Access: Readable/writable in 8, 16, and 32 bit units.

Address: EMU3nNRECU0: <EMU3n_base> + 0510_H
EMU3nNRECU1: <EMU3n_base> + 0514_H
EMU3nNRECU2: <EMU3n_base> + 0518_H
EMU3nNRECU3: <EMU3n_base> + $051C_H$
EMU3nNRECU4: <EMU3n_base> + 0520_H
EMU3nNRECU5: <EMU3n_base> + 0524_H
EMU3nNRECU6: <EMU3n_base> + 0528_H
EMU3nNRECU7: <EMU3n_base> + $052C_H$

Value after reset: $0000\ 0000_H$

Category: Independent rectangle IP2

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	UMOD	—	—	UINT	UEN	—	—	—	—	—	—	—	UPTN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	UCMP[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.144 EMU3nNRECUk Register Contents (1/2)

Bit Position	Bit	Function
31 to 29	—	These bits are read as 0. The write value should be 0.
28	UMOD	U-phase angle compare 0 match detection enable mode bit k 0: UEN being 1 is not automatically updated to 0 each time a match in comparison is detected. 1: UEN being 1 is automatically updated to 0 each time a match in comparison is detected.
27, 26	—	These bits are read as 0. The write value should be 0.
25	UINT	Interrupt output enable setting bit k (independent rectangle IP2 U-phase angle compare 0 match detection) Detection of a match in comparison constitutes an interrupt source. 0: Disabled 1: Enabled
24	UEN	U-phase angle compare 0 match detection enable bit k This bit enables updating of the rectangle wave output pattern and the input of a trigger for control of A/D conversion and latching of the angle value control on detection of a compare match. 0: Disabled 1: Enabled **CAUTION:** Manual setting by writing a register and automatic control by UMOD are performed.
23 to 17	—	These bits are read as 0. The write value should be 0.
16	UCMP[11:0]	U-phase pattern value setting bit k 0: Low level 1: High level

Table 25.144 EMU3nNRECUk Register Contents (2/2)

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	UCMP[11:0]	U-phase angle compare 0 match detection comparison value setting bits k (unsigned integer) Set the comparison value for angle compare 0 match detection.

Note: For details, see **Section 25.4.7.3, Independent Rectangle IP2**.

25.3.2.123 EMU3n Independent Rectangle IP2 V-Phase Compare Control k Register (EMU3nNRECVk) (k = 0 to 7)

Access: Readable/writable in 8, 16, and 32 bit units.

Address: EMU3nNRECV0: <EMU3n_base> + 0530_H
EMU3nNRECV1: <EMU3n_base> + 0534_H
EMU3nNRECV2: <EMU3n_base> + 0538_H
EMU3nNRECV3: <EMU3n_base> + $053C_H$
EMU3nNRECV4: <EMU3n_base> + 0540_H
EMU3nNRECV5: <EMU3n_base> + 0544_H
EMU3nNRECV6: <EMU3n_base> + 0548_H
EMU3nNRECV7: <EMU3n_base> + $054C_H$

Value after reset: $0000\ 0000_H$

Category: Independent rectangle IP2

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VMOD	—	—	VINT	VEN	—	—	—	—	—	—	—	VPTN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VCMP[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.145 EMU3nNRECVk Register Contents (1/2)

Bit Position	Bit	Function
31 to 29	—	These bits are read as 0. The write value should be 0.
28	VMOD	V-phase angle compare 0 match detection enable mode bit k 0: VEN being 1 is not automatically updated to 0 each time a match in comparison is detected. 1: VEN being 1 is automatically updated to 0 each time a match in comparison is detected.
27, 26	—	These bits are read as 0. The write value should be 0.
25	VINT	Interrupt output enable setting bit k (independent rectangle IP2 V-phase angle compare 0 match detection) Detection of a match in comparison constitutes an interrupt source. 0: Disabled 1: Enabled
24	VEN	V-phase angle compare match detection enable bit k This bit enables updating of the rectangle wave output pattern and the input of a trigger for control of A/D conversion and latching of the angle value control on detection of a compare match. 0: Disabled 1: Enabled **CAUTION:** Manual setting by writing a register and automatic control by VMOD are performed.
23 to 17	—	These bits are read as 0. The write value should be 0.
16	VPTN	V-phase pattern value setting bit k 0: Low level 1: High level

Table 25.145 EMU3nNRECVk Register Contents (2/2)

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	VCMP[11:0]	V-phase angle compare 0 match detection comparison value setting bits k (unsigned integer) Set the comparison value for angle compare 0 match detection.

Note: For details, see **Section 25.4.7.3, Independent Rectangle IP2**.

25.3.2.124 EMU3n Independent Rectangle IP2 W-phase Compare Control k Register (EMU3nNRECWk) (k = 0 to 7)

Access: Readable/writable in 8, 16, and 32 bit units.

Address: EMU3nNRECW0: <EMU3n_base> + 0550_H
EMU3nNRECW1: <EMU3n_base> + 0554_H
EMU3nNRECW2: <EMU3n_base> + 0558_H
EMU3nNRECW3: <EMU3n_base> + $055C_H$
EMU3nNRECW4: <EMU3n_base> + 0560_H
EMU3nNRECW5: <EMU3n_base> + 0564_H
EMU3nNRECW6: <EMU3n_base> + 0568_H
EMU3nNRECW7: <EMU3n_base> + $056C_H$

Value after reset: $0000\ 0000_H$

Category: Independent rectangle IP2

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	WMOD	—	—	WINT	WEN	—	—	—	—	—	—	—	WPTN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	WCMP[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.146 EMU3nNRECWk Register Contents (1/2)

Bit Position	Bit	Function
31 to 29	—	These bits are read as 0. The write value should be 0.
28	WMOD	W-phase angle compare 0 match detection enable mode bit k 0: WEN being 1 is not automatically updated to 0 each time a match in comparison is detected. 1: WEN being 1 is automatically updated to 0 each time a match in comparison is detected.
27, 26	—	These bits are read as 0. The write value should be 0.
25	WINT	Interrupt output enable setting bit k (independent rectangle IP2 W-phase angle compare 0 match detection) Detection of a match in comparison constitutes an interrupt source. 0: Disabled 1: Enabled
24	WEN	W-phase angle compare 0 match detection enable bit k This bit enables updating of the rectangle wave output pattern and the input of a trigger for control of A/D conversion and latching of the angle value control on detection of a compare match. 0: Disabled 1: Enabled **CAUTION:** Manual setting by writing a register and automatic control by WMOD are performed.
23 to 17	—	These bits are read as 0. The write value should be 0.
16	WPTN	W-phase pattern value setting bit k 0: Low level 1: High level

Table 25.146 EMU3nNRECWk Register Contents (2/2)

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	WCMP[11:0]	W-phase angle compare 0 match detection comparison value setting bits k (unsigned integer) Set the comparison value for angle compare 0 match detection.

Note: For details, see **Section 25.4.7.3, Independent Rectangle IP2**.

25.3.2.125 EMU3n IIR Filter Channel k Control Register (EMU3nIIRCTRk) (k = 0 to 2)

Access: Readable/writable in 8-bit units.

Address: EMU3nIIRCTR0: <EMU3n_base> + $05C0_H$

EMU3nIIRCTR1: <EMU3n_base> + $05C1_H$

EMU3nIIRCTR2: <EMU3n_base> + $05C2_H$

Value after reset: 00_H

Category: IIR filter

Bit	7	6	5	4	3	2	1	0
	DATSEL[3:0]				TRGSEL[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.147 EMU3nIIRCTRk Register Contents

Bit Position	Bit	Function
7 to 4	DATSEL[3:0]	IIR filter data select bits b7 b6 b5 b4 0 0 0 0: IIR filter channel 0 data software input 0 0 0 1: IIR filter channel 1 data software input 0 0 1 0: IIR filter channel 2 data software input 0 0 1 1: Data retained 0 1 0 0: 0 data software input 0 1 0 1: A/D data channel 0 0 1 1 0: A/D data channel 1 0 1 1 1: A/D data channel 2 1 0 0 0: IIR filter channel 0 operation results data 1 0 0 1: IIR filter channel 1 operation results data 1 0 1 0: IIR filter channel 2 operation results data Other than above: Data retained
3 to 0	TRGSEL[3:0]	IIR filter startup trigger select bits b3 b2 b1 b0 0 0 0 0: Software-starts IIR filter channel 0 0 0 0 1: Software-starts IIR filter channel 1 0 0 1 0: Software-starts IIR filter channel 2 0 0 1 1: Not started 0 1 0 0: Starts on A/D conversion channel group completion. 0 1 0 1: Starts on A/D conversion channel 0 completion. 0 1 1 0: Starts on A/D conversion channel 1 completion. 0 1 1 1: Starts on A/D conversion channel 2 completion. 1 0 0 0: Starts on IIR filter channel 0 completion (setting is only prohibited for EMU3nIIRCTR0). 1 0 0 1: Starts on IIR filter channel 1 completion (setting is only prohibited for EMU3nIIRCTR1). 1 0 1 0: Starts on IIR filter channel 2 completion (setting is only prohibited for EMU3nIIRCTR2). Other than above: Not started

Note: For details, see **Section 25.4.9, IIR Filters**.

25.3.2.126 EMU3n IIR Filter Initialization Register (EMU3nIIRINIT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $05C4_H$

Value after reset: 00_H

Category: IIR filter

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	INIT2	INIT1	INIT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.148 EMU3nIIRINIT Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	INIT2	IIR filter channel 2 initialization bit 0: Does not initialize delay 1 data and delay 2 data. 1: Initializes delay 1 data and delay 2 data.
1	INIT1	IIR filter channel 1 initialization bit 0: Does not initialize delay 1 data and delay 2 data. 1: Initializes delay 1 data and delay 2 data.
0	INIT0	IIR filter channel 0 initialization bit 0: Does not initialize delay 1 data and delay 2 data. 1: Initializes delay 1 data and delay 2 data.

Note: For details, see **Section 25.4.9, IIR Filters**.

Setting a bit to 1 initializes the corresponding channel. The bits are automatically reset to 0 after being set to 1. It is inhibited to write a 0. The bits are always read as 0.

25.3.2.127 EMU3n IIR Filter Software Startup Register (EMU3nIIRSFT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $05C8_H$

Value after reset: 00_H

Category: IIR filter

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TRG2	TRG1	TRG0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.149 EMU3nIIRSFT Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	TRG2	IIR filter channel 2 software start bit 0: Does not software-start the channel. 1: Software-starts the channel.
1	TRG1	IIR filter channel 1 software start bit 0: Does not software-start the channel. 1: Software-starts the channel.
0	TRG0	IIR filter channel 0 software start bit 0: Does not software-start the channel. 1: Software-starts the channel.

Note: For details, see **Section 25.4.9, IIR Filters**.

Setting a bit to 1 software-starts the corresponding channel. The bits are automatically reset to 0 after being set to 1. It is inhibited to write a 0. The bits are always read as 0.

25.3.2.128 EMU3n IIR Filter Coefficient Shift Amount Reload Register (EMU3nIIRRLD)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $05CC_H$

Value after reset: 00_H

Category: IIR filter

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	RLD2	RLD1	RLD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.150 EMU3nIIRRLD Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	RLD2	IIR filter channel 2 coefficient shift amount reload bit 0: Does not reload. 1: Reloads.
1	RLD1	IIR filter channel 1 coefficient shift amount reload bit 0: Does not reload. 1: Reloads.
0	RLD0	IIR filter channel 0 coefficient shift amount reload bit 0: Does not reload. 1: Reloads.

Note: For details, see **Section 25.4.9, IIR Filters**.

Setting a bit to 1 reloads the corresponding channel. The bits are automatically reset to 0 after being set to 1. It is inhibited to write a 0. The bits are always read as 0.

Do not write 1 to the bits while the IIR filters are running.

25.3.2.129 EMU3n IIR Filter Completion Flag Register (EMU3nIIRSTAT)

Access: Readable in 8-bit units.

Address: <EMU3n_base> + $05CE_H$

Value after reset: 00_H

Category: IIR filter

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	STAT2	STAT1	STAT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 25.151 EMU3nIIRSTAT Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	STAT2	IIR filter channel 2 completion flag bit 0: Waits for filtering to be completed 1: Filtering completed
1	STAT1	IIR filter channel 1 completion flag bit 0: Waits for filtering to be completed 1: Filtering completed
0	STAT0	IIR filter channel 0 completion flag bit 0: Waits for filtering to be completed 1: Filtering completed

Note: For details, see **Section 25.4.9, IIR Filters**.

This register is used to monitor the completion of IIR filtering.

The bits are set to 1 when the processing of the corresponding IIR filter channel becomes completed.

The bits are cleared by setting the corresponding bits of the EMU3nIIRSTATC register.

25.3.2.130 EMU3n IIR Filter Completion Flag Clear Register (EMU3nIIRSTATC)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $05CF_H$

Value after reset: 00_H

Category: IIR filter

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	STATC2	STATC1	STATC0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 25.152 EMU3nIIRSTATC Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	STATC2	IIR filter channel 2 completion flag clear bit Setting this bit to 1 resets the STAT2 bit of the EMU3nIIRSTAT register.
1	STATC1	IIR filter channel 1 completion flag clear bit Setting this bit to 1 resets the STAT1 bit of the EMU3nIIRSTAT register.
0	STATC0	IIR filter channel 0 completion flag clear bit Setting this bit to 1 resets the STAT0 bit of the EMU3nIIRSTAT register.

Note: For details, see **Section 25.4.9, IIR Filters**.

The bits are automatically reset to 0 after being set to 1.

It is inhibited to write a 0. The bits are always read as 0.

25.3.2.131 EMU3n IIR Filter Coefficient k Value Register (EMU3nIIRCOEFFk) (k = 0 to 5)

Access: Readable/writable in 16-bit units.

Address: EMU3nIIRCOEFF0: <EMU3n_base> + $05D0_H$

EMU3nIIRCOEFF1: <EMU3n_base> + $05D2_H$

EMU3nIIRCOEFF2: <EMU3n_base> + $05D4_H$

EMU3nIIRCOEFF3: <EMU3n_base> + $05D6_H$

EMU3nIIRCOEFF4: <EMU3n_base> + $05D8_H$

EMU3nIIRCOEFF5: <EMU3n_base> + $05DA_H$

Value after reset: 0000_H

Category: IIR filter

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.153 EMU3nIIRCOEFFk Register Contents

Bit Position	Bit	Function
15 to 0	DATA	IIR filter coefficient input data bits (signed integer) (b15: Signed part)

Note: For details, see **Section 25.4.9, IIR Filters**.

25.3.2.132 EMU3n IIR Filter Shift Amount Value Register (EMU3nIIRSHIFT)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $05DC_H$

Value after reset: 00_H

Category: IIR filter

Bit	7	6	5	4	3	2	1	0
	—	—	—	DATA (unsigned)				
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 25.154 EMU3nIIRSHIFT Register Contents

Bit Position	Bit	Function
7 to 5	—	These bits are read as 0. The write value should be 0.
4 to 0	DATA	IIR filter shift amount input data bits (unsigned integer) These bits specify the amount of right shift by which the data is to be arithmetically right shifted in the last stage of IIR filtering processing. b4 b3 b2 b1 b0 0 0 0 0 0: No Shifts 0 0 0 0 1: 1-bit right shift 0 0 0 1 0: 2-bit right shift 0 0 0 1 1: 3-bit right shift 0 0 1 0 0: 4-bit right shift 0 0 1 0 1: 5-bit right shift 0 0 1 1 0: 6-bit right shift 0 0 1 1 1: 7-bit right shift 0 1 0 0 0: 8-bit right shift 0 1 0 0 1: 9-bit right shift 0 1 0 1 0: 10-bit right shift 0 1 0 1 1: 11-bit right shift 0 1 1 0 0: 12-bit right shift 0 1 1 0 1: 13-bit right shift 0 1 1 1 0: 14-bit right shift 0 1 1 1 1: 15-bit right shift 1 0 0 0 0: 16-bit right shift 1 0 0 0 1: 17-bit right shift 1 0 0 1 0: 18-bit right shift 1 0 0 1 1: 19-bit right shift 1 0 1 0 0: 20-bit right shift 1 0 1 0 1: 21-bit right shift 1 0 1 1 0: 22-bit right shift 1 0 1 1 1: 23-bit right shift Other than above: No shifts

Note: For details, see **Section 25.4.9, IIR Filters**.

25.3.2.133 EMU3n IIR Filter Channel k Coefficient m Monitor Register (EMU3nIIRCOEFFMmk) (m = 0 to 5) (k = 0 to 2)

Access: Readable in 16-bit units.

Address: EMU3nIIRCOEFFM00: <EMU3n_base> + $05E0_H$
EMU3nIIRCOEFFM10: <EMU3n_base> + $05E2_H$
EMU3nIIRCOEFFM20: <EMU3n_base> + $05E4_H$
EMU3nIIRCOEFFM30: <EMU3n_base> + $05E6_H$
EMU3nIIRCOEFFM40: <EMU3n_base> + $05E8_H$
EMU3nIIRCOEFFM50: <EMU3n_base> + $05EA_H$
EMU3nIIRCOEFFM01: <EMU3n_base> + $05F0_H$
EMU3nIIRCOEFFM11: <EMU3n_base> + $05F2_H$
EMU3nIIRCOEFFM21: <EMU3n_base> + $05F4_H$
EMU3nIIRCOEFFM31: <EMU3n_base> + $05F6_H$
EMU3nIIRCOEFFM41: <EMU3n_base> + $05F8_H$
EMU3nIIRCOEFFM51: <EMU3n_base> + $05FA_H$
EMU3nIIRCOEFFM02: <EMU3n_base> + 0600_H
EMU3nIIRCOEFFM12: <EMU3n_base> + 0602_H
EMU3nIIRCOEFFM22: <EMU3n_base> + 0604_H
EMU3nIIRCOEFFM32: <EMU3n_base> + 0606_H
EMU3nIIRCOEFFM42: <EMU3n_base> + 0608_H
EMU3nIIRCOEFFM52: <EMU3n_base> + $060A_H$

Value after reset: 0000_H

Category: IIR filter

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.155 EMU3nIIRCOEFFMmk Register Contents

Bit Position	Bit	Function
15 to 0	DATA	IIR filter coefficient monitor data bits (signed integer) (b15: Signed part) Loaded with the IIR filter coefficients for IIR filter channels for monitoring of each channel.

Note: For details, see **Section 25.4.9, IIR Filters**.

25.3.2.134 EMU3n IIR Filter Channel k Shift Amount Monitor Register (EMU3nIIRSHIFTMk) (k = 0 to 2)

Access: Readable in 8-bit units.

Address: EMU3nIIRSHIFTM0: <EMU3n_base> + $05EC_H$

EMU3nIIRSHIFTM1: <EMU3n_base> + $05FC_H$

EMU3nIIRSHIFTM2: <EMU3n_base> + $060C_H$

Value after reset: 00_H

Category: IIR filter

Bit	7	6	5	4	3	2	1	0
	—	—	—	DATA (unsigned)				
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 25.156 EMU3nIIRSHIFTMk Register Contents

Bit Position	Bit	Function
7 to 5	—	These bits are read as 0. The write value should be 0.
4 to 0	DATA	IIR filter shift amount monitor data bits (unsigned integer) Loaded with the IIR filter shift amount for monitoring of each channel.

Note: For details, see **Section 25.4.9 IIR Filters**.

25.3.2.135 EMU3n IIR Filter Channel k Data Software Input Register (EMU3nIIRSFTDATk) (k = 0 to 2)

Access: Readable/writable in 32-bit units.

Address: EMU3nIIRSFTDAT0: <EMU3n_base> + 0620_H

EMU3nIIRSFTDAT1: <EMU3n_base> + 0624_H

EMU3nIIRSFTDAT2: <EMU3n_base> + 0628_H

Value after reset: $0000\ 0000_H$

Category: IIR filter

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DATA (signed)							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.157 EMU3nIIRSFTDATk Register Contents

Bit Position	Bit	Function
31 to 24	—	These bits are read as 0. The write value should be 0.
23 to 0	DATA	Data bits (signed integer) (b23: Signed part) Set the software-input IIR filter input data.

Note: For details, see **Section 25.4.9 IIR Filters**.

25.3.2.136 EMU3n IIR Filter Channel k Delay 1 Data Register (EMU3nIIRZN1DATk) (k = 0 to 2)

Access: Readable/writable in 32-bit units.

Address: EMU3nIIRZN1DAT0: <EMU3n_base> + 0630_H

EMU3nIIRZN1DAT1: <EMU3n_base> + 0638_H

EMU3nIIRZN1DAT2: <EMU3n_base> + 0640_H

Value after reset: $0000\ 0000_H$

Category: IIR filter

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DATA (signed)							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.158 EMU3nIIRZN1DATk Register Contents

Bit Position	Bit	Function
31 to 24	—	These bits are read as 0. The write value should be 0.
23 to 0	DATA	Data bits (signed integer) (b23: Signed part)

Note: For details, see **Section 25.4.9, IIR Filters**.

The IIR filter delay data must not be rewritten while the IIR filters are running.

25.3.2.137 EMU3n IIR Filter Channel k Delay 2 Data Register (EMU3nIIRZN2DATk) (k = 0 to 2)

Access: Readable/writable in 32-bit units.

Address: EMU3nIIRZN2DAT0: <EMU3n_base> + 0634_H

EMU3nIIRZN2DAT1: <EMU3n_base> + $063C_H$

EMU3nIIRZN2DAT2: <EMU3n_base> + 0644_H

Value after reset: $0000\ 0000_H$

Category: IIR filter

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DATA (signed)							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.159 EMU3nIIRZN2DATk Register Contents

Bit Position	Bit	Function
31 to 24	—	These bits are read as 0. The write value should be 0.
23 to 0	DATA	Data bits (signed integer) (b23: Signed part)

Note: For details, see **Section 25.4.9, IIR Filters**.

The IIR filter delay data must not be rewritten while the IIR filters are running.

25.3.2.138 EMU3n IIR Filter Channel k Output Data Register (EMU3nIIROUTDATk) (k = 0 to 2)

Access: Readable in 32-bit units.

Address: EMU3nIIROUTDAT0: <EMU3n_base> + 0650_H

EMU3nIIROUTDAT1: <EMU3n_base> + 0654_H

EMU3nIIROUTDAT2: <EMU3n_base> + 0658_H

Value after reset: $0000\ 0000_H$

Category: IIR filter

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DATA (signed)							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.160 EMU3nIIROUTDATk Register Contents

Bit Position	Bit	Function
31 to 24	—	These bits are read as 0. The write value should be 0.
23 to 0	DATA	Data bits (signed integer) (b23: Signed part)

Note: For details, see **Section 25.4.9, IIR Filters**.

25.3.2.139 EMU3n Checking Buffer Control Register (EMU3nCBCTR0)

Access: Readable/writable in 8-bit units.

Address: <EMU3n_base> + $06C4_H$

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	CBMON	CBEN1	CBEN0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 25.161 EMU3nCBCTR0 Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	CBMON	Buffering status bit 0: Buffering is completed or in idle state. 1: Waiting for buffering to be completed.
1	CBEN1	Buffering enable bit 1 0: Buffering is enabled by the CBEN0 bit. 1: Buffering is always enabled.
0	CBEN0	Buffering enable bit 0 Setting this bit to 1 enables buffering. The bit is automatically reset to 0 after being set to 1. Writing a 0 is invalid. The bit is always read as 0.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

The values that are to be buffered are listed below.

- EMU3n A/D data k register (EMU3nADk) (k = 0, 1, 2)
- EMU3n resolver angle monitor register (EMU3nTHTRESFIXIN)
- EMU3n d-axis current value register (EMU3nIDFIX)
- EMU3n q-axis current value register (EMU3nIQFIX)
- U/V/W-phase PWM compare value pin output values
- U/V/W-phase rectangle wave pin output values
- U/V/W-phase independent rectangle IP1 wave pin output values

CBEN0 Bit

If the CBEN0 bit is set to 1, buffering to the checking buffer register is performed only once at the timing, out of the timings enabled by the EMU3nCBTIM register, that is associated with the first event that occurred. To perform buffering again, set the CBEN0 bit to 1 in each case. When specifying two or more timings in the EMU3nCBTIM register, set the CBEN0 bit to 1 for each of the timings.

CBEN1 Bit

If the CBEN1 bit is set to 1, buffering to the checking buffer register is performed each time a timing enabled by the EMU3nCBTIM register occurs.

CBMON Bit

If buffering is enabled by the CBEN0 bit, this bit is set to 1 and held from the time the CBEN0 bit is set to 1 till the buffering becomes completed. If buffering is enabled by the CBEN1 bit, this bit is set to and held at 1 from the time the CBEN1 bit is set to 1 till the CBEN1 bit is set to 0.

25.3.2.140 EMU3n Checking Buffer Timing Select Register (EMU3nCBTIM)

Access: Readable/writable in 16-bit units.

Address: <EMU3n_base> + $06C6_H$

Value after reset: 0000_H

Category: General/common

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	IRECWBT	IRECVBT	IRECUBT	—	CMP0BT	—	—	—	—	—	—	PWMBT	—	INBT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R	R/W	R	R	R	R	R	R	R/W	R	R/W

Table 25.162 EMU3nCBTIM Register Contents

Bit Position	Bit	Function
15, 14	—	These bits are read as 0. The write value should be 0.
13	IRECWBT	Buffering timing setting bit (independent rectangle IP1 W-phase angle compare 0 match detection) 0: Disables buffering. 1: Enables buffering.
12	IRECVBT	Buffering timing setting bit (independent rectangle IP1 V-phase angle compare 0 match detection) 0: Disables buffering. 1: Enables buffering.
11	IRECUBT	Buffering timing setting bit (independent rectangle IP1 U-phase angle compare 0 match detection) 0: Disables buffering. 1: Enables buffering.
10	—	This bits is read as 0. The write value should be 0.
9	CMP0BT	Buffering timing setting bit (angle compare 0 match detection) 0: Disables buffering. 1: Enables buffering.
8 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMBT	Buffering timing setting bit (PWM IP complete) 0: Disables buffering. 1: Enables buffering.
1	—	This bits is read as 0. The write value should be 0.
0	INBT	Buffering timing setting bit (input IP complete) 0: Disables buffering. 1: Enables buffering.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.141 EMU3n A/D Data k Checking Buffer Register (EMU3nCBADk) (k = 0, 1, 2)

Access: Readable in 16-bit units.

Address: EMU3nCBAD0: <EMU3n_base> + $06C8_H$

EMU3nCBAD1: <EMU3n_base> + $06CA_H$

EMU3nCBAD2: <EMU3n_base> + $06CC_H$

Value after reset: 0000_H

Category: Checking buffer

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.163 EMU3nCBADk Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Loaded with the results of EMU3nADk register buffering.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.142 EMU3n Resolver Angle Checking Buffer Register (EMU3nCBTHTRESFIXIN)

Access: Readable in 16-bit units.

Address: <EMU3n_base> + $06CE_H$

Value after reset: 0000_H

Category: Checking buffer

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DATA (unsigned)											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.164 EMU3nCBTHTRESFIXIN Register Contents

Bit Position	Bit	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data bits (unsigned integer) Loaded with the results of EMU3nTHTRESFIXIN register buffering.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.143 EMU3n d-Axis Current Value Checking Buffer Register (EMU3nCBIDFIX)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $06D0_H$

Value after reset: $0000\ 0000_H$

Category: Checking buffer

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.165 EMU3nCBIDFIX Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer) (b31: Signed part) Loaded with the results of EMU3nIDFIX register buffering.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.144 EMU3n q-Axis Current Value Checking Buffer Register (EMU3nCBIQFIX)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $06D4_H$

Value after reset: $0000\ 0000_H$

Category: Checking buffer

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (signed)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.166 EMU3nCBIQFIX Register Contents

Bit Position	Bit	Function
31 to 0	DATA	Data bits (signed integer)(b31: Signed part) Loaded with the results of EMU3nIQFIX register buffering.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.145 EMU3n U-Phase PWM Compare Value Checking Buffer Register (EMU3nCBPWMUIP)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $06D8_H$

Value after reset: $0000\ 0000_H$

Category: Checking buffer

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DATA (unsigned)	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.167 EMU3nCBPWMUIP Register Contents

Bit Position	Bit	Function
31 to 18	—	These bits are read as 0. The write value should be 0.
17 to 0	DATA	Data bits (unsigned integer) Loaded with the results of buffering the U-phase PWM compare value pin output value.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.146 EMU3n V-Phase PWM Compare Value Checking Buffer Register (EMU3nCBPWMVIP)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $06DC_H$

Value after reset: $0000\ 0000_H$

Category: Checking buffer

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DATA (unsigned)	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.168 EMU3nCBPWMVIP Register Contents

Bit Position	Bit	Function
31 to 18	—	These bits are read as 0. The write value should be 0.
17 to 0	DATA	Data bits (unsigned integer) Loaded with the results of buffering the V-phase PWM compare value pin output value.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.147 EMU3n W-Phase PWM Compare Value Checking Buffer Register (EMU3nCBPWMWIP)

Access: Readable in 32-bit units.

Address: <EMU3n_base> + $06E0_H$

Value after reset: $0000\ 0000_H$

Category: Checking buffer

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DATA (unsigned)	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.169 EMU3nCBPWMWIP Register Contents

Bit Position	Bit	Function
31 to 18	—	These bits are read as 0. The write value should be 0.
17 to 0	DATA	Data bits (unsigned integer) Loaded with the results of buffering the W-phase PWM compare value pin output value.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.148 EMU3n Rectangle Pattern Value Checking Buffer Register (EMU3nCBBREC)

Access: Readable in 8-bit units.

Address: <EMU3n_base> + $06E4_H$

Value after reset: 00_H

Category: Checking buffer

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WPTN	VPTN	UPTN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 25.170 EMU3nCBBREC Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WPTN	rectangle W-phase output pattern bit Loaded with the result of buffering the rectangle W-phase output pattern value.
1	VPTN	rectangle V-phase output pattern bit Loaded with the result of buffering the rectangle V-phase output pattern value.
0	UPTN	rectangle U-phase output pattern bit Loaded with the result of buffering the rectangle U-phase output pattern value.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.149 EMU3n Independent Rectangle IP 1 Pattern Value Checking Buffer Register (EMU3nCBIREC)

Access: Readable in 8-bit units.

Address: <EMU3n_base> + $06E5_H$

Value after reset: 00_H

Category: Checking buffer

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WPTN	VPTN	UPTN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 25.171 EMU3nCBIREC Register Contents

Bit Position	Bit	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WPTN	Independent rectangle IP1 W-phase output pattern bit Loaded with the result of buffering the independent rectangle IP1 W-phase output pattern value.
1	VPTN	Independent rectangle IP1 V-phase output pattern bit Loaded with the result of buffering the independent rectangle IP1 V-phase output pattern value.
0	UPTN	Independent rectangle IP1 U-phase output pattern bit Loaded with the result of buffering the independent rectangle IP1 U-phase output pattern value.

Note: For details, see **Section 25.4.14, Fault Detection Function**.

25.3.2.150 EMU3n Data Set WBk Transfer Trigger Register (EMU3nDATSETWBk) (k = 0 to 2)

Access: Readable/writable in 8-bit units.

Address: EMU3nDATSETWB0: <EMU3n_base> + 0700_H

EMU3nDATSETWB1: <EMU3n_base> + 0704_H

EMU3nDATSETWB2: <EMU3n_base> + 0708_H

Value after reset: 00_H

Category: Handshaking register

Bit	7	6	5	4	3	2	1	0
	—	—	TRG5	TRG4	TRG3	TRG2	TRG1	TRG0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.172 EMU3nDATSETWBk Register Contents

Bit Position	Bit	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5	TRG5	Transfer trigger bit Triggers a bulk transfer of data from the data set 5WRITEm registers EMU3nDATSETW5m (m: 0 to 3) to the intermediate registers corresponding to the data set 5READm registers EMU3nDATSETR5m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
4	TRG4	Transfer trigger bit Triggers a bulk transfer of data from the data set 4WRITEm registers EMU3nDATSETW4m (m: 0 to 3) to the intermediate registers corresponding to the data set 4READm registers EMU3nDATSETR4m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
3	TRG3	Transfer trigger bit Triggers a bulk transfer of data from the data set 3WRITEm registers EMU3nDATSETW3m (m: 0 to 3) to the intermediate registers corresponding to the data set 3READm registers EMU3nDATSETR3m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
2	TRG2	Transfer trigger bit Triggers a bulk transfer of data from the data set 2WRITEm registers EMU3nDATSETW2m (m: 0 to 3) to the intermediate registers corresponding to the data set 2READm registers EMU3nDATSETR2m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
1	TRG1	Transfer trigger bit Triggers a bulk transfer of data from the data set 1WRITEm registers EMU3nDATSETW1m (m: 0 to 3) to the intermediate registers corresponding to the data set 1READm registers EMU3nDATSETR1m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
0	TRG0	Transfer trigger bit Triggers a bulk transfer of data from the data set 0WRITEm registers EMU3nDATSETW0m (m: 0 to 3) to the intermediate registers corresponding to the data set 0READm registers EMU3nDATSETR0m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.

Note: For details, see **Section 25.4.13, Asynchronous Data Passing Function**.

Setting these bits to 1 triggers a bulk transfer. The bits are automatically reset to 0 after being set to 1. Writing a 0 is invalid. The bits are always read as 0.

25.3.2.151 EMU3n Data Set BRk Transfer Trigger Register (EMU3nDATSETBRk) (k = 0 to 2)

Access: Readable/writable in 8-bit units.

Address: EMU3nDATSETBR0: <EMU3n_base> + 0710_H

EMU3nDATSETBR1: <EMU3n_base> + 0714_H

EMU3nDATSETBR2: <EMU3n_base> + 0718_H

Value after reset: 00_H

Category: Handshaking register

Bit	7	6	5	4	3	2	1	0
	—	—	TRG5	TRG4	TRG3	TRG2	TRG1	TRG0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.173 EMU3nDATSETBRk Register Contents

Bit Position	Bit	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5	TRG5	Transfer trigger bit Triggers a bulk transfer of data from the intermediate registers corresponding to the data set 5WRITEm registers EMU3nDATSETW5m (m: 0 to 3) to the data set 5READm registers EMU3nDATSETR5m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
4	TRG4	Transfer trigger bit Triggers a bulk transfer of data from the intermediate registers corresponding to the data set 4WRITEm registers EMU3nDATSETW4m (m: 0 to 3) to the data set 4READm registers EMU3nDATSETR4m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
3	TRG3	Transfer trigger bit Triggers a bulk transfer of data from the intermediate registers corresponding to the data set 3WRITEm registers EMU3nDATSETW3m (m: 0 to 3) to the data set 3READm registers EMU3nDATSETR3m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
2	TRG2	Transfer trigger bit Triggers a bulk transfer of data from the intermediate registers corresponding to the data set 2WRITEm registers EMU3nDATSETW2m (m: 0 to 3) to the data set 2READm registers EMU3nDATSETR2m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
1	TRG1	Transfer trigger bit Triggers a bulk transfer of data from the intermediate registers corresponding to the data set 1WRITEm registers EMU3nDATSETW1m (m: 0 to 3) to the data set 1READm registers EMU3nDATSETR1m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.
0	TRG0	Transfer trigger bit Triggers a bulk transfer of data from the intermediate registers corresponding to the data set 0WRITEm registers EMU3nDATSETW0m (m: 0 to 3) to the data set 0READm registers EMU3nDATSETR0m (m: 0 to 3). 0: Triggers no transfer. 1: Triggers bulk transfer.

Note: For details, see **Section 25.4.13 Asynchronous Data Passing Function**.

Setting these bits to 1 triggers a bulk transfer. The bits are automatically reset to 0 after being set to 1. Writing a 0 is invalid. The bits are always read as 0.

25.3.2.152 EMU3n Data Set kWRITEm Register (EMU3nDATSETWkm) (k = 0 to 5) (m = 0 to 3)

Access: Readable/writable in 16, and 32-bit units.

Address:

EMU3nDATSETW00: <EMU3n_base> + 0720_H,	EMU3nDATSETW01: <EMU3n_base> + 0722_H
EMU3nDATSETW02: <EMU3n_base> + 0724_H,	EMU3nDATSETW03: <EMU3n_base> + 0726_H
EMU3nDATSETW10: <EMU3n_base> + 0728_H,	EMU3nDATSETW11: <EMU3n_base> + $072A_H$
EMU3nDATSETW12: <EMU3n_base> + $072C_H$,	EMU3nDATSETW13: <EMU3n_base> + $072E_H$
EMU3nDATSETW20: <EMU3n_base> + 0730_H,	EMU3nDATSETW21: <EMU3n_base> + 0732_H
EMU3nDATSETW22: <EMU3n_base> + 0734_H,	EMU3nDATSETW23: <EMU3n_base> + 0736_H
EMU3nDATSETW30: <EMU3n_base> + 0738_H,	EMU3nDATSETW31: <EMU3n_base> + $073A_H$
EMU3nDATSETW32: <EMU3n_base> + $073C_H$,	EMU3nDATSETW33: <EMU3n_base> + $073E_H$
EMU3nDATSETW40: <EMU3n_base> + 0740_H,	EMU3nDATSETW41: <EMU3n_base> + 0742_H
EMU3nDATSETW42: <EMU3n_base> + 0744_H,	EMU3nDATSETW43: <EMU3n_base> + 0746_H
EMU3nDATSETW50: <EMU3n_base> + 0748_H,	EMU3nDATSETW51: <EMU3n_base> + $074A_H$
EMU3nDATSETW52: <EMU3n_base> + $074C_H$,	EMU3nDATSETW53: <EMU3n_base> + $074E_H$

Value after reset: 0000_H

Category: Handshaking register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.174 EMU3nDATSETWkm Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer) Set write data.

Note: For details, see **Section 25.4.13, Asynchronous Data Passing Function**.

The EMU3nDATSETWk0 register and EMU3nDATSETWk1 register can be handled as 16-bit data registers and combined into a 32-bit data register and used for data reads and writes. The EMU3nDATSETWk0 register is assumed to hold the lower-order 16 bits and the EMU3nDATSETWk1 register to hold the higher-order 16 bits.

The EMU3nDATSETWk2 register and EMU3n0DATSETWk3 register can be handled as 16-bit data registers and combined into a 32-bit data register and used for data reads and writes. The EMU3nDATSETWk2 register is assumed to hold the lower-order 16 bits and the EMU3nDATSETWk3 register to hold the higher-order 16 bits.

The following symbols and addresses should be used when accessing 32-bit data:

EMU3nDATSETW00W: <EMU3n_base> + 0720_H,	EMU3nDATSETW02W: <EMU3n_base> + 0724_H,
EMU3nDATSETW10W: <EMU3n_base> + 0728_H,	EMU3nDATSETW12W: <EMU3n_base> + $072C_H$,
EMU3nDATSETW20W: <EMU3n_base> + 0730_H,	EMU3nDATSETW22W: <EMU3n_base> + 0734_H,
EMU3nDATSETW30W: <EMU3n_base> + 0738_H,	EMU3nDATSETW32W: <EMU3n_base> + $073C_H$,
EMU3nDATSETW40W: <EMU3n_base> + 0740_H,	EMU3nDATSETW42W: <EMU3n_base> + 0744_H,
EMU3nDATSETW50W: <EMU3n_base> + 0748_H,	EMU3nDATSETW52W: <EMU3n_base> + $074C_H$

25.3.2.153 EMU3n Data Set kREADm Register (EMU3nDATSETRkm) (k = 0 to 5) (m = 0 to 3)

Access: Readable in 16, and 32-bit units.

Address:

EMU3nDATSETR00: <EMU3n_base> + 0760_H,	EMU3nDATSETR01: <EMU3n_base> + 0762_H
EMU3nDATSETR02: <EMU3n_base> + 0764_H,	EMU3nDATSETR03: <EMU3n_base> + 0766_H
EMU3nDATSETR10: <EMU3n_base> + 0768_H,	EMU3nDATSETR11: <EMU3n_base> + $076A_H$
EMU3nDATSETR12: <EMU3n_base> + $076C_H$,	EMU3nDATSETR13: <EMU3n_base> + $076E_H$
EMU3nDATSETR20: <EMU3n_base> + 0770_H,	EMU3nDATSETR21: <EMU3n_base> + 0772_H
EMU3nDATSETR22: <EMU3n_base> + 0774_H,	EMU3nDATSETR23: <EMU3n_base> + 0776_H
EMU3nDATSETR30: <EMU3n_base> + 0778_H,	EMU3nDATSETR31: <EMU3n_base> + $077A_H$
EMU3nDATSETR32: <EMU3n_base> + $077C_H$,	EMU3nDATSETR33: <EMU3n_base> + $077E_H$
EMU3nDATSETR40: <EMU3n_base> + 0780_H,	EMU3nDATSETR41: <EMU3n_base> + 0782_H
EMU3nDATSETR42: <EMU3n_base> + 0784_H,	EMU3nDATSETR43: <EMU3n_base> + 0786_H
EMU3nDATSETR50: <EMU3n_base> + 0788_H,	EMU3nDATSETR51: <EMU3n_base> + $078A_H$
EMU3nDATSETR52: <EMU3n_base> + $078C_H$,	EMU3nDATSETR53: <EMU3n_base> + $078E_H$

Value after reset: 0000_H

Category: Handshaking register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA (unsigned)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.175 EMU3nDATSETRkm Register Contents

Bit Position	Bit	Function
15 to 0	DATA	Data bits (unsigned integer) Loaded with read data.

Note: For details, see **Section 25.4.13, Asynchronous Data Passing Function**.

The EMU3nDATSETRk0 register and EMU3nDATSETRk1 register can be handled as 16-bit data registers and combined into a 32-bit data register and used for data reads. The EMU3nDATSETRk0 register is assumed to hold the lower-order 16 bits and the EMU3nDATSETRk1 register to hold the higher-order 16 bits.

The EMU3nDATSETRk2 register and EMU3nDATSETRk3 register can be handled as 16-bit data registers and combined into a 32-bit data register and used for data reads. The EMU3nDATSETRk2 register is assumed to hold the lower-order 16 bits and the EMU3nDATSETRk3 register to hold the higher-order 16 bits.

The following symbols and addresses should be used when accessing 32-bit data:

EMU3nDATSETR00W: <EMU3n_base> + 0760_H,	EMU3nDATSETR02W: <EMU3n_base> + 0764_H,
EMU3nDATSETR10W: <EMU3n_base> + 0768_H,	EMU3nDATSETR12W: <EMU3n_base> + $076C_H$,
EMU3nDATSETR20W: <EMU3n_base> + 0770_H,	EMU3nDATSETR22W: <EMU3n_base> + 0774_H,
EMU3nDATSETR30W: <EMU3n_base> + 0778_H,	EMU3nDATSETR32W: <EMU3n_base> + $077C_H$,
EMU3nDATSETR40W: <EMU3n_base> + 0780_H,	EMU3nDATSETR42W: <EMU3n_base> + 0784_H,
EMU3nDATSETR50W: <EMU3n_base> + 0788_H,	EMU3nDATSETR52W: <EMU3n_base> + $078C_H$

25.3.2.154 SubCPU Startup Register (EMU3CPUINIT)

Access: Readable/writable in 8-bit units.

Address: <EMU_base> + 0000 2000$_H$

Value after reset: 00$_H$

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CPINIT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.176 EMU3CPUINIT Register Contents

Bit Position	Bit	Function
31 to 1	—	These bits are read as 0. The write value should be 0.
0	CPINIT	0: Stops SubCPU operation. 1: Starts SubCPU operation (starts fetching)

Note 1. This bit should not be cleared to after having been set to 1.

This register is used to start the SubCPUs of the EMU3. To start the EMU3 after releasing the reset, access this register from the main CPU.

25.3.2.155 ADC Select Register (EMU3ADCSEL)

Access: Readable/writable in 8-bit units.

Address: <EMU_base> + 0000 2040_H

Value after reset: 00_H

Category: General/common

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SEL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.177 EMU3ADCSEL Register Contents

Bit position	Bit	Function
31 to 1	—	These bits are read as 0. The write value should be 0.
0	SEL	Selects the A/D converter data to be taken into the EMU31. 0: ADCC1 data 1: ADCC2 data

Accessing this register from the SubCPU's dedicated bus (base address is FC00 0000_H) is disallowed.

Like accesses form the CPU1 and CPU2, the register must be accessed from the SubCPUs via the base address of FF70 0000_H.

The setting made in this register must not be changed after the EMU3 is started.

25.4 Functions

25.4.1 Arithmetic Block Descriptions

1. Each variable is suffixed with the information indicating whether it is signed or unsigned, and the valid bit length. _s denotes the sign and _u denotes unsigned. The following number denotes the bit length. For a signed variable, the most significant bit is the sign bit.
   Example
   Ia_v_Lo_fix_s[31:0] represents data bits 0 to 31 of the variable Ia_v_Lo_fix. Bit 31 contains sign data.

2. For an arithmetic expression, an overflow occurs if it yields a value that exceeds the number of available bits.
   Example
   A: $7FFF_H$ B: $7FFF_H$
   C_s[15:0] ← A_s[15:0] + B_s[15:0]
   Note: The INIPOF bit of the EMU3nOFMON register is set to 1 if an overflow occurs during the operation in the input IP and the PIIPOF bit is set to 1 if an overflow occurs during the operation in the PI control IP, the PWMIPOF bit is set to 1 if an overflow occurs during the operation in the PWM IP.

3. If a division or right shift follows immediately after an arithmetic expression, a bit extension occurs and the divided or shifted value is stored.
   Example
   C_s[15:0] ← (A_s[15:0] × B_s[15:0]) >> 10
   The results of (A_s[15:0] × B_s[15:0]) is extended to 32 bits before the 16 bits that are shifted 10 bits to the right are stored in C_s[15:0].
   (Similarly, if all of A, B, and C are 32 bits long, the results of A_s[31:0] × B_s[31:0] are extended to 64 bits before the 32 bits that are shifted 10 bits to the right are stored.)

   D_s[15:0] ← (A_s[15:0] × B_s[15:0]) / C_s[15:0]
   32 bits of the results of D(A_s[15:0] × B_s[15:0]) are divided by C_s[15:0] and the results of the division are stored in D_s[15:0].

4. An arithmetic expression is bit-extended according to the number of bits of the location for storing the results.
   Example
   D_s[31:0] ← C_s[31:0] + A_s[15:0] × B_s[15:0]
   The results of (A_s[15:0] × B_s[15:0]) are extended to 32 bits and then added to C_s[31:0] as they are.
   Note: An overflow occurs if the result of adding the value to C_s[31:0] exceeds 32 bits.

   Example
   C_s[15:0] ← A_s[11:0] + B_s[11:0]
   The results of (A_s[11:0] + B_s[11:0]) are sign-extended to 16 bits and stored in C_s[15:0].

5. A division is processed as follows:
   In division, the fractional part is rounded down.

   Examples
   5 / 2 = 2
   – 5 / 2 = –2
   (1) 16 bits ← 32 bits divided by 16 bits
   0 / 0 = 0
   Negative number / 0 = 8000_H
   Positive number / 0 = $7FFF_H$

   (2) 32 bits ← 32 bits divided by 16 bits
   0 / 0 = 0
   Negative number / 0 = $8000\ 0000_H$
   Positive number / 0 = $7FFF\ FFFF_H$

   (3) 16 bits ← 32 bits divided by 16 bits
   0 / 0 = 0
   If $7FFF_H \leq$ results, then results ← $7FFF_H$
   If $8000_H \geq$ results, then results ← 8000_H

6. The variables represented in the lowercase alphanumeric character in this section is basically preset in a register, bit(s), or cache of the EMU3. It cannot be set directly by the user. Unless specifically specified, the initial value of the internal registers of the EMU3 is 0.

7. The left column of a table delineated by a double line contains the target that is subjected to the condition in question and the right column contains the location where the result is to be stored.
   A table having a double line in this section (see the table below) shows the situation such that when the target A satisfies the condition B, the result E is stored in D or that when the target A satisfies the condition C, the result F is stored in D.

   Example

Target A subjected to a condition	Result in destination D
Condition B	Result E
Condition C	Result F

8. There are sections in which modules appear in C-like code. The example given below illustrates this case.

   Example) PWM IP Func(pwm2) electrical angle offset processing

[pwm2] Electrical angle offset processing

Explanation	Adds the offset value to the angle value and outputs the results for phase adjustment.	
Arguments	*etht1i*: Input angle value (EMU3nTHTESEL) *etht2i*: Input angle value (EMU3nTHTEFIX)	**ethto*: Output angle value (EMU3nTHTEPWM)
Related registers	EMU3nPWMCTR.FLININIP: Electrical angle select bits (0,1,2) EMU3nTHTFORESFT: PWM IP electrical angle soft input register (uint12_t) EMU3nPHI: PWM IP electrical angle offset register (uint16_t) EMU3nGTHT: PWM IP electrical angle adjustment coefficient register (int16_t)	

Processing	void pwm2 (uint12_t *etht1i*, uint12_t *etht2i*,
	uint12_t **ethto*) {
	
	if(EMU3nPWMCTR.FLININIP == 0) // Select electrical angle soft input.
	**ethto* = (((EMU3nTHTFORESFT + EMU3nPHI) & 0x0FFF) * EMU3nGTHT) >> 8 ;
	else if (EMU3nPWMCTR.FLININIP == 1) // Select input from input IP.
	**ethto* = (((*etht1i* + EMU3nPHI) & 0x0FFF) * EMU3nGTHT) >> 8 ;
	else if (EMU3nPWMCTR.FLININIP == 2) // Select input from angle generation IP.
	**ethto* = (((*etht2i* + EMU3nPHI) & 0x0FFF) * EMU3nGTHT) >> 8 ;
	}	

The function name “pwm2” corresponds to a module.
Variables “uint12_t etht1i, uint12_t etht2i, uint12_t *ethto” correspond to input/output signals of the module.
uintXX_t represents an unsigned signal with a bit width of [XX-1:0] and

intYY_t represents a signed signal with a bit width of [YY-1:0].
“register_name.bit” represents a bit name of the register.

25.4.2 Angle Generation IP

The angle generation IP generates electrical angle data from a resolver angle and performs resolver angle compare match detection and electrical angle compare match detection processing.

The angle generation IP is started every time the resolver angle which is selected from the R/D converter or EMU3nRESTHSFT register is updated. Any startup request issued while the angle generation IP is active is invalid.

Figure 25.4 shows the outline of the angle generation IP processing and **Figure 25.5** shows an example of angle generation.

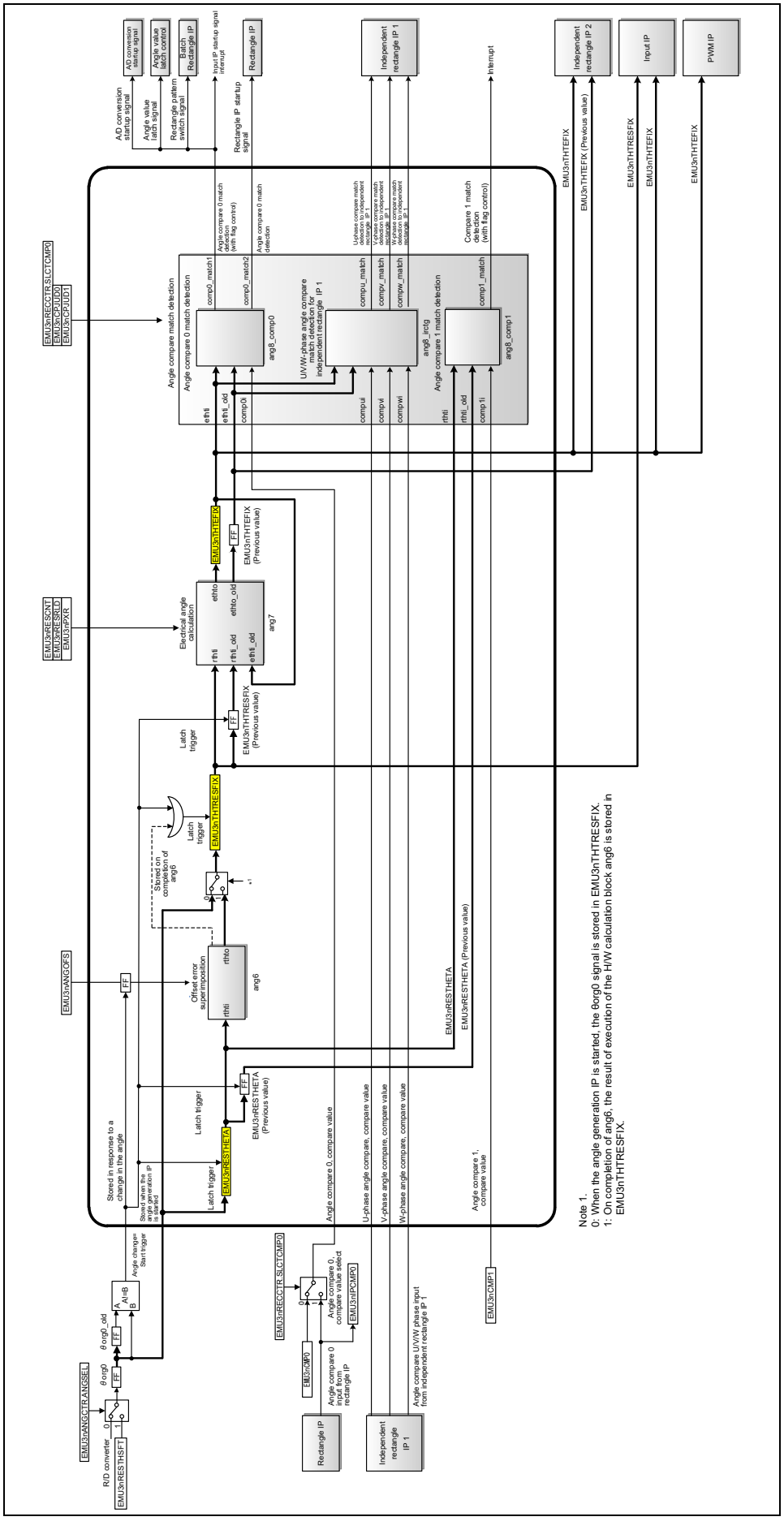
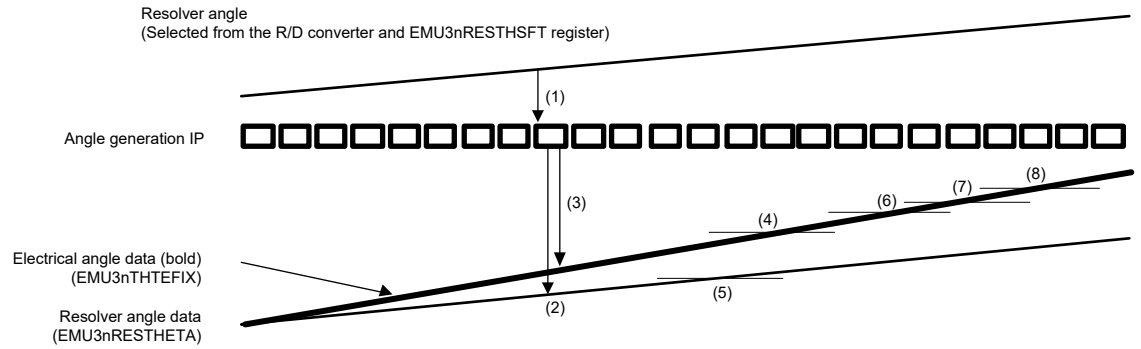


Figure 25.4 Angle Generation IP Processing

(1) The angle generation IP is started by the change of resolver angle.
(2) The resolver angle data (EMU3nRESTHETA) is generated.
(3) The electrical angle data (EMU3nTHTEFIX) is generated.
(4) When the electrical angle data matches with the compare value for electrical angle, angle compare 0 match detection signal is generated.
(5) When the resolver angle data matches with the compare value for resolver angle, angle compare 1 match detection signal is generated.
(6) When the electrical angle data matches with the compare value for independent rectangle IP 1 U-phase, angle compare match detection signal for independent rectangle IP 1 U-phase is generated.
(7) When the electrical angle data matches with the compare value for independent rectangle IP 1 V-phase, angle compare match detection signal for independent rectangle IP 1 V-phase is generated.
(8) When the electrical angle data matches with the compare value for independent rectangle IP 1 W-phase, angle compare match detection signal for independent rectangle IP 1 W-phase is generated.



The compare value for electrical angle is set by the EMU3nCMP0 register or rectangle IP.
The compare value for resolver angle is set by the EMU3nCMP1 register.
The compare value for independent rectangle IP 1 U-phase is selected from the EMU3nIRUCPPNn register (n = 0 to 2) by the independent rectangle IP 1.
The compare value for independent rectangle IP 1 V-phase is selected from the EMU3nIRVCPPNn register (n = 0 to 2) by the independent rectangle IP 1.
The compare value for independent rectangle IP 1 W-phase is selected from the EMU3nIRWCPPNn register (n = 0 to 2) by the independent rectangle IP 1.

Figure 25.5 Example of Angle Generation

Processing for Calculating an Electrical Angle from a Resolver Angle

The angle generation IP calculates an electrical angle from a resolver angle during electrical angle calculation processing which are shown in **Figure 25.4**. The angle generation IP processing and register setting are explained below according to examples 1 and 2 shown in **Figure 25.6**.

- Example 1 in **Figure 25.6**

  The configuration has 6 2-pole 3-phase coils. 1 cycle of the 3-phase waveform (electrical angle) is equal to 1 cycle of the mechanical angle. The resolver angle, however, has a twice larger frequency than the mechanical angle. The resolver angle pole count is represented by 2. Consequently, the electrical angle is calculated as follows:

  (Angle value of electrical angle) = (Angle value equivalent to 2 cycles of resolver angle (0° to 720°)) × 0.5

- Example 2 in **Figure 25.6**

  The configuration has 12 4-pole 3-phase coils. 1 cycle of the 3-phase waveform (electrical angle) is equal to 1/2 cycle of the mechanical angle. The resolver angle has the same period as the mechanical angle and the resolver angle pole count is represented by 1. Consequently, the electrical angle is calculated as follows:

  (Angle value of electrical angle) = (Angle value equivalent to 1/2 cycle of resolver angle (0° to 180°)) × 2.0

The above-mentioned “angle values equivalent to n cycles of resolver angle” exceeding 360° are calculated by extracting the transition of the position of the resolver angle pole. More specifically, a counter is provided that detects transitions of maximum value ↔ minimum value of the resolver angle and counts up and down the transitions. The sum

of the count value of the counter multiplied by 4096 and the resolver angle is equivalent to the angle value corresponding to n cycles of resolver angle.

(The maximum value of the counter can be changed using the resolver angle pole count that is specified by the EMU3nRESRLD register.)

The above "× 0.5" and "× 2.0" coefficients are set in the EMU3nPXR register.

The register settings for examples 1 and 2 in **Figure 25.6** are listed in **Table 25.178**. The transitions of the angle values are shown in the respective examples.

Table 25.178 Sample Parameter Settings for Calculating the Electrical Angle

Register	Example 1 in **Figure 25.6**	Example 2 in **Figure 25.6**
EMU3nRESRLD	"1" (resolver angle pole count 2)	"0" (resolver angle pole count 1)
EMU3nPXR	"0080_H" (0.5 fixed-point number)	"0200_H" (2.0 fixed-point number)

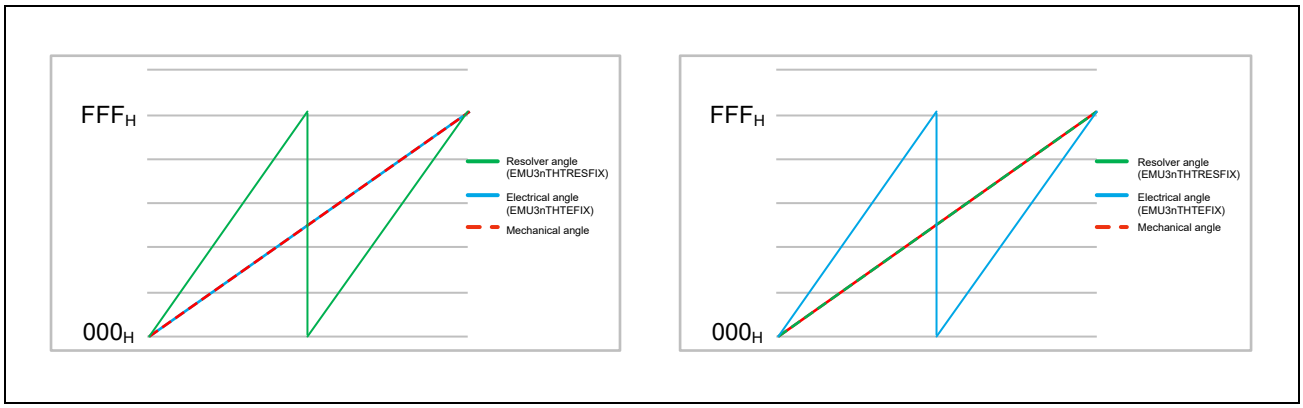


Figure 25.6 Examples of Electrical Angle Calculation (Left: Example 1, Right: Example 2)

CAUTION

The H/W accelerator expresses electrical angles of 0° to 360° with the EMU3nTHTEFIX register values 0000_H to $0FFF_H$ (unsigned integer values). There are cases in which the resolution of the resolver angle value indicated in the EMU3nRESTHETA register differs depending on the motor system; for example, the resolver values of 0° to 360° may be expressed as 0000_H to $00FF_H$ (unsigned integer values). In such a case, the magnification with which the resolver angle value is to be set in the EMU3nPXR register need be changed so as to ensure yield the 16-bit resolution.

(For inputs that cause the EMU3nRESTHETA register to be loaded with values 0000_H to $00FF_H$, the EMU3nTHTEFIX register can be loaded with values of 0000_H to $0FF0_H$ by setting EMU3nPXR to "16.0.")

The rest of this section details the functions that are shown in **Figure 25.4**.

(1) Func(ang6) Convolve offset error

[ang6] Convolution offset error

Explanation	Adds offset value to resolver angle value for output.
Arguments	rthti: Input resolver angle value (EMU3nRESTHETA) *rthto: Output resolver angle value (EMU3nTHTRESFIX)
Related registers	EMU3nANGOFS: Resolver angle offset value register (int12_t)

Processing	void ang6 (uint12_t rthti, uint12_t *rthto) {
	
	*rthto = (rthti + EMU3nANGOFS) & 0x0FFF;
	}

(2) Func(ang7) Electrical angle calculation

[ang7] Calculate electrical angle

Explanation	Calculates the electrical angle value from the resolver angle value and resolver angle number of cycles count value.
Arguments	rthti: Resolver angle value (EMU3nTHTRESFIX) rthti_old: Resolver angle value (previous value) (EMU3nTHTRESFIX (previous value)) ethti_old: Electrical angle value (EMU3nTHTEFIX (previous value)) *ethto: Electrical angle value (EMU3nTHTEFIX) *ethto_old: Electrical angle value (previous value) (EMU3nTHTEFIX (previous value))
Related register	EMU3nRESCNT: Resolver angle cycle count value register (uint3_t) EMU3nRESRLD: Resolver angle pole count value register (uint3_t) EMU3nPXR: Electrical angle generation coefficient register (int16_t)

Processing	void ang7 (uint12_t rthti, uint12_t rthti_old, uint12_t ethti_old, uint12_t *ethto, uint12_t *ethto_old) {
	
	//Note: The input value of ethti_old is loaded with the ethto value established during the preceding ang7 execution.
	// The value of ethti_old is propagated to ethto_old after ang7 execution.
	*ethto_old = ethti_old;
	
	//Note: rthti_old is loaded with the rthti value that is established during the e preceding angle generation IP execution.
	
	uint1_t max2min, min2max;
	int5_t rescnt;
	uint15_t tmp_tht;
	
	if(0xF00 < rthti_old && rthti < 0x0FF)
	max2min = 1;
	else
	max2min = 0;
	
	if(0xF00 < rthti && rthti_old < 0x0FF)
	min2max = 1;
	else
	min2max = 0;

```
  if       (max2min == 1)
    rescnt = EMU3nRESCNT + 1;
  else if(min2max == 1)
    rescnt = EMU3nRESCNT - 1;
  else
    rescnt = EMU3nRESCNT;

  if        ( rescnt > EMU3nRESRLD)
    EMU3nRESCNT = 0;
  else if ( rescnt < 0 )
    EMU3nRESCNT = EMU3nRESRLD;
  else
    EMU3nRESCNT = rescnt & 0x7;

  tmp_tht = (EMU3nRESCNT << 12) | rthti ;
  *ethto = ( (tmp_tht * EMU3nPXR) >> 8 ) & 0x0FFF ;

  *rthti_old = rthti;

}
```

(3) Func(ang8) angle compare match

Angle compare 0, angle compare 1, independent rectangle IP1 U phase, V phase, and W phase compare matches are detected as follows:

COMP: angle compare 0, angle compare 1, independent rectangle IP1 U phase, V phase, and W phase compare match register

i: 1 for angle compare 1 match detection and 0 for the other detections

Angle: EMU3nTHTEFIX for angle compare 0, independent rectangle IP1 U phase, V phase, and W phase compares.
EMU3nRESTHETA for angle compare 1

A match is recognized when the following conditions are met:

Positive rotation mode

Angle (previous value) < Compare set value ≤ Angle

Reverse rotation mode

Angle (previous value) > Compare set value ≥ Angle

But judgment 1 and judgment 2 are added as match criteria because additional conditions are required when making judgments on a compare match in between the situation in which the angle changes from "FFF_H" to "0_H" (or from "0_H" to "FFF_H").

When the angle changes from "FFF_H" to "0_H"

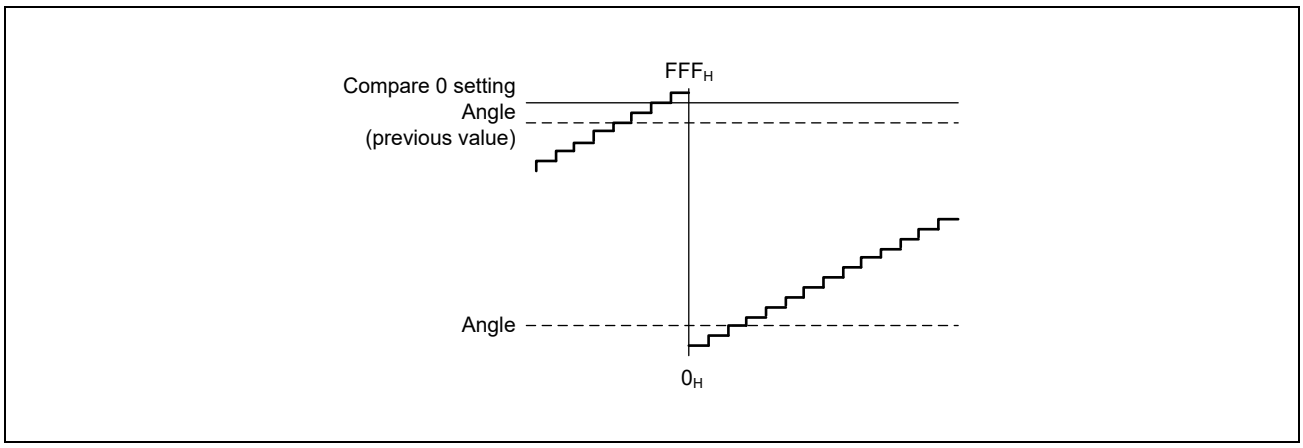


Figure 25.7 When the Angle Changes from FFF_H to 0_H

Judgment 1: FFF_H – EMU3nCPJUDi_u[7:0] < angle_u[11:0] (previous value) and angle_u[11:0] < EMU3nCPJUDi_u[7:0]

Judgment 2: FFF_H – EMU3nCPJUDi_u[7:0] < angle_u[11:0] and angle_u[11:0] (previous value) < EMU3nCPJUDi_u[7:0]

Positive rotation mode

Judgment 1 = Not met AND Judgment 2 = Not met AND angle_u[11:0] (previous value) < COMP ≤ angle_u[11:0]

Reverse rotation mode

Judgment 1 = Not met AND Judgment 2 = Not met AND angle_u[11:0] (previous value) > COMP ≥ angle_u[11:0]

When in positive mode and the electrical angle changes from "FFF_H" to "0_H"

Judgment 1 = Met AND ((angle_u[11:0] (previous value) < COMP) OR (COMP ≤ angle_u[11:0]))

When in reverse mode and the electrical angle changes from "0_H" to "FFF_H"

Judgment 2 = Met AND ((angle_u[11:0] (previous value) > COMP) OR (COMP ≥ angle_u[11:0]))

In angle compare 0 comparison value select processing, if the SLCTCMP0 bit of the EMU3nRECCTR is set to 0, the value of the EMU3nCMP0 register is set as the angle compare 0 comparison value. If the SLCTCMP0 bit of the EMU3nRECCTR register is set to 1, the results of rectangle IP operation are set. Even if the SLCTCMP0 bit of the EMU3nRECCTR register is set to 1, however, the first compare operation performed after the EMU3 is started sets up the value of the EMU3nCMP0 register as the angle compare 0 comparison value.

(4) About the compare match of the rectangle wave output

In angle compare 0 match or angle compare 1 match with the angle data, even if another compare match is performed again with the same value after one compare match is performed, the latter compare match is held cancelled until a new compare value is set up (flag control). This prevention of consecutive compare matches is applied to angle compare matches except the one for starting the rectangle IP. Any rectangle IP startup through a compare match occurring during the execution of the rectangle IP is ignored.

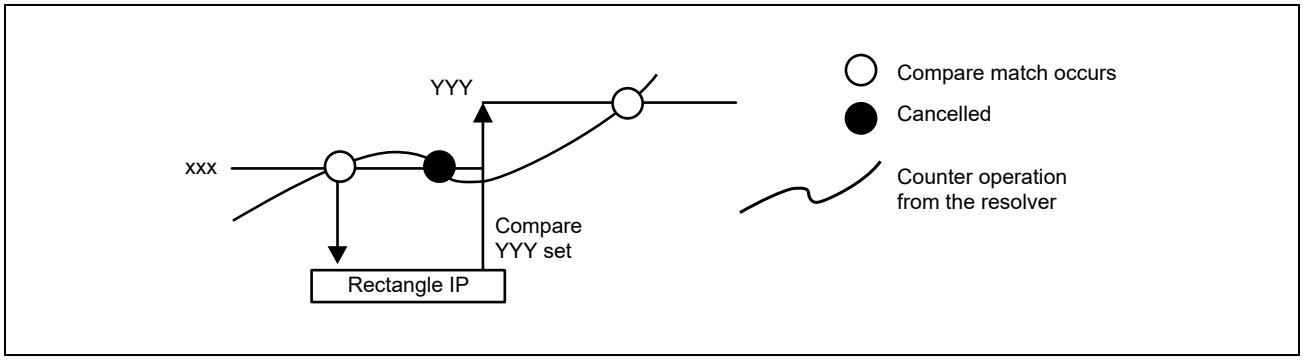


Figure 25.8 Prevention of Consecutive Compare Matches

(5) Angle comparison 0 match and angle comparison 1 match after EMU3 initialization

The settings of the angle generation IP immediately after EMU3 initialization are as follows.

- 0° is selected by the EMU3nCMP0 register for the value for comparison for an angle comparison 0 match.
- 0° is selected by the EMU3nCMP1 register for the value for comparison for an angle comparison 1 match.
- The resolver angle from the R/D converter is selected by the ANGSEL bit of the EMU3nANGCTR register for the angle value input to the angle generation IP.

If the angle value output from the R/D converter is static at around 0°, the angle indicated by the R/D converter may fluctuate (e.g. 12 bits ±4LSB), causing angle comparison 0 matches to occur and the setting of the CMP0IF bit of the EMU3nINTSD register to become “1”.

Similarly, the CMP1IF bit of the EMU3nINTSD register may become “1” in response to an angle comparison 1 match.

25.4.3 Input IP

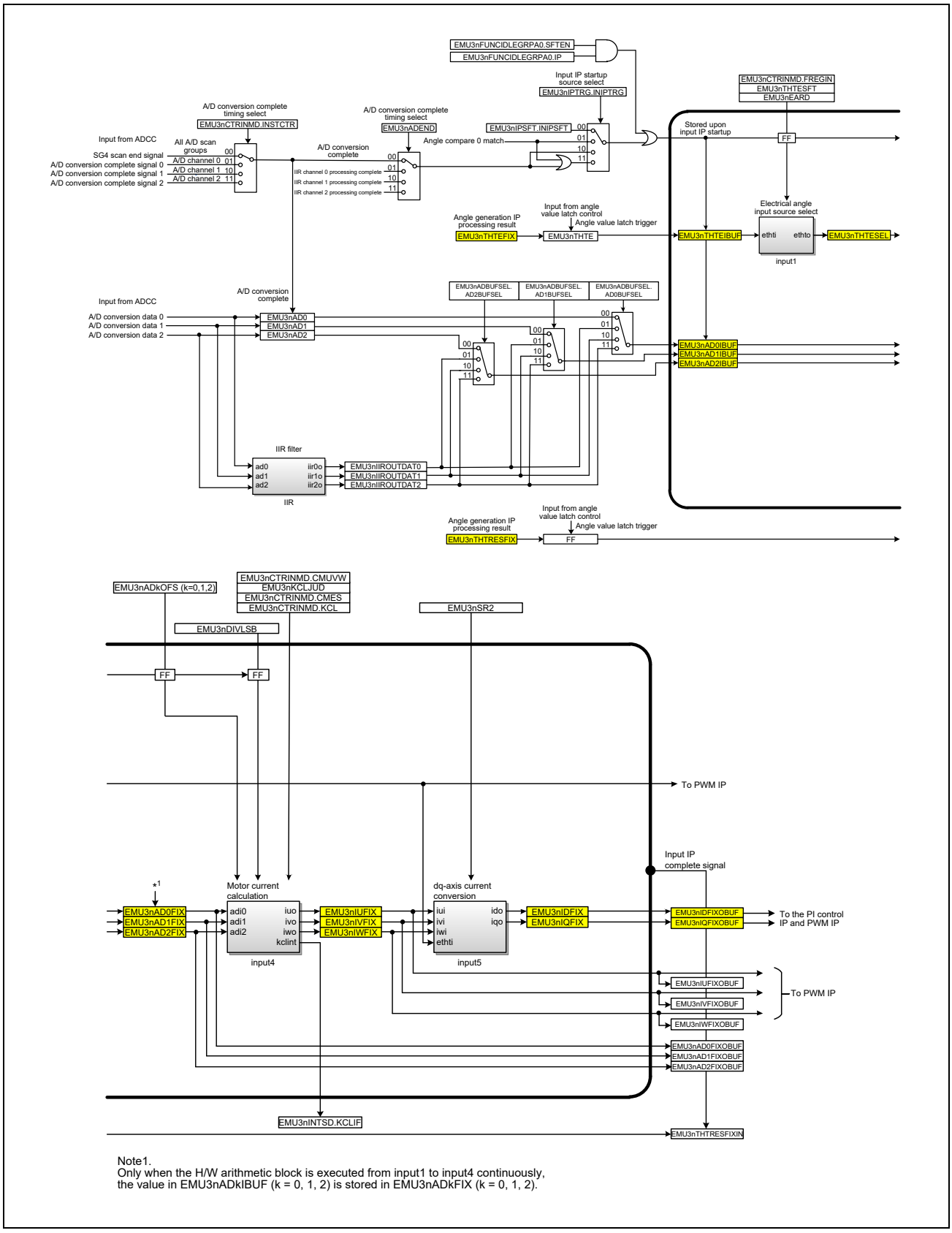
The input IP calculates the dq-axis current values based on the motor's 3-phase current values and the electrical angle. The motor's 3-phase current values are supplied from the EMU3nADk registers (k = 0, 1, 2) which contain the converted values (3-phase current values) from the A/D converters or from the EMU3nIIROUTDATk registers (k = 0, 1, 2) which contain the outputs from the IIR filters. The electrical angle can be selected through the EMU3nTHTEFIX register which contains the electrical angle input from the angle generation IP or through the EMU3nTHTESFT register which contains a software input value.

Figure 25.9 shows the flow of the input IP processing.

The input IP is started when the startup source, which is selected by the INIPTRG bits of the EMU3nIPTRG register, occurs. If the software trigger is selected, the input IP can be started by setting the INIPSFT bit of the EMU3nIPSFT register to 1. It can also be started by software by setting the IP and SFTEN bits of the EMU3nFUNCIDLEGRPA0 register to 1 regardless of the value of the INIPTRG bit in the EMU3nIPTRG register.

The INIF bit of the EMU3nINTSD register is set to 1 when the input IP processing is completed. The INIF bit can be cleared by setting the INIFC bit of the EMU3nINTSDC register to 1.

The rest of this subsection gives a detailed description of the functions that are shown in **Figure 25.9**.



Note1.
Only when the H/W arithmetic block is executed from input1 to input4 continuously, the value in EMU3nADkIBUF (k = 0, 1, 2) is stored in EMU3nADkFIX (k = 0, 1, 2).

Figure 25.9 Flow of Input IP Processing

(1) Func(input1) Electrical angle source select

[input1] Electrical angle source select

Explanation	Adds the offset value to the angle value.	
Arguments	ethti: Input angle value (EMU3nTHTEIBUF)	*ethto: Output angle value (EMU3nTHTESEL)
Related registers	EMU3nCTRINMD.FREGIN: Angle data select bit (value: 0, 1) EMU3nTHTESFT: Input IP electrical angle soft input register (uint12_t) EMU3nEARD: Electrical angle response delay correction variable register (uint12_t) (value: 0 to 4095)	

Processing	void input1 (uint12_t ethti, uint12_t *ethto) {
	
	if (EMU3nCTRINMD.FREGIN == 0)
	  *ethto = EMU3nTHTESFT ;
	else
	  *ethto = (ethti + EMU3nEARD) & 0x0FFF ;
	}	

(2) Func(input4) Motor current value calculation

[input4] Motor current value calculation

Explanation	Calculates the motor current values.	
Arguments	adi0: Input A/D ch0 current value (EMU3nAD0FIX) adi1: Input A/D ch1 current value (EMU3nAD1FIX) adi2: Input A/D ch2 current value (EMU3nAD2FIX)	*iuo: Output U phase current value (EMU3nIUFIX) *ivo: Output V-phase current value (EMU3nIVFIX) *iwo: Output W-phase current value (EMU3nIWFIX) *kclint: Output interrupt source (EMU3nINTSD.KCLIF)
Related register	EMU3nADkOFS (k = 0, 1, 2): A/D channel k origin correction value register (int16_t) EMU3nCTRINMD.CMES: Current value measurement target select bit (0,1) EMU3nDIVLSB: LSB adjustment register (int32_t) EMU3nCTRINMD.CMUVW: Current value measurement target select bit (0,1,2,4) EMU3nCTRINMD.KCL: Kirchhoff's current law violation detection bit (0,1) EMU3nKCLJUD: Kirchhoff's current law threshold value register (uint31_t)	

Processing

```
// Mortor current value calculation
void input4 (int16_t adi0, int16_t adi1, int16_t adi2,
                    int32_t *iuo, int32_t *ivo, int32_t *iwo, uint1_t *kclint) {

  int16_t tmp0, tmp1, tmp2;
  int32_t tmpu;
  int32_t kcl;

  tmp0 = adi0 - EMU3nAD0OFS;
  tmp1 = adi1 - EMU3nAD1OFS;
  tmp2 = adi2 - EMU3nAD2OFS;

  *ivo = ( tmp0×EMU3nDIVLSB ) >> 16;
  *iwo = ( tmp1×EMU3nDIVLSB ) >> 16;

  tmpu = ( tmp2×EMU3nDIVLSB ) >> 16;
  if ( EMU3nCTRINMD.KCL == 1 ) { // Check for Kirchhoff's current law violation.
    kcl = tmpu + *ivo + *iwo ;
    if ( kcl > EMU3nKCLJUD )          *kclint = 1;
    else if ( kcl < -EMU3nKCLJUD ) *kclint = 1;
    else                                  *kclint = 0;
  }

  if ( EMU3nCTRINMD.CMES == 0 ) {
    *iuo = - ( *ivo + *iwo );
  } else {
    *iuo = tmpu;
    if ( EMU3nCTRINMD.CMUVW == 2 ) {
      *ivo = - ( *iwo + *iuo );
    } else if ( EMU3nCTRINMD.CMUVW == 4 ) {
      *iwo = - ( *iuo + *ivo );
    } else if ( EMU3nCTRINMD.CMUVW == 1 ) {
      *iuo = - ( *iwo + *ivo );
  }
}
```

Processing description: Motor current value calculation processing

The A/D values are quantized as shown below when calculating the U/V/W phase current values from the A/D converted values.

The origin correction value which is used to convert unsigned A/D converted values to signed values is placed in the EMU3nADkOFS register. Signed integer A/D data is calculated by subtracting the value of the EMU3nADkOFS register from the A/D converted values.

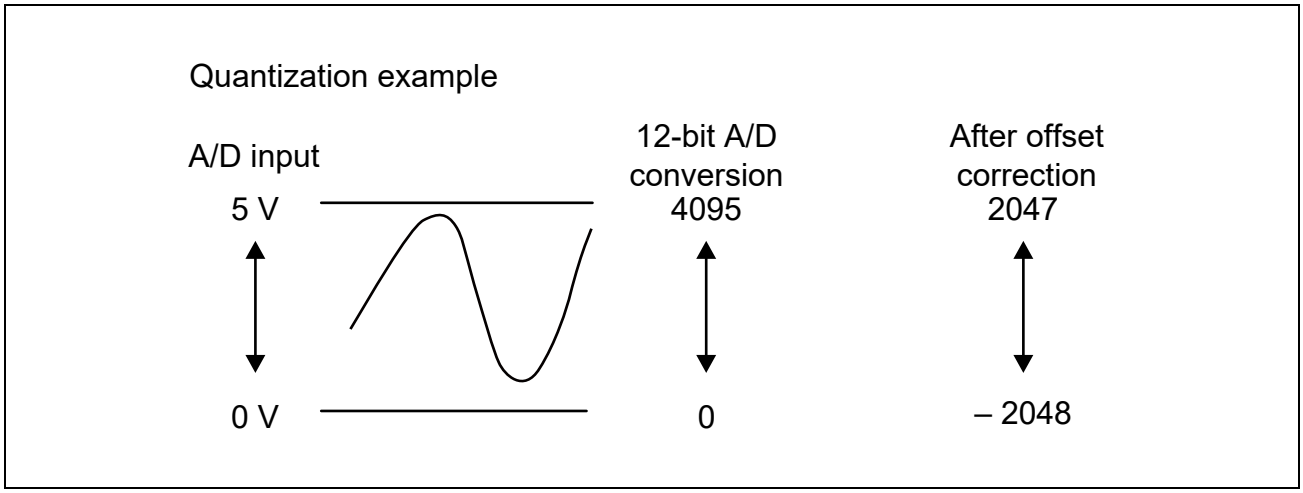


Figure 25.10 Example Quantization of the A/D Values and Calculation of Signed Integer A/D Data by Offset Correction

Processing description: Kirchhoff's current law violation detection processing

The sum of the motor current values of the U/V/W phases are basically 0 according to the first law of Kirchhoff (current law).

If the sum of the phase currents exceeds the threshold value, it is supposed that a circuit fault has occurred in the control feedback loop of either one phase.

If the KCL bit of the EMU3nCTRINMD register is set to 1 (enabling the detection of Kirchhoff's current law violations), a Kirchhoff's current law violation detect interrupt source is generated when the sum of the phase currents exceeds the threshold value specified by the EMU3nKCLJUD register.

The KCLIF bit of the EMU3nINTSD register is set to 1 when the above-mentioned interrupt source occurs. The KCLIF bit can be cleared by setting the KCLIFC bit of the EMU3nINTSDC register.

A Kirchhoff's current law interrupt source is generated when

$-Jud \leq Iu + Iv + Iw \leq Jud$

is not established.

Iu: Motor current value of the U phase

Iv: Motor current value of the V phase

Iw: Motor current value of the W phase

Jud: Kirchhoff's current law threshold value (EMU3nKCLJUD)

(3) Func(input5) dq-axis current conversion

[input5] dq-axis current conversion

Explanation	Converts the UVW phase current values to dq-axis current values.	
Arguments	iui: Input U phase current value (EMU3nIUFIX) ivi: Input V phase current value (EMU3nIVFIX) iwi: Input W phase current value (EMU3nIWFIX) ethti: Input electrical angle (EMU3nTHTESEL)	*ido: Output d-axis current value (EMU3nIDFIX) *iqo: Output q-axis current value (EMU3nIQFIX)
Related register	EMU3nSR2: dq-axis current conversion coefficient register (int32_t)	

Processing

```
void input5 (int32_t   iui, int32_t   ivi, int32_t   iwi, uint12_t ethti,
int32_t *ido, int32_t *iqo ) {

   int32_t sin60,sin30;
   int32_t cos60,cos30;

  //sin(θ+60°) = (sin(θ) * cos(60°) + cos(θ) * sin(60°) ) >> 15;
  sin60 = (sin_table[ethti] * 0x4000 + cos_table[ethti] * 0x6EDA ) >> 15;
  //sin(θ+30°) = (sin(θ) * cos(30°) + cos(θ) * sin(30°) ) >> 15;
  sin30 = (sin_table[ethti] * 0x6EDA + cos_table[ethti] * 0x4000 ) >> 15;
  //cos(θ+60°) = (cos(θ) * cos(60°) - sin(θ) * sin(60°)) >> 15;
  cos60 = (cos_table[ethti] * 0x4000 - sin_table[ethti] * 0x6EDA ) >> 15;
  //cos(θ+30°) = (cos(θ) * cos(30°) - sin(θ) * sin(30°)) >> 15;
  cos30 = (cos_table[ethti] * 0x6EDA - sin_table[ethti] * 0x4000 ) >> 15;

  *ido = (EMU3nSR2 * (   ((cos_table[ethti] * iui ) >> 15)
    - ((cos60                  * ivi ) >> 15)
    - ((sin30                  * iwi ) >> 15))) >> 16;
  *iqo = (EMU3nSR2 * (- ((sin_table[ethti] * iui) >> 15)
  + ((sin60                  * ivi ) >> 15)
  - ((cos30                  * iwi ) >> 15))) >> 16;
}

// sin(), cos() represents signed fixed-point data whose fractional part is 15 bits long.
// sin(30°) = 0x00004000; value obtained by multiplying “sin(30°)” by 0x8000
// cos(60°) = 0x00004000; value obtained by multiplying “cos(60°)” by 0x8000
// sin(60°) = 0x00006EDA; value obtained by multiplying “sin(60°)” by 0x8000
// cos(30°) = 0x00006EDA; value obtained by multiplying “cos(30°)” by 0x8000
// sin_table and cos_table represent
// 32-bit signed fixed-point data whose fractional part is 15 bits long and which
// are generated from tables whose index is the 12-bit electrical angle (in 1-bit increments).
```

Processing description: dq-axis current conversion processing

The dq-axis current values (EMU3nIDFIX and EMU3nIQFIX) are generated from the U/V/W phase current values (EMU3nIUFIX, EMU3nIVFIX, and EMU3nIWFIX) and the electrical angle θ (EMU3nTHTESEL).

The formulas for generating the dq-axis current values (Id, Iq) from the U/V/W phase current values (Iu, Iv, Iw) input data.

$$I_d = \sqrt{\frac{2}{3}} \times (\cos\theta \times I_u - \cos(\theta + 60°) \times I_v - \sin(\theta + 30°) \times I_w)$$

$$I_q = \sqrt{\frac{2}{3}} \times (-\sin\theta \times I_u + \sin(\theta + 60°) \times I_v - \cos(\theta + 30°) \times I_w)$$

θ: Electrical angle specified by the EMU3nTHTESEL register

Iu: U phase current value specified by the EMU3nIUFIX register

Iv: V phase current value specified by the EMU3nIVFIX register

Iw: W phase current value specified by the EMU3nIWFIX register

Id: d-axis current value specified by the EMU3nIDFIX register

Iq: q-axis current value specified by the EMU3nIQFIX register

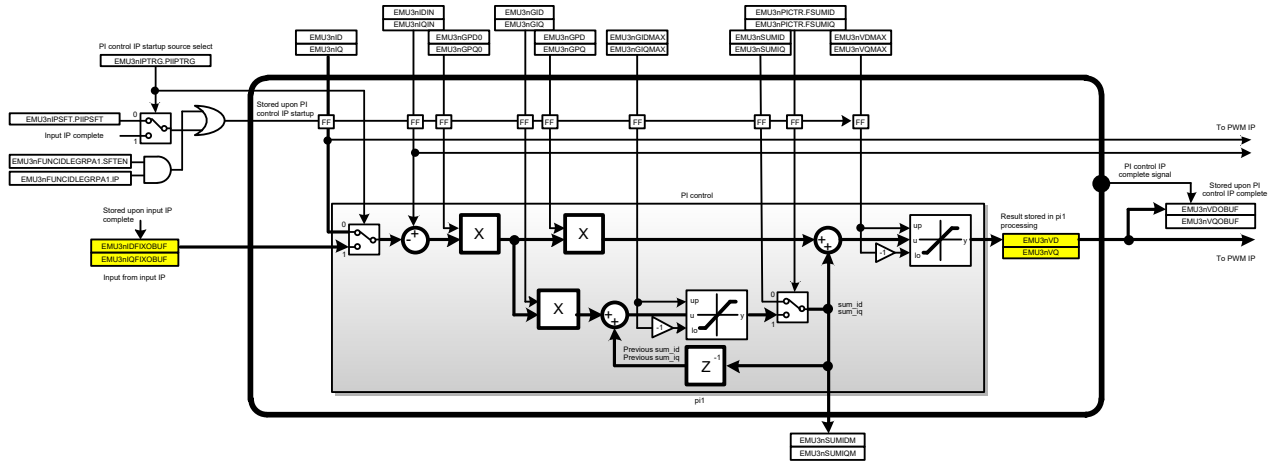
25.4.4 PI Control IP

The PI control IP generates dq-axis voltage values from the dq-axis current values. The dq-axis current values can be selected from the EMU3nIDFIX register and EMU3nIQFIX register that hold the operation results from the input IP or from the EMU3nID register and EMU3nIQ register that hold software input value.

The PI control IP is started upon completion of input IP processing if the PIIPTRG bit of the EMU3nIPTRG register is set to 1. If the PIIPTRG bit is set to 0, it can be started by setting the PIIPSFT bit of the EMU3nIPSFT register. It can also be started by software by setting the IP and SFTEN bits of the EMU3nFUNCIDLEGRPA1 register to 1 regardless of the value of the INIPTRG bit in the EMU3nIPTRG register.

When the PI control IP processing is completed, the PIIF bit of the EMU3nINTSD register is set to 1. The PIIF bit can be cleared by setting the PIIFC bit of the EMU3nINTSDC register to 1.

Figure 25.11 shows the flow of PI control IP processing.



Note: "Z-1" in the figure denotes a delay and also indicates that the results of sum_id and sum_iq that are established during the preceding PI control IP execution are referenced.

Figure 25.11 Flow of PI control IP Processing

(1) Func(pi1) pi control

[pi1] pi control

Explanation	Generates d- and q-axis voltage values from the d- and q-axis current values.	
Arguments	***in_id***: Input d-axis current value (**EMU3nIDFIXOBUF**) ***in_iq***: Input q-axis current value (**EMU3nIQFIXOBUF**)	****out_vd***: Output d-axis voltage value (**EMU3nVD**) ****out_vq***: Output q-axis voltage value (**EMU3nVQ**)
Related registers	**EMU3nIPTRG.PIIPTRG** : PI control IP startup trigger select bit (value: 0,1) **EMU3nIDIN** : d-axis directive current value register (int32_t) **EMU3nIQIN** : q-axis directive current value register (int32_t) **EMU3nID** : d-axis current value software input register (int32_t) **EMU3nIQ** : q-axis current value software input register (int32_t) **EMU3nGPD0** : d-axis proportional gain 0 register (int32_t) **EMU3nGPQ0** : q-axis proportional gain 0 register (int32_t) **EMU3nGID** : d-axis integral gain register (int32_t) **EMU3nGIQ** : q-axis integral gain register (int32_t) **EMU3nPICTR.FSUMID** : d-axis integrated value select bit (value: 0,1) **EMU3nPICTR.FSUMIQ** : q-axis integrated value select bit (value: 0,1) **EMU3nSUMID** : d-axis integrated value software input register (int32_t) **EMU3nSUMIQ** : q-axis integrated value software input register (int32_t) **EMU3nGIDMAX** : d-axis integrated maximum value register (uint31_t) **EMU3nGIQMAX** : q-axis integrated maximum value register (uint31_t) **EMU3nSUMIDM** : d-axis integrated value monitor register (int32_t) **EMU3nSUMIQM** : q-axis integrated value monitor register (int32_t) **EMU3nGPD** : d-axis proportional gain register (int32_t) **EMU3nGPQ** : q-axis proportional gain register (int32_t) **EMU3nVDMAX** : d-axis voltage maximum value register (uint31_t) **EMU3nVQMAX** : q-axis voltage maximum value register (uint31_t)	

Processing

```
void pi1 (int32_t   in_id,   int32_t   in_iq,
int32_t *out_vd, int32_t *out_vq ) {

  static int32_t sum_id=0, sum_iq=0;
  int32_t ids, iqs;

  if(EMU3nIPTRG.PIIPTRG == 0) { // Started with a soft trigger.
    ids = ( (EMU3nIDIN - EMU3nID) * EMU3nGPD0) >> 16;
    iqs = ( (EMU3nIQIN - EMU3nIQ) * EMU3nGPQ0) >> 16;
  } else { // Started with input IP completion.
    ids = ( (EMU3nIDIN - in_id ) * EMU3nGPD0) >> 16;
    iqs = ( (EMU3nIQIN - in_iq ) * EMU3nGPQ0) >> 16;
  }

  sum_id += (ids * EMU3nGID ) >> 16;
  sum_iq += (iqs * EMU3nGIQ ) >> 16;

  if ( EMU3nGIDMAX < sum_id )
    sum_id = EMU3nGIDMAX;
  else if ( sum_id < -EMU3nGIDMAX )
    sum_id = -EMU3nGIDMAX;
  if ( EMU3nGIQMAX < sum_iq )
    sum_iq = EMU3nGIQMAX;
  else if ( sum_iq < -EMU3nGIQMAX )
    sum_iq = -EMU3nGIQMAX;

  if ( EMU3nPICTR.FSUMID == 0)
    sum_id = EMU3nSUMID;
  if ( EMU3nPICTR.FSUMIQ == 0)
    sum_iq = EMU3nSUMIQ;
```

```
EMU3nSUMIDM = sum_id;
EMU3nSUMIQM = sum_iq;

*out_vd = sum_id + ((ids * EMU3nGPD ) >> 16);
*out_vq = sum_iq + ((iqs * EMU3nGPQ ) >> 16);

if ( EMU3nVDMAX < *out_vd )
   *out_vd = EMU3nVDMAX;
else if ( *out_vd < -EMU3nVDMAX )
   *out_vd = -EMU3nVDMAX;
if ( EMU3nVQMAX < *out_vq )
   *out_vq = EMU3nVQMAX;
else if ( *out_vq < -EMU3nVQMAX )
   *out_vq = -EMU3nVQMAX;

}
```

25.4.5 PWM IP

The PWM IP calculates the PWM compare values from the dq-axis voltage values and electrical angle. The dq-axis voltage values can be selected from the EMU3nVD register and EMU3nVQ register that hold the operation results of the PI control IP and the electrical angle can be selected from the input from the input IP, the input from the angle generation IP, or from the EMU3nTHTFORESFT register which holds the software input.

Figure 25.12 shows the flow of PWM IP processing.

If the PWMIPTRG bit of the EMU3nIPTRG register is set to 1, the PWM IP is started upon completion of PI control IP processing. If the PWM3nIPTRG bit is set to 0, the PWM IP can be started by setting the PWMIPSFT bit of the EMU3nIPSFT register to 1. It can also be started by software by setting the IP and SFTEN bits of the EMU3nFUNCIDLEGRPA2 register to 1 regardless of the value of the INIPTRG bit in the EMU3nIPTRG register.

The PWMIF bit can be cleared by setting the PWMIFC bit of the EMU3nINTSDC registe to 1.

The rest of this subsection gives a detailed description of the functions that are shown in **Figure 25.12**.

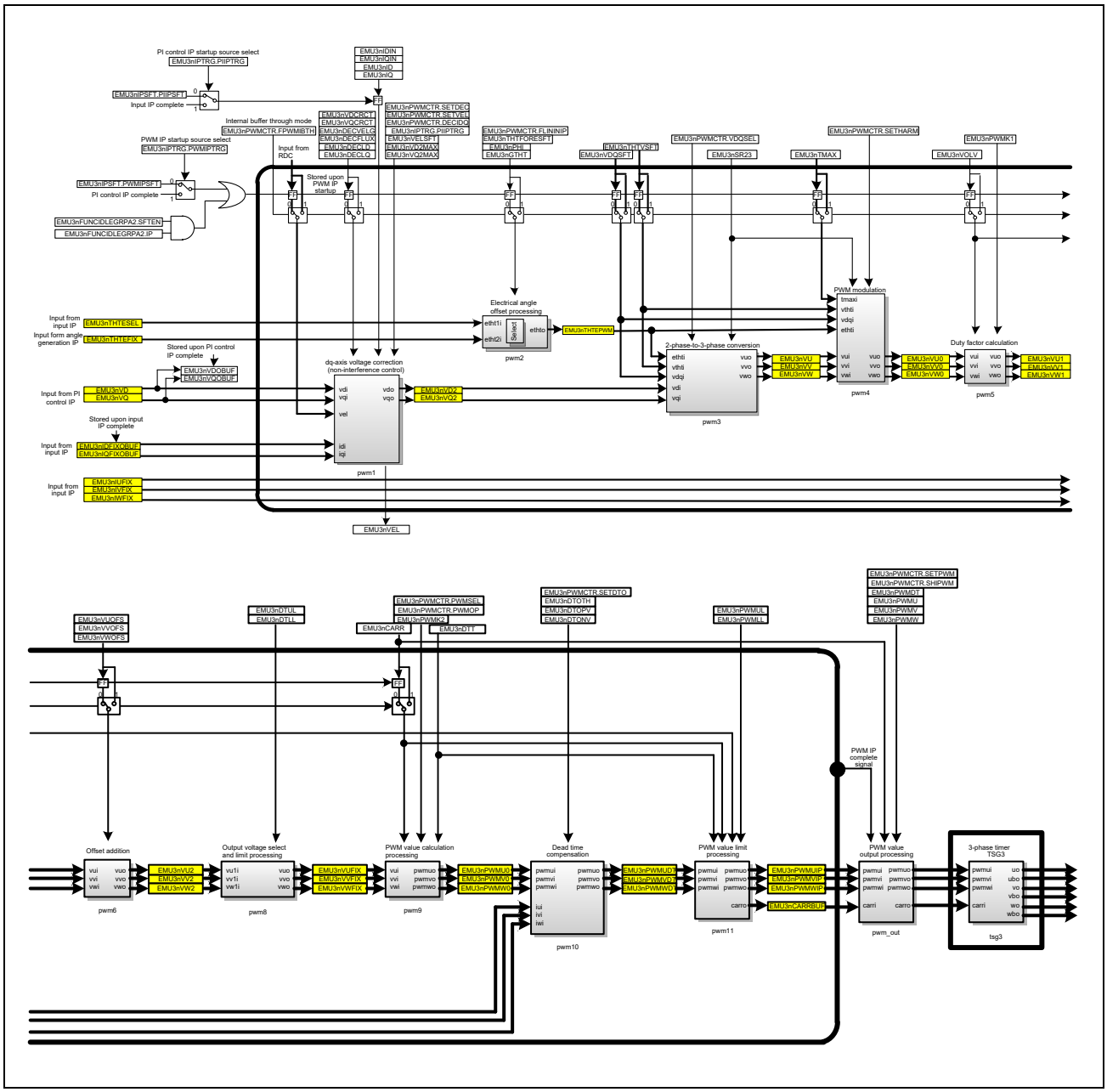


Figure 25.12 Flow of PWM IP Processing

(1) Func(pwm1) dq-axis voltage correction (non-interference control)

[pwm1] dq-axis voltage correction (non-interference control) processing

Explanation	Adds arbitrary values to the dq-axis voltage values through registers. Performs non-interference control calculations.
Arguments	*vdi*: Input d-axis voltage value (EMU3nVD) **vdo*: Output d-axis voltage value (EMU3nVD2) *vqi*: Input q-axis voltage value (EMU3nVQ) **vqo*: Output q-axis voltage value (EMU3nVQ2) *idi*: Input d-axis current value (EMU3nIDFIXOBUF) *iqi*: Input q-axis current value (EMU3nIQFIXOBUF) *vel*: Input angular velocity value (input from RDC)
Related registers	EMU3nVDCRCT : d-axis voltage correction value register (int32_t) EMU3nVQCRCT : q-axis voltage correction value register (int32_t) EMU3nPWMCTR.SETDEC : Non-interference control bit (0,1) EMU3nPWMCTR.SETVEL : Angular velocity select bit (0,1) EMU3nPWMCTR.DECIDQ : Non-interference control dq-axis current value select bits (0,1, 2) EMU3nID : d-axis current value software input register(int32_t) EMU3nIQ : q-axis current value software input register (int32_t) EMU3nIDIN : d-axis directive current value register (int32_t) EMU3nIQIN : q-axis directive current value register (int32_t) EMU3nVELSFT : Angular velocity value soft input register (int32_t) EMU3nVEL : Angular velocity value register (int32_t) EMU3nDECVELG : Non-interference control coefficient angular velocity value gain register (int32_t) EMU3nDECFLUX : Non-interference control coefficient magnetic flux value register (int32_t) EMU3nDECLD : Non-interference control coefficient Ld value register (int32_t) EMU3nDECLQ : Non-interference control coefficient Lq value register (int32_t) EMU3nVD2MAX : Non-interference control d-axis maximum value register (uint31_t) EMU3nVQ2MAX : Non-interference control q-axis maximum value register (uint31_t)

Processing	void pwm1 (int26_t *vel*, int32_t *vdi*, int32_t *vqi*, int32_t *idi*, int32_t *iqi*,
	int32_t **vdo*, int32_t **vqo*) {
	
	int32_t tmpd, tmpq;
	EMU3nVEL = (vel * 0x001DF5E7) >>16;
	if(EMU3nPWMCTR.SETDEC == 0){ // Disable non-interference control processing.
	tmpd = *vdi* + EMU3nVDCRCT;
	tmpq = *vqi* + EMU3nVQCRCT;
	} else { // Enable non-interference control processing.
	
	int32_t tmp_vel;
	int32_t tmp_id, tmp_iq;
	int32_t tmp0, tmp1, tmp2, tmp3;
	
	if(EMU3nPWMCTR.SETVEL == 0) // Select angular velocity soft input.
	tmp_vel = EMU3nVELSFT;
	else {
	tmp_vel = EMU3nVEL;
	}

```
    if(EMU3nPWMCTR.DECIDQ == 0 ){ // Select dq-axis current value soft input (when using PI control IP).
      tmp_id = EMU3nID;
      tmp_iq = EMU3nIQ;
    } else if(EMU3nPWMCTR.DECIDQ == 1 ){ // Select dq-axis current values established after completion of input IP
processing.
      tmp_id = idi;
      tmp_iq = iqi;
    } else if(EMU3nPWMCTR.DECIDQ == 2 ){ // Select dq-axis directive current value.
      tmp_id = EMU3nIDIN;
      tmp_iq = EMU3nIQIN;
    }

    tmp0 = (tmp_vel * EMU3nDECVELG   ) >> 16;

    tmp1 = (tmp_id   * EMU3nDECLD      ) >> 16;
    tmp2 = (tmp_iq   * EMU3nDECLQ      ) >> 16;

    tmp1 = (tmp0     * tmp1            ) >> 12;
    tmp2 = (tmp0     * tmp2            ) >> 12;
    tmp3 = (tmp0     * EMU3nDECFLUX    ) >> 28;

    tmpd = vdi - tmp2;
    tmpq = vqi + tmp1 + tmp3;
  }

  if      (tmpd >   EMU3nVD2MAX)
    *vdo =   EMU3nVD2MAX;
  else if(tmpd < -EMU3nVD2MAX)
    *vdo = -EMU3nVD2MAX;
  else
    *vdo = tmpd;
  if      (tmpq >   EMU3nVQ2MAX)
    *vqo =   EMU3nVQ2MAX;
  else if(tmpq < -EMU3nVQ2MAX)
    *vqo = -EMU3nVQ2MAX;
  else
    *vqo = tmpq;
}
```

Processing description: angular velocity calculation processing

The PWM IP is provided with the capability to calculate the angular velocity of the resolver angle for exercising non-interference control.

The angular velocity value output from the R/D converter is a signed 26-bit value and its 1LSB is equivalent to approximately 0.07[min^{-1}]. The data signal equivalent to the RDC angular velocity [min^{-1}] is multiplied by a correction coefficient inside the H/W accelerator to obtain the actual angular velocity [min^{-1}] in hexadecimal before being multiplied by ($2\pi/60$). The resultant value is stored in the EMU3nVEL register as a signed 32-bit angular velocity value [rad/sec] with a 12-bit fractional part.

Angular velocity [rad/sec] (12-bit fractional part)

= RDC angular velocity value [25:0] (signed bit extended) × correction coefficient (0.0698...) × (2π/60)

= RDC angular velocity value [25:0] (signed bit extended) × $001DF5E7_H$ >> 16

$001DF5E7_H$: Fixed value signal having a total length of 32 bits whose value is the correction coefficient (0.0698...) × (2π / 60) multiplied by $1000\ 0000_H$ (268435456).

When using the EMU3nVELSFT register with a software input value, configure it for a signed 32-bit angular velocity value [rad/sec] with a 12-bit fractional part.

Processing description: Non-interference control

The PWM IP is provided with the capability to exercise non-interference control.

Since the d-axis current and q-axis current interfere with each other, the control efficiency is improved by making corrections on the d- and q-axes in advance.

The correction processing is performed as shown below. The values of the coefficients need be preset from the CPU program. The angular velocity can be calculated using the angular velocity value in the EMU3nVEL register which is output from the R/D converter or the software input value through the EMU3nVELSFT register.

$vd^* = vd' - (iq \times (\omega \times \omega\ gain) \times Lq)$

$vq^* = vq' + (id \times (\omega \times \omega\ gain) \times Ld) + ((\omega \times \omega\ gain) \times \phi)$

vd*: d-axis voltage value output by the non- interference control function

vq*: q-axis voltage value output by the non- interference control function

vd': d-axis voltage value (d-axis after PI control) to be input to the non-interference control function

vq': q-axis voltage value (q-axis after PI control) to be input to the non- interference control function

id: d-axis current value specified by either one of EMU3nID, EMU3nIDFIXOBUF, and EMU3nIDIN in the figure (d-axis before PI control)

iq: q-axis current value specified by either one of EMU3nIQ, EMU3nIQFIXOBUF, and EMU3nIQIN in the figure (q-axis before PI control)

ω: Angular velocity value specified by EMU3nVEL or EMU3nVELSFT*[1]

ω gain: Coefficient value for the angular velocity value specified by EMU3nDECVELG

Ld: d-axis inductance value specified by EMU3nDECLD

Lq: q-axis inductance value specified by EMU3nDECLQ

ϕ: Magnetic flux value specified by EMU3nDECFLUX

Note 1. Since ω represents the angular velocity of the resolver angle, it is necessary to represent the coefficient for converting to the angular velocity of the electrical angle also with the ω gain.

CAUTION

The current values to be input for PI control are used as the input to non-interference control calculations.

When the input IP, PI control IP, and PWM IP are run for H/W arithmetic operations, the values of the EMU3nIDFIXOBUF (EMU3nIQFIXOBUF) registers which hold the results of input IP operation are used as the input current values for the PI control IP and the input current values for non-interference control by the PWM IP. When the PI control IP is not to be used for PI control, write the value that is calculated during the software processing into the EMU3nIDFIXOBUF (EMU3nIQFIXOBUF) register, which is used to hold the results of input IP operation, as the input current values for non-interference control processing before starting the PWM IP.

(2) Func(pwm2) Electrical angle offset processing

[pwm2] Electrical angle offset processing

Explanation	Adds the offset value to the angle value and outputs the results for phase adjustment.
Arguments	*etht1i*: Input angle value (EMU3nTHTESEL) **ethto*: Output angle value (EMU3nTHTEPWM) *etht2i*: Input angle value (EMU3nTHTEFIX)
Related registers	EMU3nPWMCTR.FLININIP: Electrical angle select bits (0,1,2) EMU3nTHTFORESFT: PWM IP electrical angle soft input register (uint12_t) EMU3nPHI: PWM IP electrical angle offset register (uint16_t) EMU3nGTHT: PWM IP electrical angle adjustment coefficient register (int16_t)

Processing	void pwm2 (uint12_t etht1i, uint12_t *etht2i*,
	uint12_t **ethto*) {
	
	if(EMU3nPWMCTR.FLININIP == 0) // Select electrical angle soft input.
	**ethto* = (((EMU3nTHTFORESFT + EMU3nPHI) & 0x0FFF) * EMU3nGTHT) >> 8 ;
	else if (EMU3nPWMCTR.FLININIP == 1) // Select input from input IP.
	**ethto* = (((*etht1i* + EMU3nPHI) & 0x0FFF) * EMU3nGTHT) >> 8 ;
	else if (EMU3nPWMCTR.FLININIP == 2) // Select input from angle generation IP.
	**ethto* = (((*etht2i* + EMU3nPHI) & 0x0FFF) * EMU3nGTHT) >> 8 ;
	}

(3) Func(pwm3) 2-phase-to-3-phase conversion

[pwm3] 2-phase-to-3-phase conversion processing

Explanation	Converts the dq-axis voltage values to UVW phase voltage values.
Arguments	*ethti*: Input electrical angle value (EMU3nTHTEPWM) *vthti*: Input dq-axis voltage phase angle software input value (EMU3nTHTVSFT) *vdi*: Input d-axis voltage value (EMU3nVD2) *vqi*: Input q-axis voltage value (EMU3nVQ2) *vdqi*: Input dq-axis voltage software input values (EMU3nVDQSFT) **vuo*: Output U phase voltage value (EMU3nVU) **vvo*: Output V phase voltage value (EMU3nVV) **vwo*: Output W phase voltage value (EMU3nVW)
Related registers	EMU3nPWMCTR.VDQSEL : dq-axis voltage input select bit (0,1) EMU3nSR23 : 3-phase voltage conversion coefficient register(int32_t)

Processing	void pwm3 (uint12_t *ethti*, uint12_t *vthti*, int32_t *vdi*, int32_t *vqi*, int32_t *vdqi*,
	int32_t **vuo*, int32_t **vvo*, int32_t **vwo*) {
	
	int32_t tmpd;
	int32_t A1, A2, B1, B2;
	
	// sin() and cos() represent signed fixed-point data having a 15-bit fractional part.
	const int32_t cos0 = 0x00008000; // Value of “cos(0°)” multiplied by 0x8000.
	const int32_t sin0 = 0x00000000; // “sin(0°)” multiplied by 0x8000.
	const int32_t cos120 = 0xFFFFC000; // “cos(120°)” multiplied by 0x8000.
	const int32_t sin120 = 0x00006EDA; // “sin(120°)” multiplied by 0x8000.
	
	if (EMU3nPWMCTR.VDQSEL == 0){
	tmpd = vdi;
	A1 = cos_table[ethti];
	A2 = sin_table[ethti];
	// cos(θ+120°) = (cos(θ) * cos(120°) – sin(θ) * sin(120°)) >> 15;
	B1 = (cos_table[ethti] * cos120 – sin_table[ethti] * sin120) >> 15;
	// sin(θ+120°) = (sin(θ) * cos(120°) + cos(θ) * sin(120°)) >> 15;
	B2 = (sin_table[ethti] * cos120 + cos_table[ethti] * sin120) >> 15;
	} else { // Use Vdq/Vθ
	tmpd =vdqi;
	A1 = cos_table[ethti+vthti];
	A2 = 0;
	// cos((θ+Vθ)+120°) = (cos(θ+Vθ) * cos(120°) – sin(θ+Vθ) * sin(120°)) >> 15;
	B1 = (cos_table[ethti+vthti] * cos120 – sin_table[ethti+vthti] * sin120) >> 15;
	B2 = 0;
	}
	*vuo = (EMU3nSR23 * (((A1 * tmpd) >> 15) - ((A2 * vqi) >> 15))) >> 16;
	*vwo = (EMU3nSR23 * (((B1 * tmpd) >> 15) - ((B2 * vqi) >> 15))) >> 16;
	*vvo = -(*vuo + *vwo);
	}
	
	// sin_table and cos_table represent
	/// 32-bit signed fixed-point data whose fractional part is 15 bits long and which
	/// are generated from tables whose index is the 12-bit electrical angle (in 1-bit increments).

Processing description: 2-phase-to-3-phase conversion processing

The U/V/W phase voltage values are generated as the input data to the dq-axes from the following two types of data in addition to the electrical angle θ (EMU3nTHTEPWM).

Given below are the formulas for generating the U/V/W phase voltage values (Vu, Vv, Vw) from the dq-axis input data.

Input data “d-axis voltage: Vd (EMU3nVD2), q-axis voltage Vq (EMU3nVQ2)”

$$V_u = \sqrt{\frac{2}{3}} \cdot \left(V_d \cdot \cos\theta - V_q \cdot \sin\theta\right)$$

$$V_w = \sqrt{\frac{2}{3}} \cdot \left(V_d \cdot \cos(\theta + 120°) - V_q \cdot \sin(\theta + 120°)\right)$$

$$V_v = -(V_u + V_w)$$

Input data “dq-axis voltage values: Vdq (EMU3nVDQSFT), dq-axis voltage phase angle: Vθ (EMU3nTHTVSFT)”

$$V_{dq} = \sqrt{{V_d}^2 + {V_q}^2}$$

$$V_\theta = \tan^{-1}\frac{V_q}{V_d}$$

$$V_u = \sqrt{\frac{2}{3}} \cdot V_{dq} \cdot \cos(\theta + V_\theta)$$

$$V_w = \sqrt{\frac{2}{3}} \cdot V_{dq} \cdot \cos(\theta + V_\theta + 120°)$$

$$V_v = -(V_u + V_w)$$

θ: Electrical angle specified by the EMU3nTHTEPWM register

Vd: d-axis voltage specified by the EMU3nVD2 register

Vq: q-axis voltage specified by the EMU3nVQ2 register

Vdq: dq-axis voltage specified by the EMU3nVDQSFT register

Vθ: dq-axis voltage phase angle specified by the EMU3nTHTVSFT register

Vu: U-phase voltage value specified by the EMU3nVU register

Vv: V-phase voltage value specified by the EMU3nVV register

Vw: W-phase voltage value specified by the EMU3nVW register

(4) Func(pwm4) PWM modulation

This function modulates the 3-phase voltage values EMU3nVU, EMU3nVV, and EMU3nVW and outputs the results to EMU3nVU0, EMU3nVV0, and EMU3nVW0. For details of processing, contact our sales representative.

[pwm4] PWM modulation processing

Processing description: Performs modulation to improve the output voltage characteristics.

The PWM IP is provided with the function to modulate the 3-phase voltage waveforms for improved output voltage characteristics.

Figure 25.13 shows the waveforms that are obtained without using the modulation function and **Figure 25.14** waveforms that are obtained using the modulation function. Using the modulation function, it is possible to get line voltages of large amplitudes from 3-phase voltage waveforms of the same amplitude.

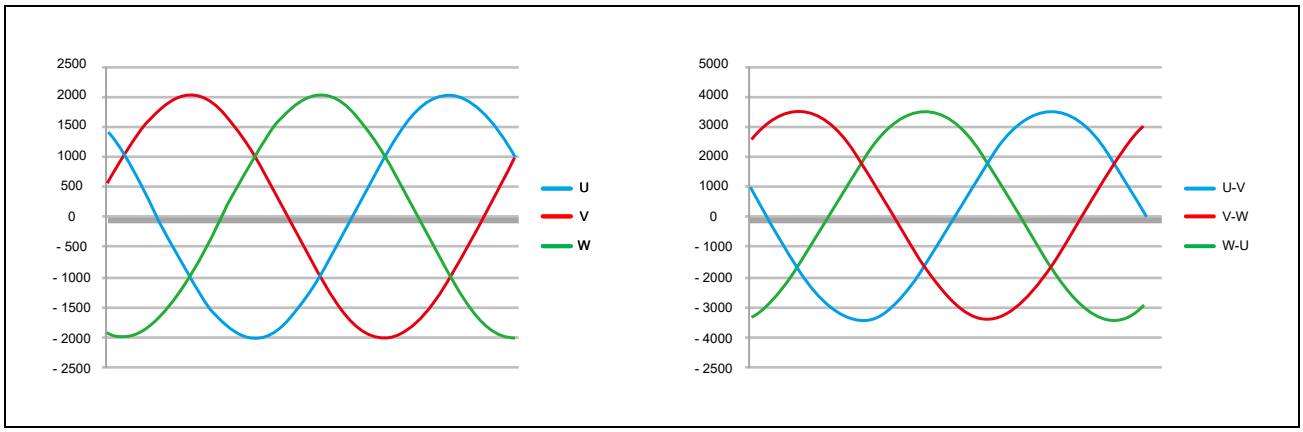


Figure 25.13 3-phase Waveforms without Modulation (Left: 3-phase Voltage Waveforms, Right: Line Voltage Waveforms)

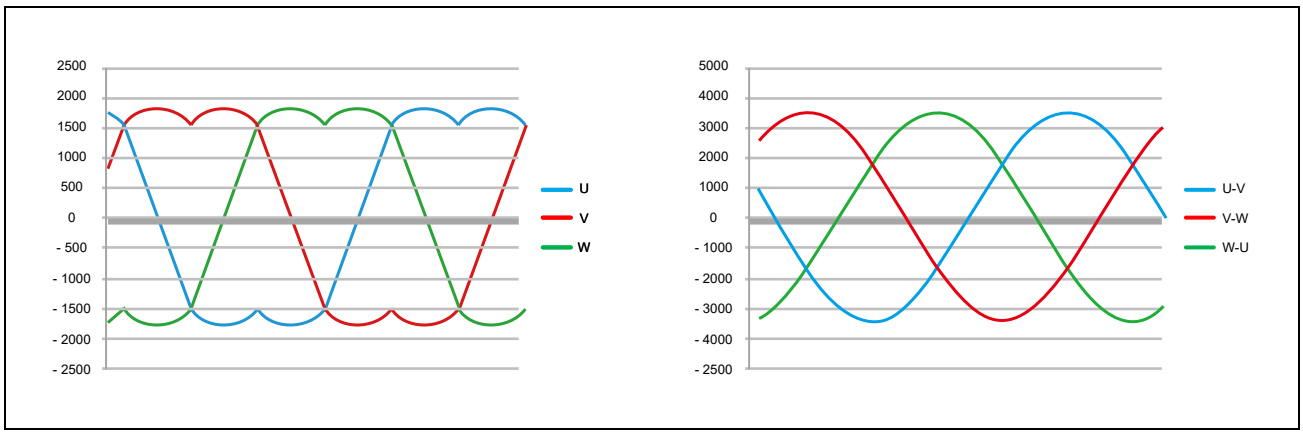


Figure 25.14 3-phase Waveforms with Modulation (Left: 3-phase Voltage Waveforms, Right: Line Voltage Waveforms)

The PWM IP can generate the following 8 types of waveforms according to the SETHARM bit settings of the EMU3nPWMCTR register:

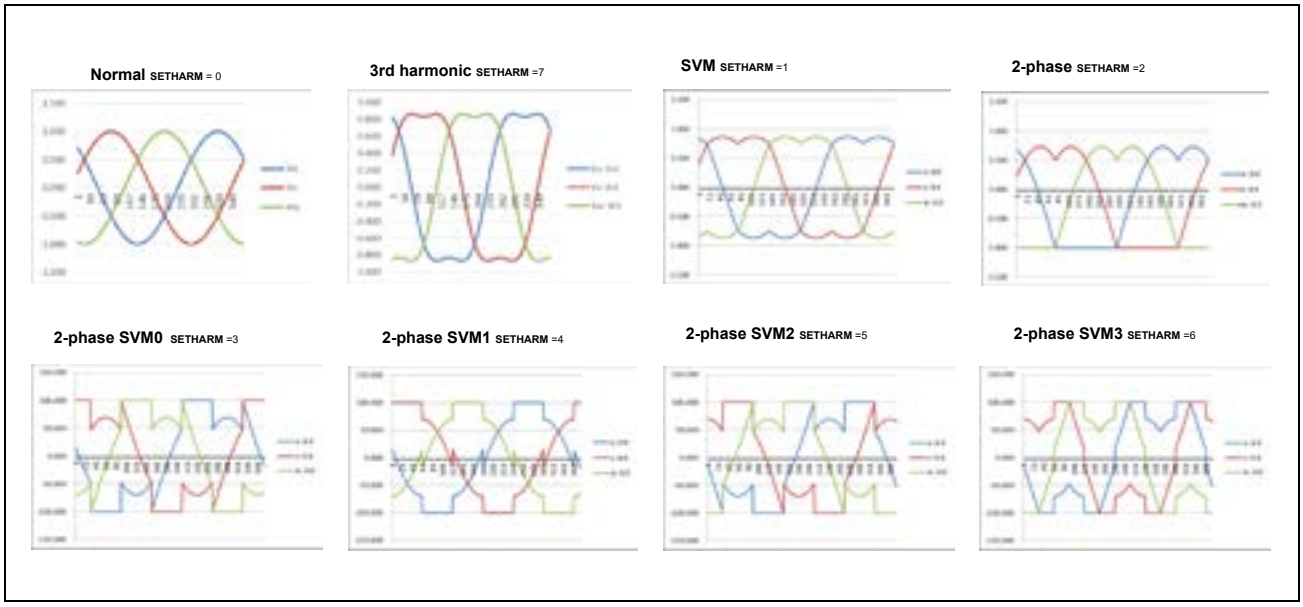


Figure 25.15 Waveforms that can be Generated according to the SETHARM Bit Settings

The peak value of the 2-phase SVM can be made greater by setting the EMU3nTMAX register with a ratio value greater than the value of the EMU3nVDQSFT register (dq-axis voltage value: Vdq).

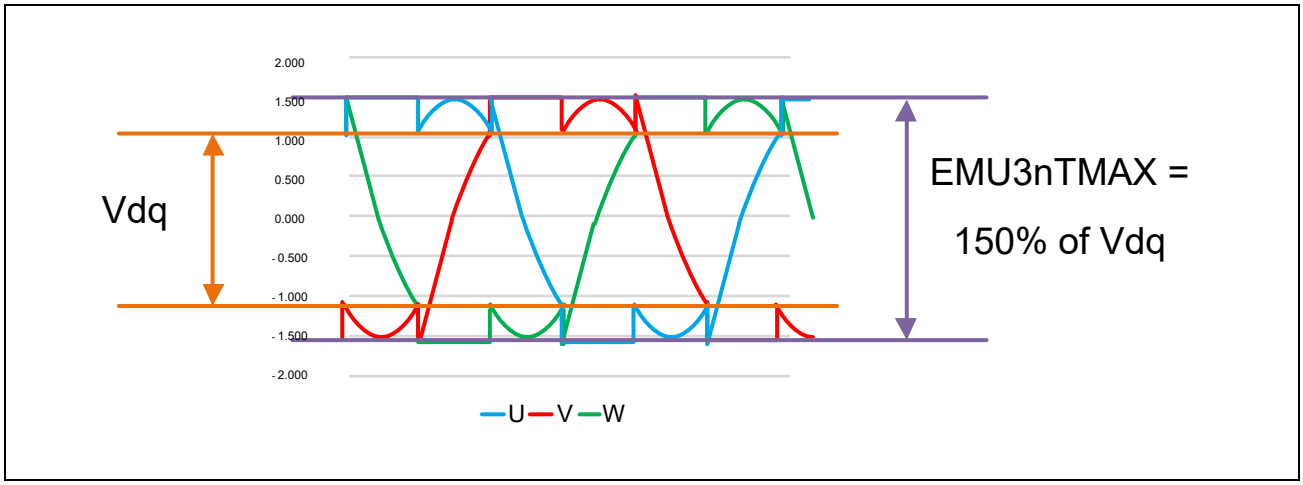


Figure 25.16 Example of Setting up the Peak Value using the EMU3nTMAX Register

Furthermore, the peak value can also be made greater by changing the coefficient value that is set in the EMU3nSR23 register.

(5) Func(pwm5) Duty factor calculation

[pwm5] Duty factor calculation processing

Explanation	Multiplies the UVW phase voltage values by the gain specified through a register for making scaling adjustment.
Arguments	*vui*: Input U phase voltage value (EMU3nVU0) **vuo*: Output U phase voltage value (EMU3nVU1) *vvi*: Input V phase voltage value (EMU3nVV0) **vvo*: Output V phase voltage value (EMU3nVV1) *vwi*: Input W phase voltage value (EMU3nVW0) **vwo*: Output W phase voltage value (EMU3nVW1)
Related registers	EMU3nPWMK1 : Digit position alignment 1 register (uint32_t) EMU3nVOLV : Input voltage register (int16_t)

Processing	void pwm5 (int32_t *vui*, int32_t *vvi*, int32_t *vwi*,
	int32_t **vuo*, int32_t **vvo*, int32_t **vwo*) {
	
	**vuo* = ((*vui* * EMU3nPWMK1) >> 16) / EMU3nVOLV ;
	**vvo* = ((*vvi* * EMU3nPWMK1) >> 16) / EMU3nVOLV ;
	**vwo* = ((*vwi* * EMU3nPWMK1) >> 16) / EMU3nVOLV ;
	}

(6) Func(pwm6) Offset addition

[pwm6] Offset addition processing

Explanation	Adds a given value from a register to correct the UVW phase voltage values.
Arguments	*vui*: Input U phase voltage value (EMU3nVU1) **vuo*: Output U phase voltage value (EMU3nVU2) *vvi*: Input V phase voltage value (EMU3nVV1) **vvo*: Output V phase voltage value (EMU3nVV2) *vwi*: Input W phase voltage value (EMU3nVW1) **vwo*: Output W phase voltage value (EMU3nVW2)
Related registers	EMU3nVUOFS: U phase output voltage correction amount register (int16_t) EMU3nVVOFS: V phase output voltage correction amount register (int16_t) EMU3nVWOFS: W phase output voltage correction amount register (int16_t)

Processing	void pwm6 (int32_t *vui*, int32_t *vvi*, int32_t *vwi*,
	int32_t **vuo*, int32_t **vvo*, int32_t **vwo*) {
	
	**vuo* = *vui* + EMU3nVUOFS ;
	**vvo* = *vvi* + EMU3nVVOFS ;
	**vwo* = *vwi* + EMU3nVWOFS ;
	}

(7) Func(pwm8) Output voltage selection and limit processing

[pwm8] Output voltage selection and limit processing

Explanation	Clip the UVW phase data based on the upper- and lower-limit values specified in registers.
Arguments	*vu1i*: Input U phase voltage value (EMU3nVU2) **vuo*: Output U phase voltage value (EMU3nVUFIX) *vv1i*: Input V phase voltage value (EMU3nVV2) **vvo*: Output V phase voltage value (EMU3nVVFIX) *vw1i*: Input W phase voltage value (EMU3nVW2) **vwo*: Output W phase voltage value (EMU3nVWFIX)
Related registers	EMU3nDTUL: Duty factor upper limit register (int32_t) EMU3nDTLL: Duty factor lower limit register (int32_t)

Processing	void pwm8 (int32_t *vu1i*, int32_t *vv1i*, int32_t *vw1i*,
	int32_t **vuo*, int32_t **vvo*, int32_t **vwo*) {
	
	int32_t tmpU, tmpV, tmpW ;
	
	tmpU = *vu1i*;
	tmpV = *vv1i*;
	tmpW = *vw1i*;
	
	if (tmpU > EMU3nDTUL) **vuo* = EMU3nDTUL;
	else if(tmpU < EMU3nDTLL) **vuo* = EMU3nDTLL;
	else **vuo* = tmpU;
	if (tmpV > EMU3nDTUL) **vvo* = EMU3nDTUL;
	else if(tmpV < EMU3nDTLL) **vvo* = EMU3nDTLL;
	else **vvo* = tmpV;
	if (tmpW > EMU3nDTUL) **vwo* = EMU3nDTUL;
	else if(tmpW < EMU3nDTLL) **vwo* = EMU3nDTLL;
	else **vwo* = tmpW;
	
	}

(8) Func(pwm9) PWM value calculation processing

[pwm9] PWM value calculation processing

Explanation	Calculates the PWM compare values from the duty factor values for the uvw phases.
Arguments	*vui*: Input U phase voltage value (EMU3nVUFIX) *vvi*: Input V phase voltage value (EMU3nVVFIX) *vwi*: Input W phase voltage value (EMU3nVWFIX) **pwmuo*: Output U phase PWM compare value (EMU3nPWMU0) **pwmvo*: Output V phase PWM compare value (EMU3nPWMV0) **pwmwo*: Output W phase PWM compare value (EMU3nPWMW0)
Related registers	EMU3nPWMCTR.PWMSEL : PWM arithmetic reference value select bit(0 or 1) EMU3nPWMCTR.PWMOP : PWM directive value operation select bits(0,1,2) EMU3nCARR : Carrier period register (uint16_t) EMU3nDTT : Dead time setting register (uint16_t) EMU3nPWMK2 : Digit position alignment 2 register (int16_t)

Processing	void pwm9 (int32_t *vui*, int32_t *vvi*, int32_t *vwi*,
	int32_t **pwmuo*, int32_t **pwmvo*, int32_t **pwmwo*) {
	
	uint16_t tmp_base, tmp_zero;
	uint32_t tmpU, tmpV, tmpW;
	
	if(EMU3nPWMCTR.PWMSEL == 0)
	tmp_base = EMU3nCARR >>1;
	else
	tmp_base = (EMU3nCARR + EMU3nDTT) >>1;
	
	if(EMU3nPWMCTR.PWMOP == 2)
	tmp_zero = 0;
	else
	tmp_zero = (EMU3nCARR + EMU3nDTT) >>1;
	
	tmpU = ((*vui* * (tmp_base * EMU3nPWMK2)) >> 16) ;
	tmpV = ((*vvi* * (tmp_base * EMU3nPWMK2)) >> 16) ;
	tmpW = ((*vwi* * (tmp_base * EMU3nPWMK2)) >> 16) ;
	
	if(EMU3nPWMCTR.PWMOP == 0) {
	**pwmuo* = tmp_zero + tmpU ;
	**pwmvo* = tmp_zero + tmpV ;
	**pwmwo* = tmp_zero + tmpW ;
	} else if(EMU3nPWMCTR.PWMOP == 1) {
	**pwmuo* = tmp_zero - tmpU ;
	**pwmvo* = tmp_zero - tmpV ;
	**pwmwo* = tmp_zero - tmpW ;
	} else if(EMU3nPWMCTR.PWMOP == 2) {
	**pwmuo* = tmpU ;
	**pwmvo* = tmpV ;
	**pwmwo* = tmpW ;
	}
	}

(9) Func(pwm10) Dead time compensation

[pwm10] Dead time compensation processing

Explanation	Adds values given through registers according to the polarity of the current values to the UVW phase PWM compare values.
Arguments	*pwmui*: Input U phase PWM compare value (EMU3nPWMU0) *pwmvi*: Input V phase PWM compare value (EMU3nPWMV0) *pwmwi*: Input W phase PWM compare value (EMU3nPWMW0) *iui*: Input U phase current value (EMU3nIUFIX) *ivi*: Input V phase current value (EMU3nIVFIX) *iwi*: Input W phase current value (EMU3nIWFIX) **pwmuo*: Output U phase PWM compare value (EMU3nPWMUDT) **pwmvo*: Output V phase PWM compare value (EMU3nPWMVDT) **pwmwo*: Output W phase PWM compare value (EMU3nPWMWDT)
Related registers	EMU3nPWMCTR.SETDTO : Dead time compensation processing setting bit (0 or 1) EMU3nDTOTH : Dead time compensation threshold value register (uint31_t) EMU3nDTOPV : Dead time compensation positive current time addend register (int16_t) EMU3nDTONV : Dead time compensation negative current time addend register (int16_t)

Processing

```
void pwm10 ( int32_t  pwmui,  int32_t  pwmvi,  int32_t  pwmwi,
             int32_t  iui,  int32_t  ivi,  int32_t  iwi,
             int32_t *pwmuo,  int32_t *pwmvo,  int32_t *pwmwo) {

  if ( (iui >  EMU3nDTOTH) && EMU3nPWMCTR.SETDTO == 1 )
    *pwmuo = pwmui + EMU3nDTOPV;
  else if( (iui < -EMU3nDTOTH) && EMU3nPWMCTR.SETDTO == 1 )
    *pwmuo = pwmui + EMU3nDTONV;
  else
    *pwmuo = pwmui;

  if ( (ivi >  EMU3nDTOTH) && EMU3nPWMCTR.SETDTO == 1 )
    *pwmvo = pwmvi + EMU3nDTOPV;
  else if( (ivi < -EMU3nDTOTH) && EMU3nPWMCTR.SETDTO == 1 )
    *pwmvo = pwmvi + EMU3nDTONV;
  else
    *pwmvo = pwmvi;

  if ( (iwi >  EMU3nDTOTH) && EMU3nPWMCTR.SETDTO == 1 )
    *pwmwo = pwmwi + EMU3nDTOPV;
  else if( (iwi < -EMU3nDTOTH) && EMU3nPWMCTR.SETDTO == 1 )
    *pwmwo = pwmwi + EMU3nDTONV;
  else
    *pwmwo = pwmwi;

}
```

Processing description: Dead time compensation processing

The PWM IP is provided with the function to perform dead time compensation processing for the purpose of compensating for the dead time that is added.

Figure 25.17 shows the possible influence that the dead time exerts on the output of an inverter. Since the potential in the dead time period varies according to the positive/negative state of the motor current, the output waveform may get longer or shorter than the ideal waveform represented by the 3-phase PWM compare values. The PWM IP can make the inverter output closer to the ideal waveform by adding correction value to the PWM compare values in advance according to the positive or negative state of the current values. Depending on the threshold value, the addition of the correction value is cancelled if the motor current values lie near the value of 0.

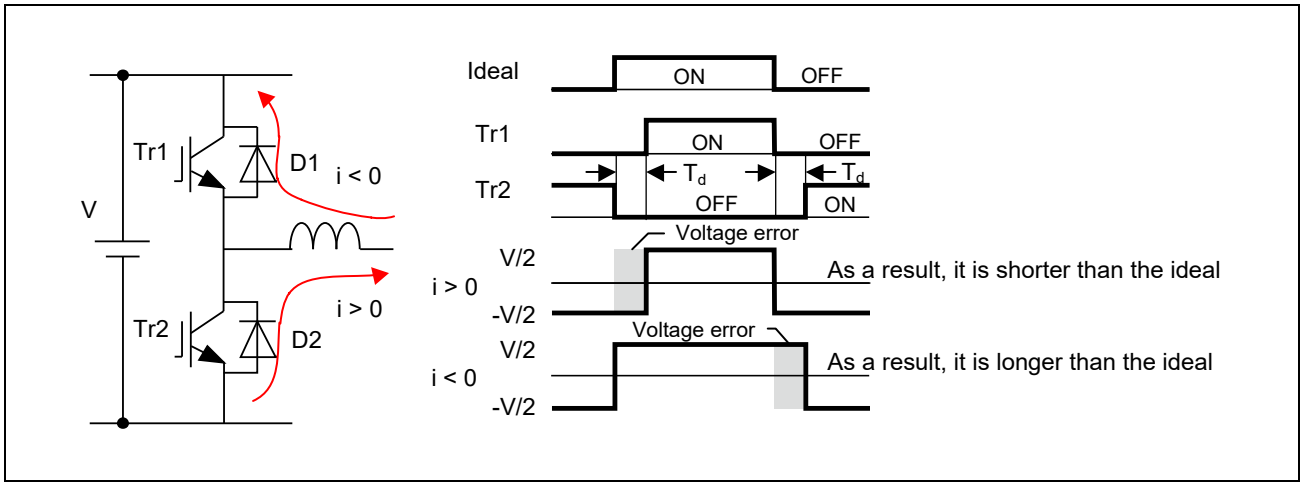


Figure 25.17 Dead Time Adding Processing by TSG3

Table 25.179 gives sample register settings.

Table 25.179 Sample Register Settings for Dead Time Compensation Processing

Register	Use Dead Time Compensation	Do not Use Dead Time Compensation
EMU3nPWMCTR.SETDTO	1	0
EMU3nDTOTH	Set the threshold value	Value after reset
EMU3nDTOPV	Set a correction value	Value after reset
EMU3nDTONV	Set a negative correction value.	Value after reset

(10) Func(pwm11) PWM value limit processing

[pwm11] PWM value limit processing

Explanation	Clips the PWM compare values based on the upper- and lower-limit values specified through registers.
Arguments	*pwmui*: Input U phase PWM compare value (EMU3nPWMUDT) *pwmvi*: Input V phase PWM compare value (EMU3nPWMVDT) *pwmwi*: Input W phase PWM compare value (EMU3nPWMWDT) **pwmuo*: Output U phase PWM compare value (EMU3nPWMUIP) **pwmvo*: Output V phase PWM compare value (EMU3nPWMVIP) **pwmwo*: Output W phase PWM compare value (EMU3nPWMWIP) **carro*: Output carrier period value (EMU3nCARRBUF)
Related registers	EMU3nCARR : Carrier period register(uint16_t) EMU3nDTT : Dead time setting register (uint16_t) EMU3nPWMUL : PWM upper-limit value register (uint16_t) EMU3nPWMLL : PWM lower-limit value register (uint16_t) EMU3nVOLV : Input voltage register (int16_t)

Processing	void pwm11 (int32_t *pwmui*, int32_t *pwmvi*, int32_t *pwmwi*,
	uint16_t **pwmuo*, uint16_t **pwmvo*, uint16_t **pwmwo*, uint16_t *carro) {
	
	int16_t pwm_max = EMU3nCARR + EMU3nDTT - EMU3nPWMUL;
	if(EMU3nVOLV == 0)
	*pwmuo = (EMU3nCARR + EMU3nDTT) >>1 ;
	else if(pwmui >= pwm_max)
	*pwmuo = EMU3nCARR + EMU3nDTT;
	else if(pwmui <= EMU3nPWMLL)
	*pwmuo = 0;
	else
	*pwmuo = pwmui;
	if(EMU3nVOLV == 0)
	*pwmvo = (EMU3nCARR + EMU3nDTT) >>1 ;
	else if(pwmvi >= pwm_max)
	*pwmvo = EMU3nCARR + EMU3nDTT;
	else if(pwmvi <= EMU3nPWMLL)
	*pwmvo = 0;
	else
	*pwmvo = pwmvi;
	if(EMU3nVOLV == 0)
	*pwmwo = (EMU3nCARR + EMU3nDTT) >>1 ;
	else if(pwmwi >= pwm_max)
	*pwmwo = EMU3nCARR + EMU3nDTT;
	else if(pwmwi <= EMU3nPWMLL)
	*pwmwo = 0;
	else
	*pwmwo = pwmwi;
	*carro = EMU3nCARR;
	}

(11) Func(pwm_out) PWM value output processing

[pwm_out] PWM value output processing

Explanation	Outputs the PWM compare values to outside the H/W accelerator. The pwm_out function is independent of any PWM IP operations and runs even when the PWM IP is stopped.
Arguments	*pwmui*: Input U phase PWM compare value (EMU3nPWMUIP) *pwmvi*: Input V phase PWM compare value (EMU3nPWMVIP) *pwmwi*: Input W phase PWM compare value (EMU3nPWMWIP) *carri*: Input carrier period value (EMU3nCARRBUF) **pwmuo*: Output U phase PWM compare value **pwmvo*: Output V phase PWM compare value **pwmwo*: Output W phase PWM compare value **carro*: Output carrier period value
Related registers	EMU3nPWMCTR.SETPWM: PWM setting bit (0 or 1) EMU3nPWMCTR.SHIPWM: Output carrier/PWM compare value shift bit (0 or 1) EMU3nPWMDT: PWM compare value software transfer trigger bit (0 or 1) EMU3nPWMU: U phase PWM compare value software input register (uint16_t) EMU3nPWMV: V phase PWM compare value software input register (uint16_t) EMU3nPWMW: W phase PWM compare value software input register (uint16_t) EMU3nCARR: Carrier period register(uint16_t)

Processing	void pwm_out (uint16_t *pwmui*, uint16_t *pwmvi*, uint16_t *pwmwi*, uint16_t *carri*,
	uint18_t **pwmuo*, uint18_t **pwmvo*, uint18_t **pwmwo*, uint18_t **carro*) {
	
	uint16_t tmpu, tmpv, tmpw, tmpc;
	uint18_t tmpsu, tmpsv, tmpsw, tmpsc;
	
	if (EMU3nPWMCTR.SETPWM == 0) { // Output carrier/PWM compare value software input
	tmpu = EMU3nPWMU;
	tmpv = EMU3nPWMV;
	tmpw = EMU3nPWMW;
	tmpc = EMU3nCARR;
	} else {
	tmpu = pwmui;
	tmpv = pwmvi;
	tmpw = pwmwi;
	tmpc = carri;
	}
	if (EMU3nPWMCTR.SHIPWM == 1) { // Shift the output carrier/PWM compare value1 bit to the left.
	tmpsu = tmpu << 1;
	tmpsv = tmpv << 1;
	tmpsw = tmpw << 1;
	tmpsc = tmpc << 1;
	} else {
	tmpsu = tmpu;
	tmpsv = tmpv;
	tmpsw = tmpw;
	tmpsc = tmpc;
	}
	if ((EMU3nPWMCTR.SETPWM == 0 && EMU3nPWMDT == 1) ||
	(EMU3nPWMCTR.SETPWM == 1 && "PWM IP completion")) {

```
        *pwmuo = tmpsu;
        *pwmvo = tmpsv;
        *pwmwo = tmpsw;
        *carro = tmpsc;
    }
}
```

Processing description: PWM value output processing

The value of the EMU3nCARR register is buffered into the inside of the PWM IP when the PWM IP is started.

The buffered value is transferred to the TSG3 upon completion of the PWM IP processing as the value of the EMU3nCARRBUF register, which holds the data output from the H/W arithmetic block Func(pwm11), together with the values of the EMU3nPWMUIP, EMU3nPWMVIP, and EMU3nPWMWIP registers at the same time.

In the PWM data transfer (software startup) started by the PWMDT bit of the EMU3nPWMDT register, the values of the EMU3nCARR, EMU3nPWMU, EMU3nPWMV, and EMU3nPWMW registers are also transferred to the TSG3 at the same time.

The register values transfer to the TSG3 are, depending on the value of the SHIPWM bit of the EMU3nPWMCTR register, output as they are or shifted by 1 bit to the left as in the table below.

Table 25.180 Relationship between the TSG3 Registers and the EMU3 Registers that are Involved in TSG3 Transfer

TSG3 Register	EMU3nPWMCTR.SHIPWM = 0	EMU3nPWMCTR.SHIPWM = 1
TSG3nCMP0E[17:0]	EMU3nCARRBUF[15:0] (EMU3nCARR[15:0])	EMU3nCARRBUF[15:0] << 1 (EMU3nCARR[15:0] << 1)
TSG3nCMP2E[17:0] TSG3nCMP1E[17:0]	EMU3nPWMUIP[15:0] (EMU3nPWMU[15:0])	EMU3nPWMUIP[15:0] << 1 (EMU3nPWMU[15:0] << 1)
TSG3nCMP6E[17:0] TSG3nCMP5E[17:0]	EMU3nPWMVIP[15:0] (EMU3nPWMV[15:0])	EMU3nPWMVIP[15:0] << 1 (EMU3nPWMV[15:0] << 1)
TSG3nCMP10E[17:0] TSG3nCMP9E[17:0]	EMU3nPWMWIP[15:0] (EMU3nPWMW[15:0])	EMU3nPWMWIP[15:0] << 1 (EMU3nPWMW[15:0] << 1)

Note: Items enclosed in parentheses in lower columns represent EMU3 registers associated with PWM data transfer (software startup).

25.4.6 Rectangle IP

The rectangle IP calculates the rectangle wave output level and compare values. It can also input the voltage phases. **Figure 25.18** shows the flow of rectangle IP processing.

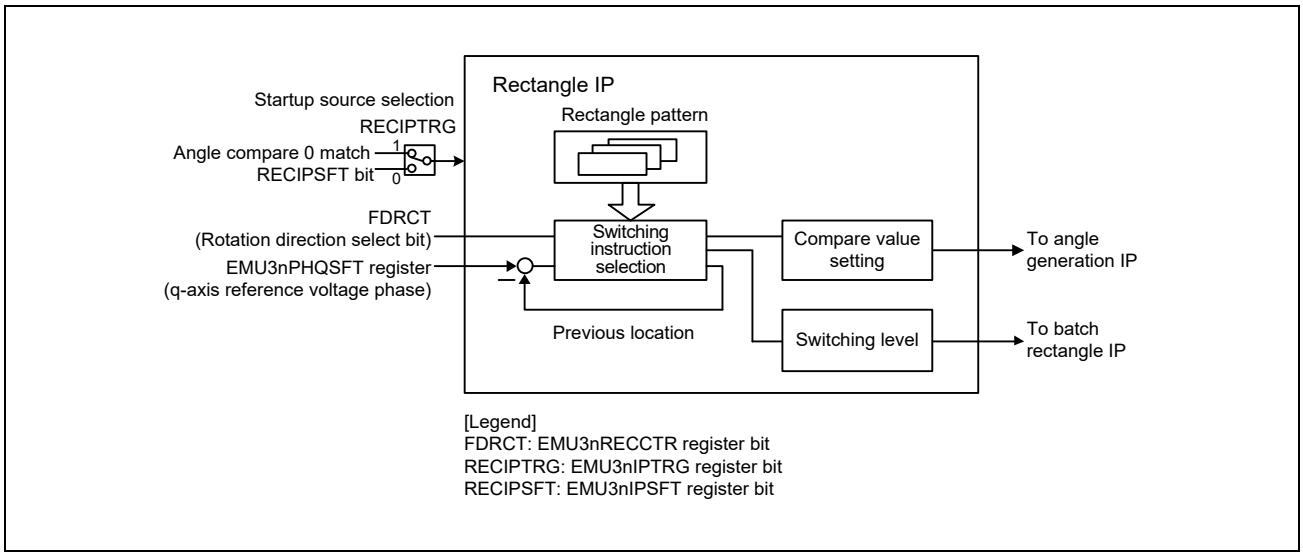


Figure 25.18 Flow of Rectangle IP Processing

The rectangle IP starts when an angle compare 0 match is detected if the RECIPTRG bit of the EMU3nIPTRG register is set to 1. If the RECIPTRG bit is set to 0, the rectangle IP can be started by setting the RECIPSFT bit of the EMU3nIPSFT register to 1.

The RECIF bit of the EMU3nINTSD register is set to 1 when the rectangle IP processing is completed.

The RECIF bit can be cleared by setting the RECIFC bit of the EMU3nINTSDC register to 1.

The completion of rectangle IP processing can be used as a trigger to generate an interrupt.

(1) Switching instruction generation

"Feedback data select" FIPPOSI bit of the EMU3nRECCTR register	psw_old_u[2:0]
"0" (Use EMU3 calculation value)	EMU3nPSW_u[2:0]
"1" (Use user-supplied value)	EMU3nPSWSFT_u[2:0]

$\phi q\_s[11:0] \leftarrow EMU3nPHQSFT\_s[11:0]$

Direction of rotation FDRCT bit of the EMU3nRECCTR register	psw_old_u[2:0]	EMU3nPSW_u[2:0]	Switching angle EMU3nIPCMP0_u[11:0]	Next switching
"0" (Positive rotation)	0	1	60° – ϕq_s[11:0]	EMU3nPTNAB A side
	1	2	120° – ϕq_s[11:0]	EMU3nPTNAB B side
	2	3	180° – ϕq_s[11:0]	EMU3nPTNCD C side
	3	4	240° – ϕq_s[11:0]	EMU3nPTNCD D side
	4	5	300° – ϕq_s[11:0]	EMU3nPTNEF E side
	5	0	360° – ϕq_s[11:0]	EMU3nPTNEF F side
"1" (Reverse rotation)	0	5	180° + ϕq_s[11:0]	EMU3nPTNEF E side
	1	0	240° + ϕq_s[11:0]	EMU3nPTNEF F side
	2	1	300° + ϕq_s[11:0]	EMU3nPTNAB A side
	3	2	360° + ϕq_s[11:0]	EMU3nPTNAB B side
	4	3	60° + ϕq_s[11:0]	EMU3nPTNCD C side
	5	4	120° + ϕq_s[11:0]	EMU3nPTNCD D side

Note 1. ϕ appearing in the text is calculated in degrees but actually calculated using values of 0 to FFF_H.
$60° = 2AA_H$, $120° = 555_H$, $180° = 800_H$, $240° = AAA_H$, $300° = D55_H$, $360°(0°) = 000_H$
EMU3nIPCMP0_u[11:0] is handled as an unsigned 12-bit value by masking the calculation results with $\&0FFF_H$.

25.4.7 Rectangle Wave Generation Block

The rectangle wave generation block consists of batch rectangle IP, independent rectangle IP1, and independent rectangle IP2. The rectangle wave generated here is output via the TSG3. The RECMD bit of the EMU3nIRECCTR register can be used to select the IP of which the rectangle wave is to be output.

The batch rectangle IP updates the output waveforms of the U/V/W phases all at once based on the rectangle wave output level and compare values that are calculated by the rectangle IP in the arithmetic block.

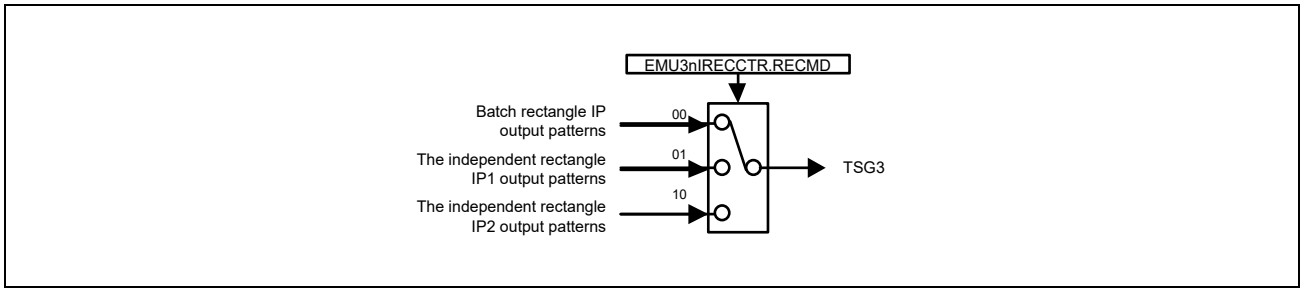


Figure 25.19 Output Waveform Selection Function

The independent rectangle IP1 and independent rectangle IP2 update the output waveforms at different timings for different phases based on the compare values and output levels that are specified separately for each of the U, V, and W phases.

Note: The batch rectangle IP, independent rectangle IP1, and independent rectangle IP2 must not be switched while they are generating rectangle wave outputs.

25.4.7.1 Batch Rectangle IP

The methods of switching the rectangle wave pattern output by the H/W accelerator are divided into hardware processing using the rectangle IP's arithmetic processing and software processing in which register values are rewritten at appropriate times by CPU software.

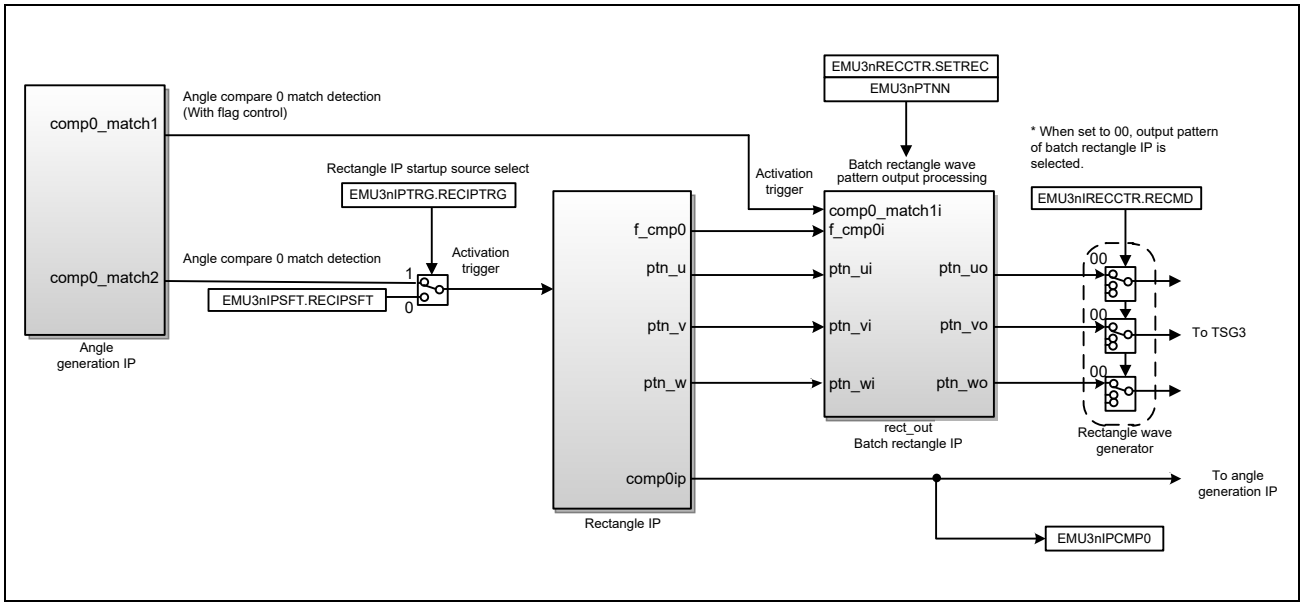


Figure 25.20 Processing from the Angle Generation IP to Rectangle Wave Generation Block

Hardware Processing

The batch rectangle IP selects, every time started for arithmetic processing, the compare value to be used for angle compare 0 match detection and the rectangle wave pattern data that is to be applied on detection of an angle compare 0 match, switch the rectangle wave pattern output with the selected data on each detection of an angle compare 0 match.

To apply the value selected by the batch rectangle IP to compare value established on the detection of an angle compare 0 match, it is necessary to set the SLCTCMP0 bit of the EMU3nRECCTR register to 1.

CAUTION

Even if the SLCTCMP0 bit of the EMU3nRECCTR register is set to 1, the value of the EMU3nCMP0 register is used as the compare value for the first angle compare 0 match processing that is performed after the H/W accelerator is started.

If the FDRCT bit of the EMU3nRECCTR register is set to 0 (positive rotation), the case switches in the order of "0→1→2→3→4→5→0→…" shown in **Table 25.181 (1/2)** on each angle compare 0 match detection occurring every 60°. If the FDRCT bit of the EMU3nRECCTR register is set to 1 (reverse rotation), the case switches in the order of "0→5→4→3→2→1→0→…" shown in **Table 25.181 (2/2)** on each angle compare 0 match detection occurring every 60°.

The selection of the rectangle wave patterns is set by EMU3nPTNAB register, EMU3nPTNCD register, and EMU3nPTNEF register if the SETREC bit of the EMU3nRECCTR register is set to 1. If the SETREC bit of the EMU3nRECCTR register is set to 0, the value of only the EMU3nPTNN is selected and therefore it is necessary for the CPU software to rewrite the EMU3nPTNN register on each detection of an angle compare 0 match.

Table 25.181 Sets of Compare Values for Angle Compare 0 Match Detection and Rectangle Wave Patterns to be selected by the Batch Rectangle IP (1/2)

	* Positive Rotation Mode (FDRCT bit = 0)		
Case*1	EMU3nIPCMP0 register	rectangle wave pattern (SETREC bit = 1)	rectangle wave pattern (SETREC bit = 0)
0	2AA (60°)	EMU3nPTNAB register, nPTNA bits (n=W,V,U)	EMU3nPTNN register, nPTN bits (n=W, V, U)
1	555 (120°)	EMU3nPTNAB register, nPTNB bits (n=W,V,U)	Same as above
2	800 (180°)	EMU3nPTNCD register, nPTNC bits (n=W,V,U)	Same as above
3	AAA (240°)	EMU3nPTNCD register, nPTND bits (n=W,V,U)	Same as above
4	D55 (300°)	EMU3nPTNEF register, nPTNE bits (n=W,V,U)	Same as above
5	000 (0°)	EMU3nPTNEF register, nPTNF bits (n=W,V,U)	Same as above

Note 1. "Case" denotes the value of psw_old_u[2:0]. See **Section 25.4.6, Rectangle IP.**

Table 25.181 Sets of Compare Values for Angle Compare 0 Match Detection and Rectangle Wave Patterns to be selected by the Batch Rectangle IP (2/2)

	* Reverse Rotation Mode (FDRCT bit = 1)		
Case*1	EMU3nIPCMP0 register	Rectangle wave pattern (SETREC bit = 1)	Rectangle wave pattern (SETREC bit = 0)
0	800 (180°)	EMU3nPTNEF register, nPTNE bits (n=W,V,U)	EMU3nPTNN register, nPTN bits (n=W, V, U)
1	AAA (240°)	EMU3nPTNEF register, nPTNF bits (n=W,V,U)	Same as above.
2	D55 (300°)	EMU3nPTNAB register, nPTNA bits (n=W,V,U)	Same as above.
3	000 (0°)	EMU3nPTNAB register, nPTNB bits (n=W,V,U)	Same as above.
4	2AA (60°)	EMU3nPTNCD register, nPTNC bits (n=W,V,U)	Same as above.
5	555 (120°)	EMU3nPTNCD register, nPTND bits (n=W,V,U)	Same as above.

Note 1. “Case” denotes the value of psw_old_u[2:0]. See **Section 25.4.6, Rectangle IP.**

For the operation of the batch rectangle hardware, see **Section 25.4.6, Rectangle IP.**

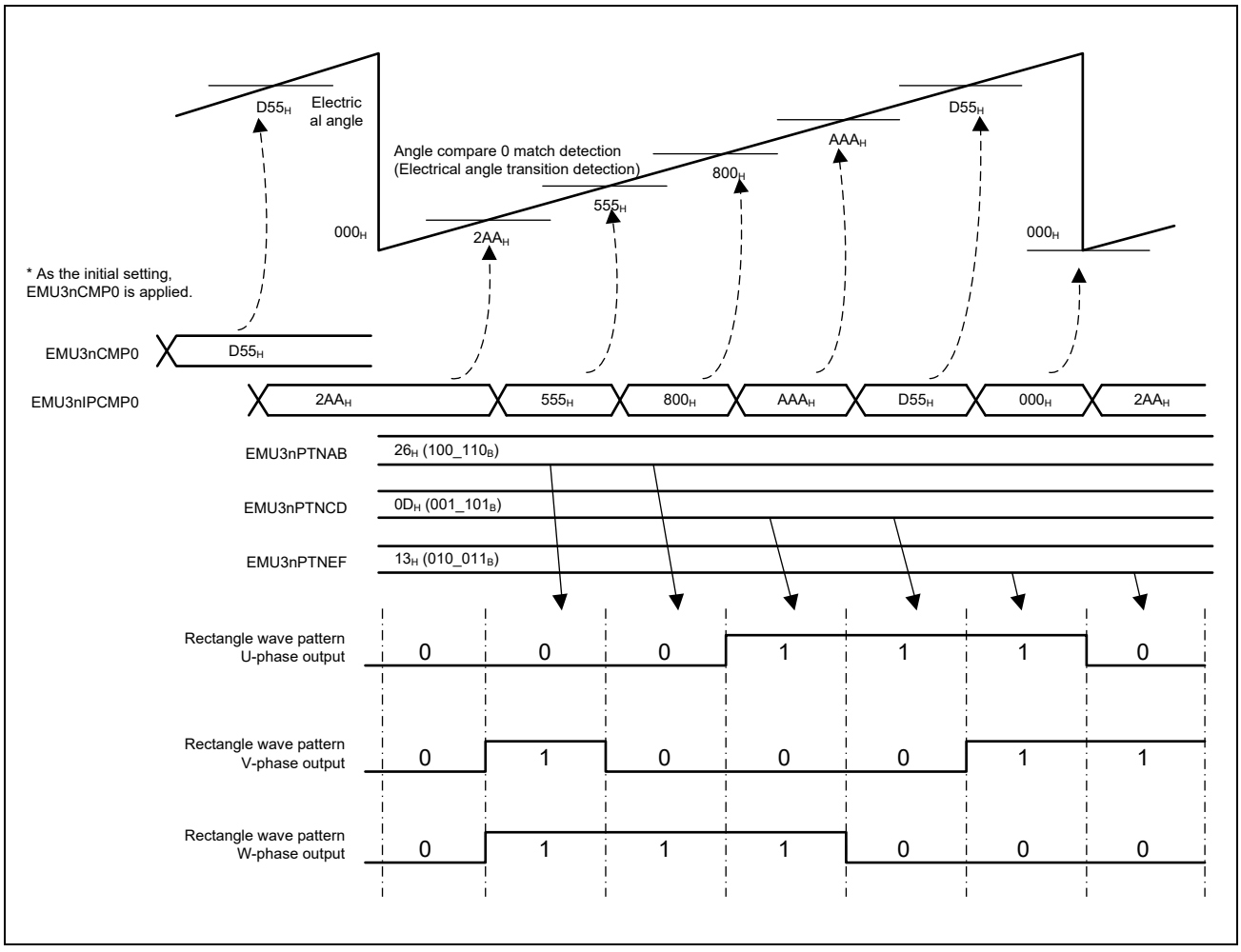


Figure 25.21 Example of Hardware Processing of the Batch Rectangle IP

Software Processing

The EMU3nCMP0 register is configured for the comparison value for angle compare 0 match detection by setting the SLCTCMP0 bit of the EMU3nRECCTR register to 0 and the EMU3nPTNN register is configured for the rectangle wave pattern value by setting the SETREC bit of the EMU3nRECCTR register to 0.

When an angle compare 0 match is detected based on the EMU3nCMP0 register, the rectangle wave pattern is switched by the value that is specified in the EMU3nPTNN register. Software-based rectangle wave pattern switching is accomplished under control of the CPU software by repeating the cycle of processing to rewrite the EMU3nCMP0 register and the EMU3nPTNN register and to wait angle compare 0 match detection.

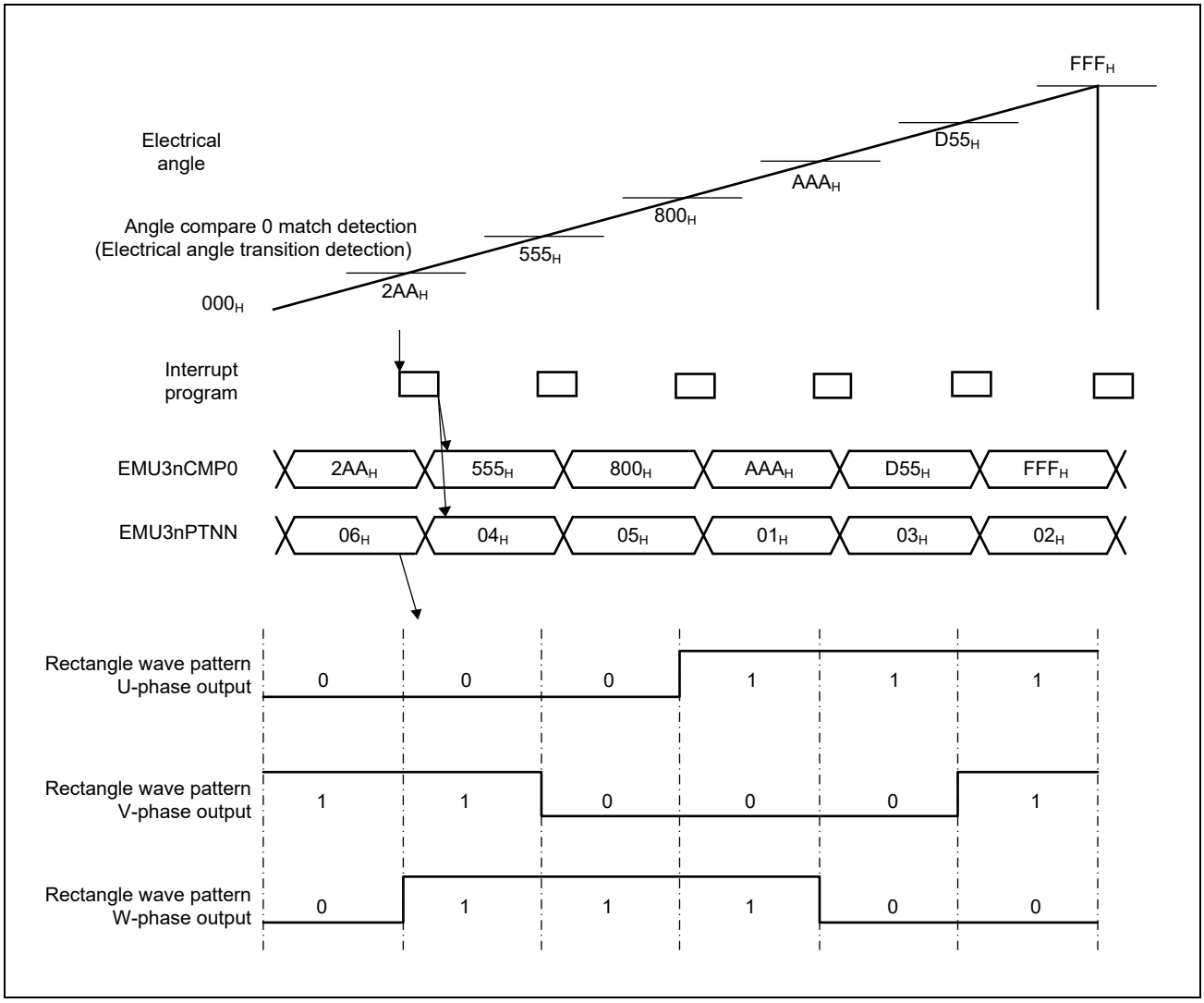


Figure 25.22 Example of Software Processing of the Batch Rectangle IP

25.4.7.2 Independent Rectangle IP1

The independent rectangle IP1 can generate more complex waveforms than the batch rectangle IP by switching each of the U/V/W phases of the rectangle wave patterns when they match the given electrical angles respectively.

The phase switchings in the independent rectangle IP1 occur independently of one another. When a match occurs between the electrical angle and one of the comparison values for angle compare 0 match detection, the output level of the matching phase is switched. Three unique comparison values and pattern values (EMU3nIRkCPPN0 to EMU3nIRkCPPN2 registers (k = U, V, W)) can be set for each of the U/V/W phases. The independent rectangle IP1 switches the comparison value on each detection of an electrical angle compare match. The interrupt source determination bits (IRECWIF, IRECVIF, and IRECUIF bits of the EMU3nINTSD register) are switched to 1 only when electrical angle compare matches are detected based on the comparison values in the EMU3nIRkCPPN2 registers (k = U, V, W).

Independent rectangle IP1 operation

The H/W accelerator outputs the rectangle wave patterns from the independent rectangle IP1 to the TSG3 when the RECMD bits of the EMU3nIRECCTR register are set to 01.

Figure 25.23 shows the configuration of the independent rectangle IP1 for 1 phase. Three circuits of similar configuration are implemented in the independent rectangle IP1 for 3 phases.

The comparison values 0-2 and pattern values 0-2 (EMU3nIRkCPPN0 to EMU3nIRkCPPN2 registers (k = U, V, W)) are rewritten by the CPU. The independent rectangle IP1 selects a set of comparison and pattern values according to an internal selector value. The selector value is set to 001_B after a reset. It switches, on each detection of an electrical angle compare match, in the order of 001_B (compare 0 value/pattern 0 value selected)→010_B (compare 1 value/pattern 1 value selected)→100_B (compare 2 value/pattern 2 value selected)→001_B (compare 0 value/pattern 0 value selected)→…. The interrupt source determination bit is set to 1 when an electrical angle compare match detection is accepted in the select value 100_B state. The current selector value can be referenced through the EMU3nIRSELM register.

Flags 0 to 2 are set to 1 when the CPU write the configuration register (EMU3nIRkCPPN0 to EMU3nIRkCPPN2 (k = U, V, W)) as the comparison value/pattern value and to 0 when the corresponding angle compare match is detected by the angle generation IP. If a flag is set to 0, neither selector value nor pattern output are updated even when the corresponding angle compare match is detected. In this case, no interrupt source will be generated. The state of these flags can be referenced through the EMU3nIRFLGM register.

Flags 0 to 2 are reset to 0 and the select value to 001_B by writing a 1 to the flag select signal initialization register (EMU3nIRCTRST).

The pattern outputs of the respective phases can be updated by writing a value into the UINIPTN, VINIPTN, and WINIPTN bits of the EMU3nIRPTN register. Reading these bits yield the pattern value that has been last written.

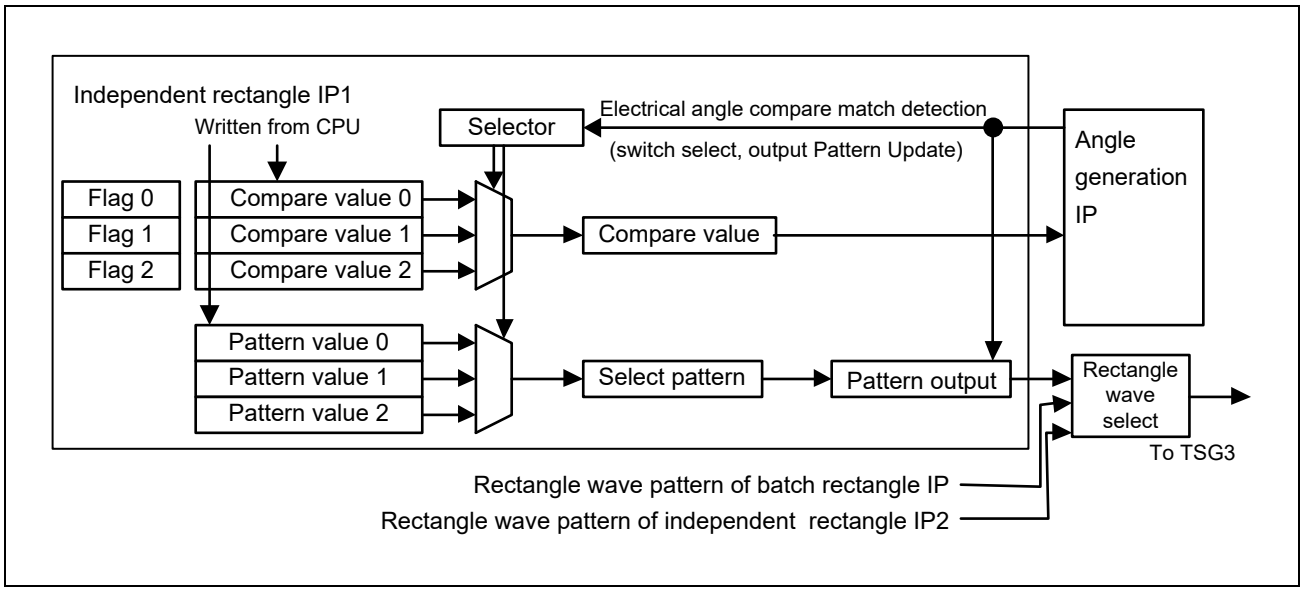


Figure 25.23 Independent Rectangle IP1 Configuration (for 1 Phase)

CAUTION

Since the selector value transits in the order of $001_B \rightarrow 010_B \rightarrow 100_B \rightarrow 001_B \rightarrow \ldots$, it is applied in the order of compare 0 value→compare 1 value→compare 2 value→compare 0 value→…. If a 0 flag occurs in the middle of this processing, however, the transition of the selector value is held pending until that flag is set to 1. In such a case, update the compare value/pattern registers sequentially.

25.4.7.3 Independent Rectangle IP2

The independent rectangle IP2 generates more flexible rectangle wave output patterns by matching 8 compare values which can be separately specified for the U, V, and W phases and the electrical angle.

If the RECMD bits of the MU3nIRECCTR register are set to 10 (enable independent rectangle IP2 waveform output), the independent rectangle IP2 sends its output pattern to the TSG3. When either of U phase compare match, V phase compare match, and W phase compare match occurs, the rectangle output pattern of the corresponding phase is updated.

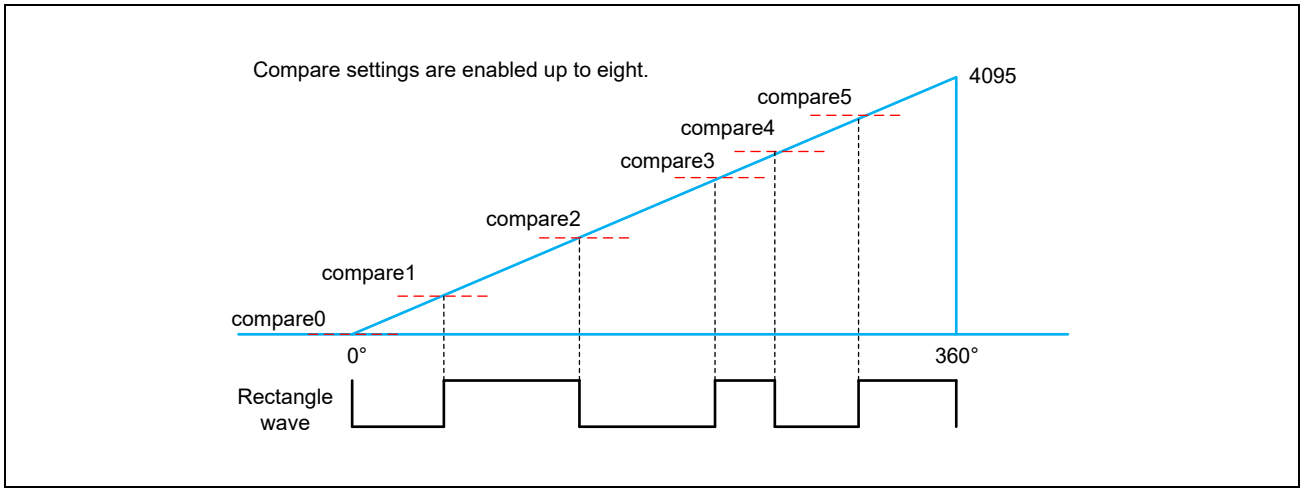


Figure 25.24 Example of Application of the Independent Rectangle IP2

The compare matching of the respective phases can be configured in the EMU3nNRECUk register, EMU3nNRECVk register, and EMU3nNRECWk register.

Table 25.182 Compare Match Settings for the Respective Phases

U phase compare control k register (EMU3nNRECUk) (k = 0-7)	V phase compare control k register (EMU3nNRECVk) (k = 0-7)	W phase compare control k register (EMU3nNRECWk) (k = 0-7)	Function
UPTN bit	VPTN bit	WPTN bit	Sets the output level.
UCMP[11:0]	VCMP[11:0]	WCMP[11:0]	Sets the compare value.
UEN bit	VEN bit	WEN bit	Enables or disables updating of rectangle wave output patterns.
UMOD bit	VMOD bit	WMOD bit	Specifies whether to set UEN (or VEN or WEN) = 0 at every compare match which occurs when UEN (or VEN or WEN) = 1.
UINT bit	VINT bit	WINT bit	Enables or disables compare match interrupts.

A compare match is generated when the electrical angle (previous value) and electrical angle (this time value) inputs, which are output from the angle generation IP, are across the compare value that is set up by the independent rectangle IP2.

Two types of functions to correct the electrical angle (adding an offset) are available for the electrical angle (previous value) and electrical angle (this time value).

The 2 types of offsets may be used at the same time or reconfigured while the independent rectangle IP2 is running.

(Example)
It is possible to make an additional correction on the U phase only while correcting the 3 phases at the same time.

Table 25.183 Settings for Electrical Angle Correction

Register	U phase	V phase	W phase	Function
3-phase common angle correction value register (EMU3nNRECOFSALL)	√	√	√	Angle correction common to 3 phases Adds a common offset value to all of the U, V, and W phases.
U phase angle correction value register (EMU3nNRECOFSU)	√	—	—	U phase angle correction Adds a specific offset to the U phase.
V phase angle correction value register (EMU3nNRECOFSW)	—	√	—	V phase angle correction Adds a specific offset to the V phase.
W phase angle correction value register (EMU3nNRECOFSW)	—	—	√	W phase angle correction Adds a specific offset to the W phase.

CAUTION

No compare match occurs if the spanning of the compare value is caused by the addition of an offset that is made while the independent rectangle IP2 is running.

CAUTION

In cases where multiple valid values for comparison are spanned in the same phase, the output level of the rectangular wave depends on the value of the corresponding pattern value setting bit of a comparison control register.

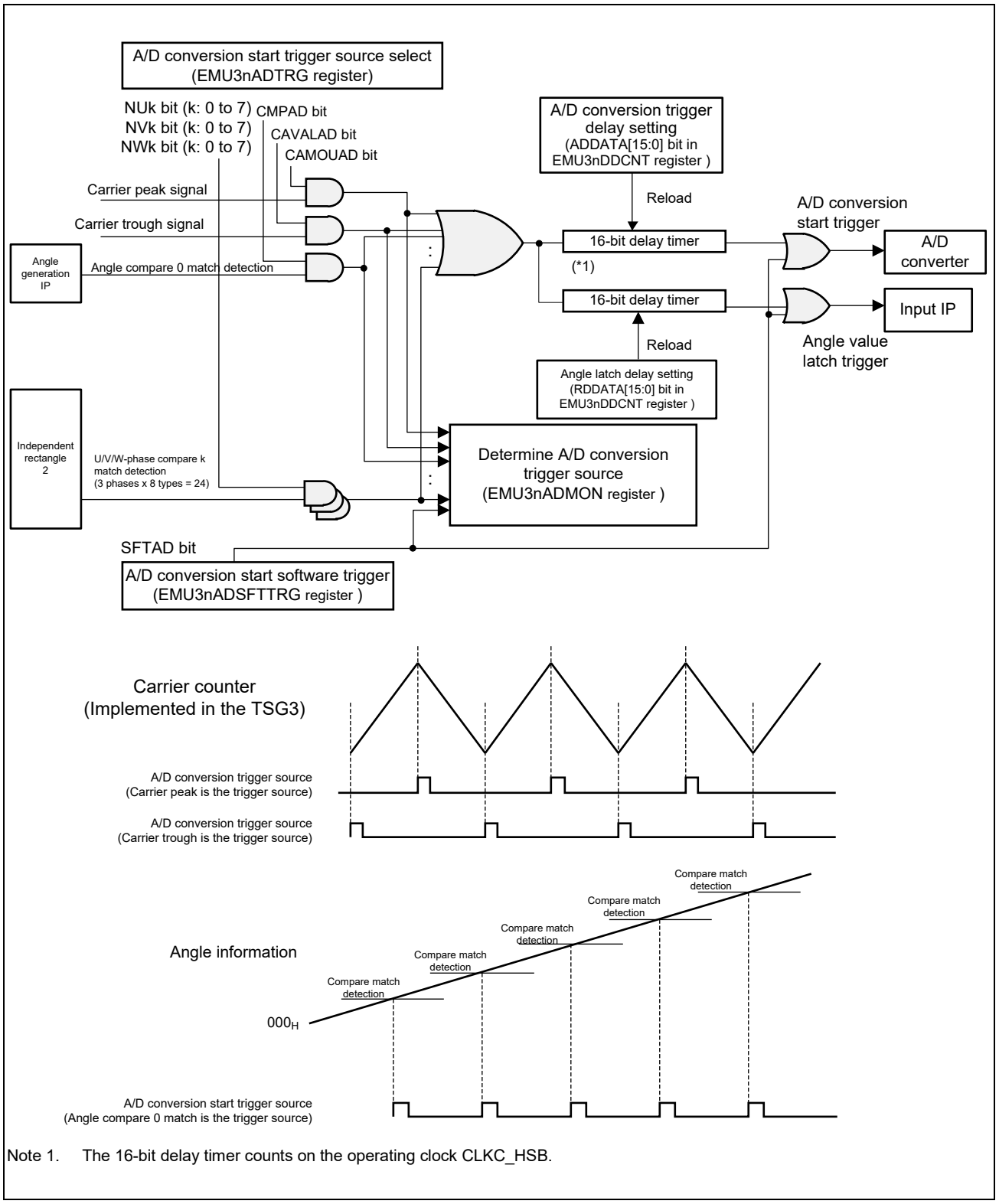
Example: U Phase

- When the pattern value setting bits of multiple valid comparison control registers (EMU3nNRECUk.UPTN) are all 0
  The output level becomes “0” at the time of a match in comparison.
- When the pattern value setting bits of multiple valid comparison control registers (EMU3nNRECUk.UPTN) are all 1
  The output level becomes “1” at the time of a match in comparison.
- When the pattern value setting bits of multiple valid comparison control registers (EMU3nNRECUk.UPTN) have a mixture of the values 0 and 1.
  The output level becomes “1” at the time of a match in comparison.

25.4.8 A/D Conversion Control and Angle Value Latching Control

By configuring the CAMOUAD bit, CAVALAD bit, CMPAD bit, NUk bits, NVk bits, and NWk bits (k = 0 to 7) of the EMU3nADTRG register as A/D conversion start trigger source enable bit, it is possible to generate “A/D conversion start trigger to A/D converter” and “angle value latch trigger to input IP” at the peak and trough of the carrier, on the detection of an angle compare 0 match by the angle generation IP, and on the detection of U/V/W phase compare k matches (3 phases × 8 types = total of 24 lines) by the independent rectangle IP2.

Writing a 1 into the SFTAD bit of the EMU3nADSFTTRG register generates “A/D conversion start trigger to A/D converter” and “angle value latch trigger to input IP” as software startup triggers.



Note 1. The 16-bit delay timer counts on the operating clock CLKC_HSB.

Figure 25.25 Configuration of the Function to Generate the A/D Conversion Start Trigger and Angle Value Latch Trigger

It is possible to determine the source that causes the event by reading the EMU3nADMON register. Any new A/D conversion start trigger sources that occur within the period from the generation of an A/D conversion start trigger source to the end of the conversion are invalidated but the corresponding A/D startup source determination bits of the EMU3nADMON register are set to 1. The source determination bits of the EMU3nADMON register are cleared by setting the corresponding bits of the EMU3nADMONC register to 1.

The delay from the time an A/D conversion start trigger source occurs till the time an A/D conversion start trigger occurs can be extended by setting a value in the ADDATA[15:0] bits of the EMU3nDDCNT register and the delay from the time an A/D conversion start trigger source occurs till the time an angle value latch trigger to input IP occurs can be extended by setting a value into the RDDATA[15:0] bits of the EMU3nDDCNT register. The delay counter is a 16-bit timer which starts counting on the occurrence of an A/D conversion start trigger and generates a trigger when it reaches 0.

25.4.9 IIR Filters

Three channels of quadratic IIR filters are provided to filter out the current values from the A/D converters. Six filter coefficients need be set up for one channel of IIR filter.

A filter startup signal input starts the IIR filtering processing on the current values from the A/D converters. The filtering results (24 bits) are stored in a filtering results register. The filtering results may be shifted to the right as required when they are to be used as post-filtering current values in EMU3.

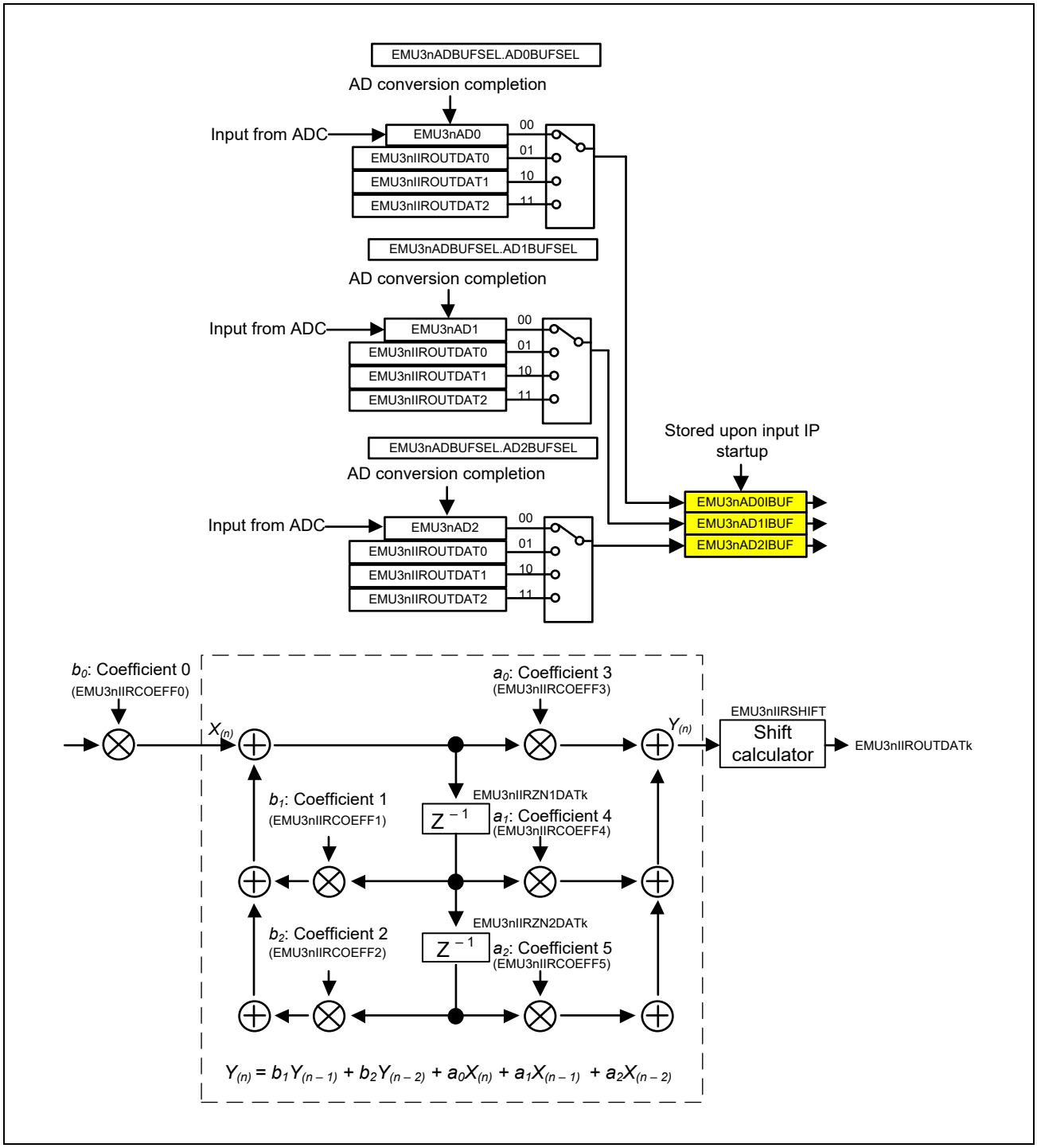
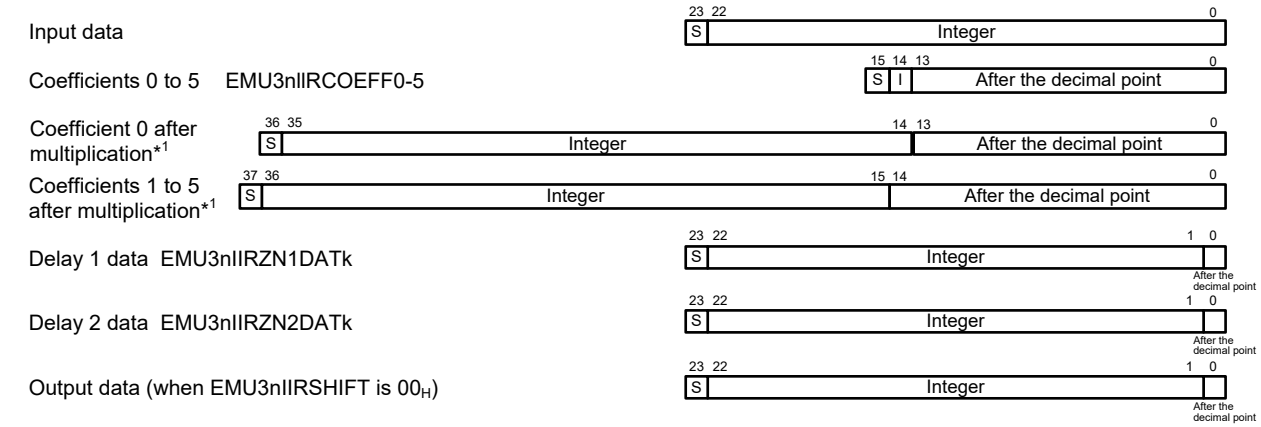


Figure 25.26 Flow of IIR Filtering

The input/output data formats of the IIR filters are shown below.



Note 1. Only the higher-order 24 bits of the data after multiplication are used in the final stage of calculation.

Figure 25.27 The Input/Output Data Formats of the IIR Filters

In addition to the current values from the A/D converters, EMU3nIIRCTRk register can be used to select the input data to the IIR filters, such as software input data and filtering results from other channels of filters, as well as the filtering startup sources.

If a startup trigger is generated when no IIR filtering operations are performed on any of the channels, the channel to be started for IIR filtering processing is selected before filtering is started.

If IIR filtering is in progress on a channel, the new filtering processing is held pending until the current filtering ends.

When triggers are generated for two or more channels, the channels are selected according to their priorities of CH0 > CH1 > CH2.

Table 25.184 Triggers and Input Signals that can be selected using the EMU3nIIRCTRk Register

EMU3nIIRCTRk Register, Bits TRGSEL[3:0]	Startup Trigger
“0000”	IIR filter channel 0 software startup (EMU3nIIRSFT.TRG0)
“0001”	IIR filter channel 1 software startup (EMU3nIIRSFT.TRG1)
“0010”	IIR filter channel 2 software startup (EMU3nIIRSFT.TRG2)
“0100”	A/D converter channel group completion startup
“0101”	A/D converter channel 0 completion.
“0110”	A/D conversion channel 1 completion.
“0111”	A/D conversion channel 2 completion.
“1000”	IIR filter channel 0 completion.
“1001”	IIR filter channel 1 completion.
“1010”	IIR filter channel 2 completion.
Other than above	Not started.

EMU3nIIRCTRk Register, Bits DATSEL[3:0]	Input Signal
“0000”	IIR filter channel 0 software data input value (EMU3nIIRSFTDAT0)
“0001”	IIR filter channel 1 data software input value (EMU3nIIRSFTDAT1)
“0010”	IIR filter channel 2 data software input value (EMU3nIIRSFTDAT2)
“0100”	0 software data input value
“0101”	A/D data channel 0
“0110”	A/D data channel 1
“0111”	A/D data channel 2
“1000”	IIR filter channel 0 operation result data
“1001”	IIR filter channel 1 operation results data
“1010”	IIR filter channel 2 operation result data
Other than above	Data retained.

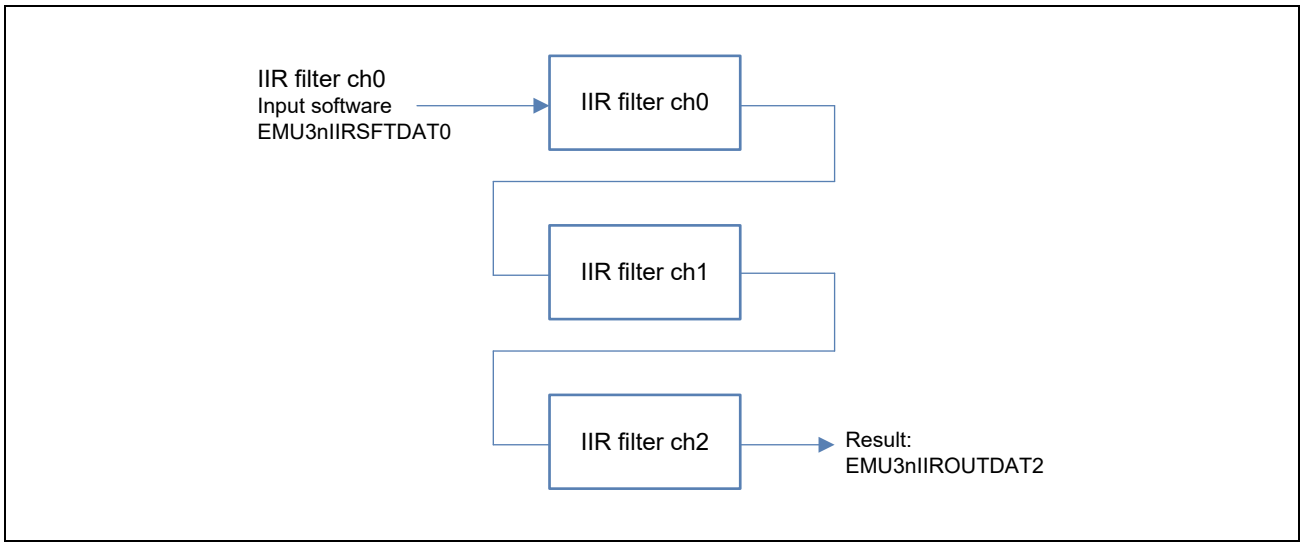


Figure 25.28 Example of IIR Filtering

CAUTION

This facility is provided with 3 input channels and 3 output channels. Its arithmetic block is common to all the channels and therefore performs arithmetic operations for the 3 channels on a time-division basis. The setup of the filtering coefficients and input signal need be completed before the IIR filtering processing is started.

The operations performed before the IIR filtering results are output must handle data values that fit within 24 bits.

25.4.10 Pulse Cycle Measurement Timer

The pulse cycle measurement timer counts the number of pulse cycles per interval between the adjacent rising edges of the Z-phase signal from the R/D converter or the input signal from the Z pin occur within the pulse rising interval.

Table 25.185 Pulse Cycle Measurement Timer Specifications

Item	Specification
Operating clock name	CLKC_HSB
Count start/stop	Controlled by the STR bit of the EMU3nPMTCTR register.
Count operation	• 25-bit up-count • Resets to 0 and continues counting during a capture operation or on an overflow.
Counter read	The 25-bit count value can be read out by reading the EMU3nPMTCNT register.
Counter write	It is possible to write a count value into the counter through the EMU3nPMTCNT register regardless of whether the timer is running or stopped.
Overflow-time operation	Select the destination of the interrupt by setting the OVFSW bit of the EMU3nPMTCTR register. 0: Sets the OVF bit of the EMU3nPMTOF register to 1. 1: Generates an interrupt source signal.
Capture source	• On the rising edge of the Z-phase signal • Software trigger (writing 1 into the STTRG bit of the EMU3nPMTCTR register)
Capture operation	Loads the count value into the EMU3nPMTCAP register on detection of a capture source.
Capture value read	The count captured value can be read from the 25-bit counter by reading the EMU3nPMTCAP register.

CAUTION

Since the pulse period measurement timer runs on the clock CLKC_HSB, it takes some time for the results of writing a register associated with the pulse period measurement timer to be reflected. When reading the latest value of the EMU3nPMTCNT register, EMU3nPMTCAP register, or EMU3nPMTOF register, read the registers after waiting for 10 cycles or more of the CLK_EMU_L clock from completing to write into the timer.

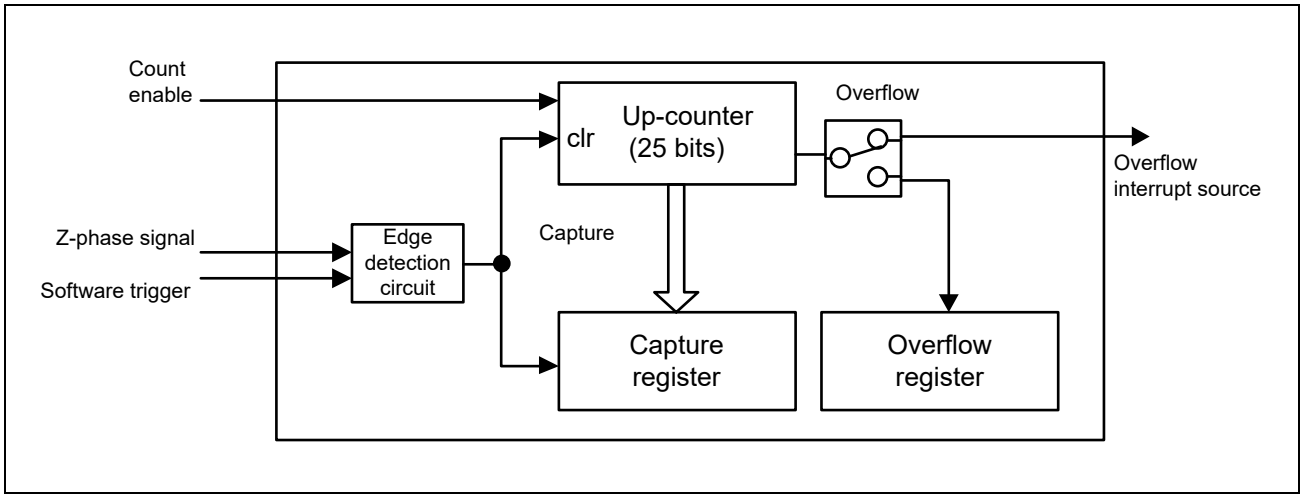


Figure 25.29 Pulse Period Measurement Timer Configuration

When measuring the Z-phase rise interval

When a rising edge of the Z-phase signal pulse is detected, the counter value is stored in the EMU3nPMTCAP register. At this moment, the timer is reset to a count value of 0 and counting is continued.

25.4.11 Resolver Angle Measurement timer

The resolver angle measurement timer counts the number of the number of pulse cycles per interval between the adjacent rising edges or the number of pulse cycles per specified resolver angle interval of the Z-phase signal.

Table 25.186 Resolver Angle Measurement Timer Specifications

Item	Specification
Operating clock name	CLKC_HSB
Count start/stop	Controlled by the EN bit of the EMU3nPMT2CTR register.
Count operation	• 32-bit up-count • Resets to 0 and continues counting during a capture operation or on an overflow.
Counter read	The count value of the 32-bit counter can be read by reading the EMU3nPMT2CNT register.
Counter write	It is possible to write a count value into the counter through the EMU3nPMT2CNT register regardless of whether the timer is running or stopped.
Overflow-time operation	Generates an interrupt source signal.
Capture source	• The interval as every rising edges of the Z-phase signal is selectable through the EMU3nPMT2INVL register. • Software trigger (writing 1 into the SCAPTRG bit of the EMU3nPMT2SFT register)
Capture operation	Loads the count value into the EMU3nPMT2CAP register on detection of a capture source. At this moment, an interrupt source signal is generated.
Capture value read	The count captured value can be read from the 32-bit counter by reading the EMU3nPMT2CAP register.

CAUTION

Since the resolver angle measurement timer runs on the clock CLKC_HSB, it takes some time for the results of writing a register associated with the resolver angle measurement timer to be reflected. When reading the latest value of the EMU3nPMT2CNT register or EMU3nPMT2CAP register, read the registers after waiting for 10 cycles or more of the CLK_EMU_L clock from completing to write into the timer.

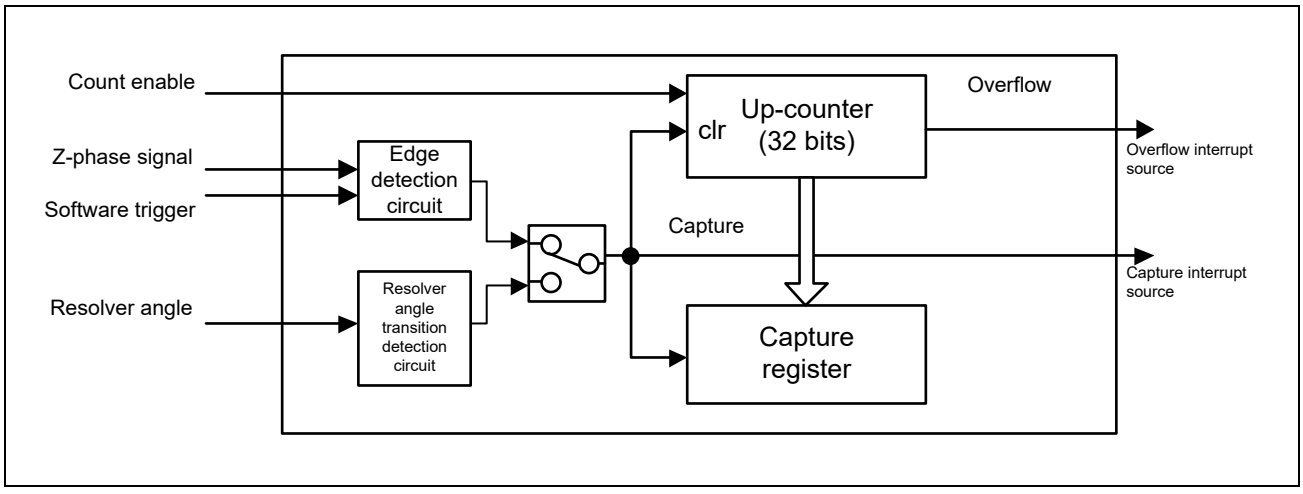


Figure 25.30 Resolver Angle Measurement Timer Configuration

When measuring the Z-phase rise interval

When a rising edge of the Z-phase signal pulse is detected, the counter value is stored in the EMU3nPMT2CAP register. At this moment, the timer is reset to a count value of 0 and counting is continued.

25.4.12 Internal Buffer Registers

Table 25.187 to Table 25.193 show the registers that have internal buffers and their update timings.

The operation of readable/writable registers without buffer may become unstable if they are updated while the IPs are running. Such registers need be updated before starting the IPs. Since the value of a register with a buffer is propagated into the IP at the update timing of each buffer as shown in **Figure 25.31**, the register can be updated at any time even when the IP is running.

When read-only registers without buffer are read while each IP is running, the results of the preceding or current IP processing are read out. Such a register must be read after the IP processing is completed. The value of a register with a buffer is propagated out of the IP when the buffer is updated as shown in **Figure 25.32,** so that the value can be read as the register's value.

The column entitled “simultaneous reflection control” in **Table 25.190** shows the control bit that enables the function to ensure the simultaneity of the updates of more than one register. As shown in **Figure 25.33**, the value is propagated into the IP interior when the PWM IP is started if the FPWMREFPER bit of the EMU3nREFCTR register is set to 1.

For the registers for which “Register set” is indicated in the “Timing” column of **Table 25.193**, if buffering is enabled by the EMU3nCBCTR0 register at the timing specified by the EMU3nCBTIM register, their buffer is updated and its value can be read out as the value of the respective registers.

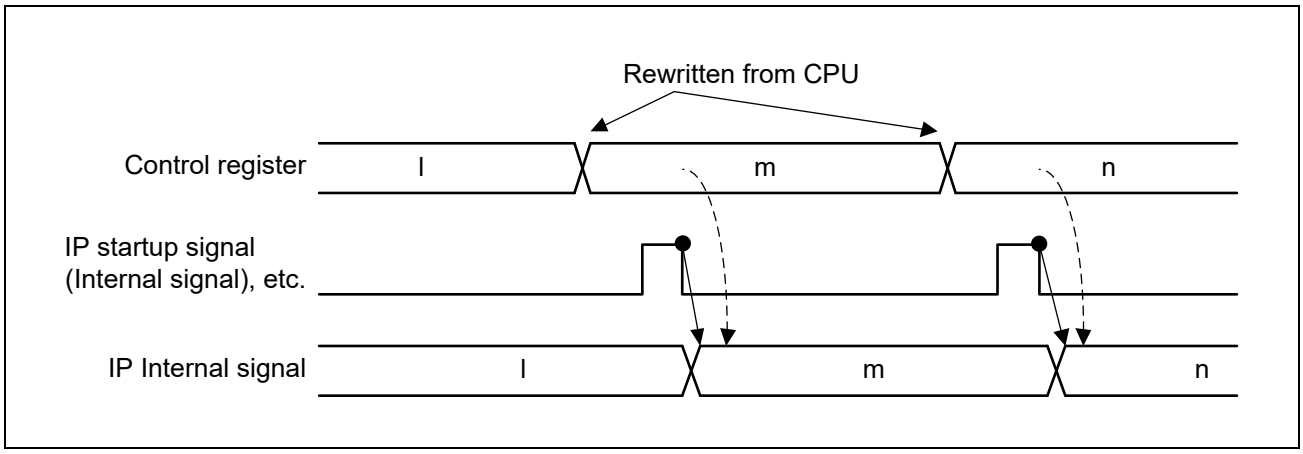


Figure 25.31 Outline of the Buffering of a Readable/Writable Register

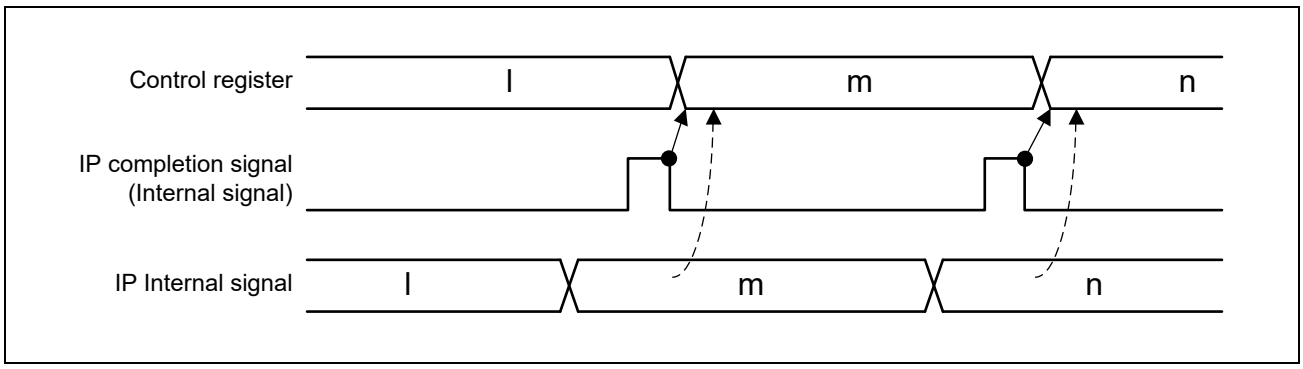


Figure 25.32 Outline of the Buffering of a Read-only Register

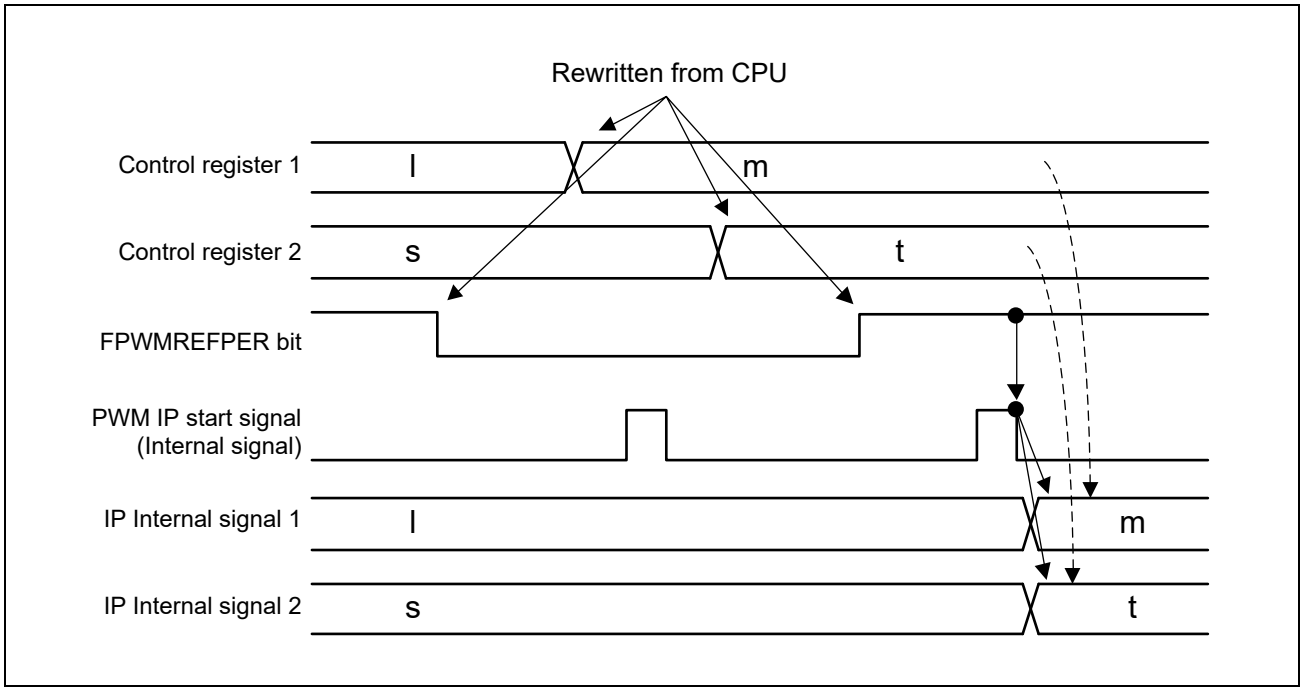


Figure 25.33 Outline of Simultaneous Reflection Control by the FPWMREFPER Bit

Table 25.187 List of Register Buffers (Angle Generation IP)

Register Name	Symbol Name	Bit Name	Timing	Remarks
EMU3n resolver angle offset value register	EMU3nANGOFS	All bits	Angle changes being detected	—

Table 25.188 List of Register Buffers (Input IP)

Register Name	Symbol Name	Bit Name	Timing	Remarks
EMU3n input IP control register	EMU3nCTRINMD	FREGIN	Input IP startup	—
EMU3n electrical angle register	EMU3nTHTEFIX	All bits	(Angle value latch trigger) Input IP startup	—
EMU3n input IP electrical angle software input register	EMU3nTHTESFT	All bits	Input IP startup	—
EMU3n electrical angle response delay correction variable register	EMU3nEARD	All bits	Input IP startup	—
EMU3n post error convolution resolver angle register	EMU3nTHTRESFIX	All bits	(Angle value latch trigger)	—
EMU3n resolver angle monitor register	EMU3nTHTRESFIXIN	All bits	Input IP completion	—
EMU3n A/D data 0 register	EMU3nAD0	All bits	Input IP startup	—
EMU3n A/D data 1 register	EMU3nAD1	All bits	Input IP startup	—
EMU3n A/D data 2 register	EMU3nAD2	All bits	Input IP startup	—
EMU3n IIR filter channel 0 output data register	EMU3nIIROUTDAT0	All bits	Input IP startup	—
EMU3n IIR filter channel 1 output data register	EMU3nIIROUTDAT1	All bits	Input IP startup	—
EMU3n IIR filter channel 2 output data register	EMU3nIIROUTDAT2	All bits	Input IP startup	—
EMU3n A/D data 0 origin correction value register	EMU3nAD0OFS	All bits	Input IP startup	—
EMU3n A/D data 1 origin correction value register	EMU3nAD1OFS	All bits	Input IP startup	—
EMU3n A/D data 2 origin correction value register	EMU3nAD2OFS	All bits	Input IP startup	—
EMU3n LSB adjustment register	EMU3nDIVLSB	All bits	Input IP startup	—
EMU3n A/D data 0 converted value output buffer register	EMU3nAD0FIXOBUF	All bits	Input IP completion	—
EMU3n A/D data 1 conversion value output buffer register	EMU3nAD1FIXOBUF	All bits	Input IP completion	—
EMU3n A/D data 2 conversion value output buffer register	EMU3nAD2FIXOBUF	All bits	Input IP completion	—
EMU3n U phase current value output buffer register	EMU3nIUFIXOBUF	All bits	Input IP completion	—
EMU3n V phase current value output buffer register	EMU3nIVFIXOBUF	All bits	Input IP completion	—
EMU3n W phase current value output buffer register	EMU3nIWFIXOBUF	All bits	Input IP completion	—
EMU3n d-axis current value output buffer register	EMU3nIDFIXOBUF	All bits	Input IP completion	—
EMU3n q-axis current value output buffer register	EMU3nIQFIXOBUF	All bits	Input IP completion	—

Table 25.189 List of Register Buffers (PI Control IP)

Register Name	Symbol Name	Bit Name	Timing	Remarks
EMU3n PI Control IP Control Register	EMU3nPICTR	FSUMID, FSUMIQ	PI control IP startup	—
EMU3n d-axis directive current value register	EMU3nIDIN	All bits	PI control IP startup	—
EMU3n q-axis directive current value register	EMU3nIQIN	All bits	PI control IP startup	—
EMU3n d-axis proportional gain 0 register	EMU3nGPD0	All bits	PI control IP startup	—
EMU3n q-axis proportional gain 0 register	EMU3nGPQ0	All bits	PI control IP startup	—
EMU3n d-axis integral gain register	EMU3nGID	All bits	PI control IP startup	—
EMU3n q-axis integral gain register	EMU3nGIQ	All bits	PI control IP startup	—
EMU3n d-axis integrated maximum value register	EMU3nGIDMAX	All bits	PI control IP startup	—
EMU3n q-axis integrated maximum value register	EMU3nGIQMAX	All bits	PI control IP startup	—
EMU3n d-axis proportional gain register	EMU3nGPD	All bits	PI control IP startup	—
EMU3n q-axis proportional gain register	EMU3nGPQ	All bits	PI control IP startup	—
EMU3n d-axis voltage maximum value register	EMU3nVDMAX	All bits	PI control IP startup	—
EMU3n q-axis voltage maximum register	EMU3nVQMAX	All bits	PI control IP startup	—
EMU3n d-axis current value software input register	EMU3nID	All bits	PI control IP startup	—
EMU3n q-axis current value software input register	EMU3nIQ	All bits	PI control IP startup	—
EMU3n d-axis integrated value software input register	EMU3nSUMID	All bits	PI control IP startup	—
EMU3n q-axis integrated value software input register	EMU3nSUMIQ	All bits	PI control IP startup	—
EMU3n d-axis voltage value output buffer register	EMU3nVDOBUF	All bits	PI control IP completion	—
EMU3n q-axis voltage value output buffer register	EMU3nVQOBUF	All bits	PI control IP completion	—

Table 25.190 List of Register Buffers (PWM IP)

Register Name	Symbol Name	Bit Name	Timing	Simultaneous Reflection Control
EMU3n d-axis voltage correction value register	EMU3nVDCRCT	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n q-axis voltage correction value register	EMU3nVQCRCT	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n PWM IP electrical angle offset register	EMU3nPHI	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n carrier period register	EMU3nCARR	All bits	PWM IP startup	—
EMU3n PWM IP control register	EMU3nPWMCTR	FLININIP	PWM IP startup	—
EMU3n PWM IP electrical angel software input register	EMU3nTHTFORESFT	All bits	PWM IP startup	—
EMU3n PWM IP electrical angle adjustment coefficient register	EMU3nGTHT	All bits	PWM IP startup	—
EMU3n U phase voltage correction value register	EMU3nVUOFS	All bits	PWM IP startup	—
EMU3n V phase voltage correction value register	EMU3nVVOFS	All bits	PWM IP startup	—
EMU3n W phase voltage correction value register	EMU3nVWOFS	All bits	PWM IP startup	—
EMU3n Input voltage register	EMU3nVOLV	All bits	PWM IP startup	—
EMU3n non-interference control coefficient angular velocity value gain register	EMU3nDECVELG	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n non-interference control coefficient Ld value register	EMU3nDECLD	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n non-interference control coefficient Lq value register	EMU3nDECLQ	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n non-interference control coefficient magnetic flux value register	EMU3nDECFLUX	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n PWM modulation peak value register	EMU3nTMAX	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n dq-axis voltage phase angle software input register	EMU3nTHTVSFT	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER
EMU3n dq-axis voltage value software input register	EMU3nVDQSFT	All bits	PWM IP startup	EMU3nREFCTR.FPWMREFPER

Table 25.191 List of Register Buffers (Rectangle IP)

Register Name	Symbol Name	Bit Name	Timing	Remarks
EMU3n rectangle IP control register	EMU3nRECCTR	FDRCT, FIPPOSI	Rectangle IP startup	—
EMU3n q-axis reference voltage phase software input register	EMU3nPHQSFT	All bits	Rectangle IP startup	—
EMU3n switching instruction software input register	EMU3nPSWSFT	All bits	Rectangle IP startup	—

Table 25.192 List of Register Buffers (IIR Filter)

Register Name	Symbol Name	Bit Name	Timing	Remarks
EMU3n IIR filter coefficient 0 value register	EMU3nIIRCOEFF0	All bits	EMU3nIIRRLD set	—
EMU3n IIR filter coefficient 1value register	EMU3nIIRCOEFF1	All bits	EMU3nIIRRLD set	—
EMU3n IIR filter coefficient 2 value register	EMU3nIIRCOEFF2	All bits	EMU3nIIRRLD set	—
EMU3n IIR filter coefficient 3 value register	EMU3nIIRCOEFF3	All bits	EMU3nIIRRLD set	—
EMU3n IIR filter coefficient 4 value register	EMU3nIIRCOEFF4	All bits	EMU3nIIRRLD set	—
EMU3n IIR filter coefficient 5 value register	EMU3nIIRCOEFF5	All bits	EMU3nIIRRLD set	—
EMU3n IIR filter shift amount value register	EMU3nIIRSHIFT	All bits	EMU3nIIRRLD set	—

Table 25.193 List of Register Buffers (Checking Buffer)

Register Name	Symbol Name	Bit Name	Timing	Remarks
EMU3n A/D data 0 checking buffer register	EMU3nCBAD0	All bits	Register set	—
EMU3n A/D data 1 checking buffer register	EMU3nCBAD1	All bits	Register set	—
EMU3n A/D data 2 checking buffer register	EMU3nCBAD2	All bits	Register set	—
EMU3n resolver angle checking buffer register	EMU3nCBTHTRESFIXIN	All bits	Register set	—
EMU3n d-axis current value checking buffer register	EMU3nCBIDFIX	All bits	Register set	—
EMU3n q-axis current value checking buffer register	EMU3nCBIQFIX	All bits	Register set	—
EMU3n U phase PWM compare value checking buffer register	EMU3nCBPWMUIP	All bits	Register set	—
EMU3n V phase PWM compare value checking buffer register	EMU3nCBPWMVIP	All bits	Register set	—
EMU3n W phase PWM compare value checking buffer register	EMU3nCBPWMWIP	All bits	Register set	—
EMU3n rectangle pattern value checking buffer register	EMU3nCBBREC	All bits	Register set	—
EMU3n independent rectangle IP1 pattern value checking buffer register	EMU3nCBIREC	All bits	Register set	—

25.4.13 Asynchronous Data Passing Function

The EMU3 is provided with the function to transfer two or more items of data simultaneously between tasks running at different cycle timings (register handshaking function).

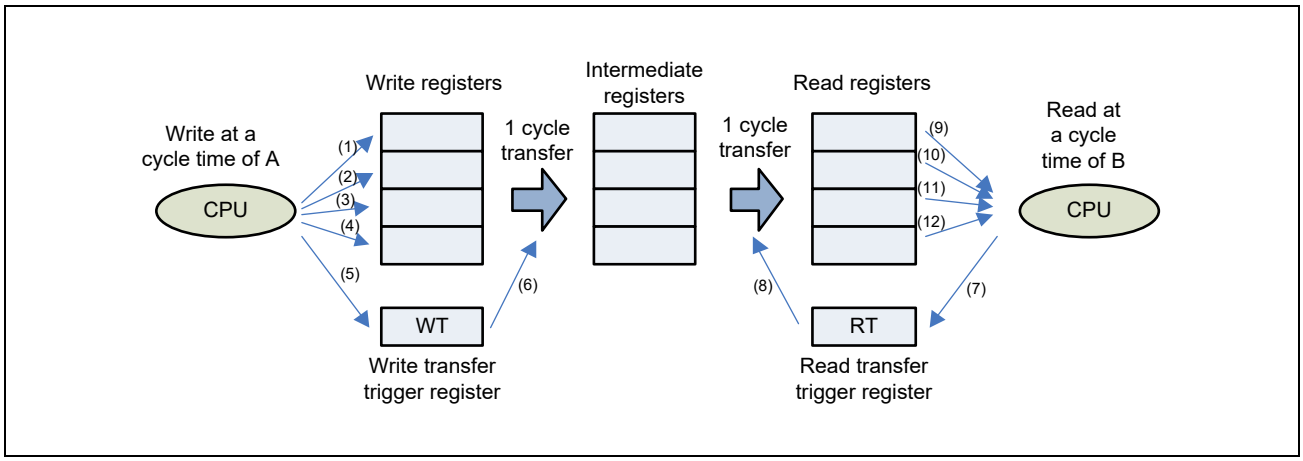


Figure 25.34 Operations of Asynchronous Handshaking Registers

Figure 25.34 shows an example of transferring data from the CPU with a cycle time of A to the CPU with a cycle time of B.

The CPU which has finished regular processing at a cycle time of A writes data sets sequentially into the data set write register (EMU3nDATSETWkm) which are used for handshaking. After the writing of all data sets is completed, the CPU transfer them to the intermediate registers in 1 cycle by manipulating the bits of the data set WBk transfer trigger register (EMU3nDATSETWBk). Subsequently, the CPU can write data into the data set write register (EMU3nDATSETWkm) again at an arbitrary timing.

The data stored in the intermediate registers are taken out at a cycle time of B. The data sets which are temporarily stored in the intermediate registers are transferred to the read registers by manipulating the bits of the data set BRk transfer trigger register (EMU3nDATSETBRk). The data that is read into the read registers can be used at any timing within the cycles B.

25.4.14 Fault Detection Function

The H/W accelerator is provided with the function to buffer the resultant values of motor control processing, which is performed to detect circuit faults, in dedicated registers. It is possible to buffer the following values at the specified timings such as “input IP completion” and “detection of an angle compare 0 match”.

The buffered values can be converted and their correctness be verified through the diagnostic program of the CPU.

Table 25.194 Checking Buffer Register

Register and Value (Source)	Checking Buffer Register (Destination)
EMU3n A/D data k register (EMU3nADk) (k = 0, 1, 2)	EMU3n A/D data k checking buffer register (EMU3nCBADk) (k = 0, 1, 2)
EMU3n resolver angle monitor register (EMU3nTHTRESFIXIN)	EMU3n resolver angle checking buffer register (EMU3nCBTHTRESFIXIN)
EMU3n d-axis current value register (EMU3nIDFIX)	EMU3n d-axis current value checking buffer register (EMU3nCBIDFIX)
EMU3n q-axis current value register (EMU3nIQFIX)	EMU3n q-axis current value checking buffer register (EMU3nCBIQFIX)
U phase PWM compare value pin output value	EMU3n U phase PWM compare value checking buffer register (EMU3nCBPWMUIP)
V phase PWM compare value pin output value	EMU3n V phase PWM compare value checking buffer register (EMU3nCBPWMVIP)
W phase PWM compare value pin output value	EMU3n W phase PWM compare value checking buffer register (EMU3nCBPWMWIP)
U/V/W phase rectangle output pattern value	EMU3n rectangle pattern value checking buffer register (EMU3nCBBREC)
U/V/W phase independent rectangle 1 output pattern value	EMU3n Independent rectangle 1 pattern value checking buffer register (EMU3nCBIREC)

The use of the buffering function makes it possible to transfer the target register and its value to the corresponding checking buffer register at the timing selected through the EMU3nCBTIM register.

Register buffering can be enabled in two ways. When the CBEN1 bit of the EMU3nCBCTR0 register to 0, buffering is permitted only once every time the CBEN0 bit is set to 1. The CBMON bit of the EMU3nCBCTR0 register is set to 1 when the CBEN0 bit is set to 1 and reset to 0 when the buffering is completed.

When the CBEN1 bit is set to 1, buffering is always enabled. The CBMON bit is held at 1 while the CBEN1 bit is set to 1.

Setting the CBUFINT bit of the EMU3nINTk registers (k = 0 to 7) to 1 (enable interrupts) causes a checking buffering completion interrupt to be generated upon completion of buffering.

25.4.15 Interrupt Control

The EMU3 has 16 interrupt output pins as listed in **Table 25.195***[1].

EMU3n interrupts 0 to 7 can be assigned two or more selected interrupt sources from the H/W accelerator (channel n). If multiple interrupt source conditions of a single interrupt are satisfied simultaneously, the result may be the output of only a single instance of the interrupt signal.

The interrupt source select register (EMU3nINTk) can be used to select the interrupt sources. The priorities of the interrupts to the CPU should be set in the interrupt controller.

Table 25.195 Interrupt Signal Outputs

Name	Function	Register that can Select the Interrupt Source
EMU30 interrupt 0	Channel 0 H/W accelerator interrupt source 0	EMU30 interrupt source select register 0
EMU30 interrupt 1	Channel 0 H/W accelerator interrupt source 1	EMU30 interrupt source select register 1
EMU30 interrupt 2	Channel 0 H/W accelerator interrupt source 2	EMU30 interrupt source select register 2
EMU30 interrupt 3	Channel 0 H/W accelerator interrupt source 3	EMU30 interrupt source select register 3
EMU30 interrupt 4	Channel 0 H/W accelerator interrupt source 4	EMU30 interrupt source select register 4
EMU30 interrupt 5	Channel 0 H/W accelerator interrupt source 5	EMU30 interrupt source select register 5
EMU30 interrupt 6	Channel 0 H/W accelerator interrupt source 6	EMU30 interrupt source select register 6
EMU30 interrupt 7	Channel 0 H/W accelerator interrupt source 7	EMU30 interrupt source select register 7
EMU31 interrupt 0	Channel 1 H/W accelerator interrupt source 0	EMU31 interrupt source select register 0
EMU31 interrupt 1	Channel 1 H/W accelerator interrupt source 1	EMU31 interrupt source select register 1
EMU31 interrupt 2	Channel 1 H/W accelerator interrupt source 2	EMU31 interrupt source select register 2
EMU31 interrupt 3	Channel 1 H/W accelerator interrupt source 3	EMU31 interrupt source select register 3
EMU31 interrupt 4	Channel 1 H/W accelerator interrupt source 4	EMU31 interrupt source select register 4
EMU31 interrupt 5	Channel 1 H/W accelerator interrupt source 5	EMU31 interrupt source select register 5
EMU31 interrupt 6	Channel 1 H/W accelerator interrupt source 6	EMU31 interrupt source select register 6
EMU31 interrupt 7	Channel 1 H/W accelerator interrupt source 7	EMU31 interrupt source select register 7

Note 1. EMU31 interrupts 0 to 7 are only supported for the RH850/C1M-A2.

Types of interrupt sources that can be assigned to the interrupt outputs from the H/W accelerator

- Angle compare 0 match detection, angle compare 1 match
- Independent rectangle IP1 U phase angle compare detection, V phase angle compare detection, W phase angle compare detection
- Independent rectangle IP2 U phase angle compare detection, V phase angle compare detection, W phase angle compare detection
- Carrier trigger peak and trough
- Input IP completion, PI control IP completion, PWM IP completion, rectangle IP completion
- pulse cycle measurement timer overflow detection
- Resolver angle measurement timer capture/overflow detection
- A/D conversion completion
- IIR channel 0, 1, 2 processing completion
- Checking register update completion

- Kirchhoff's current law violation
- Input, PI control, and PWM IP WAIT transition detection
- Angle generation IP WAIT transition detection

25.5 Operations

25.5.1 EMU3 Initialization

A sample flow for initializing the H/W accelerator is given below. There is no need to configure any functions that are not to be used.

This setting procedure should be observed when restarting the EMU3 (operation→reset→operation) using the EMUST bit.

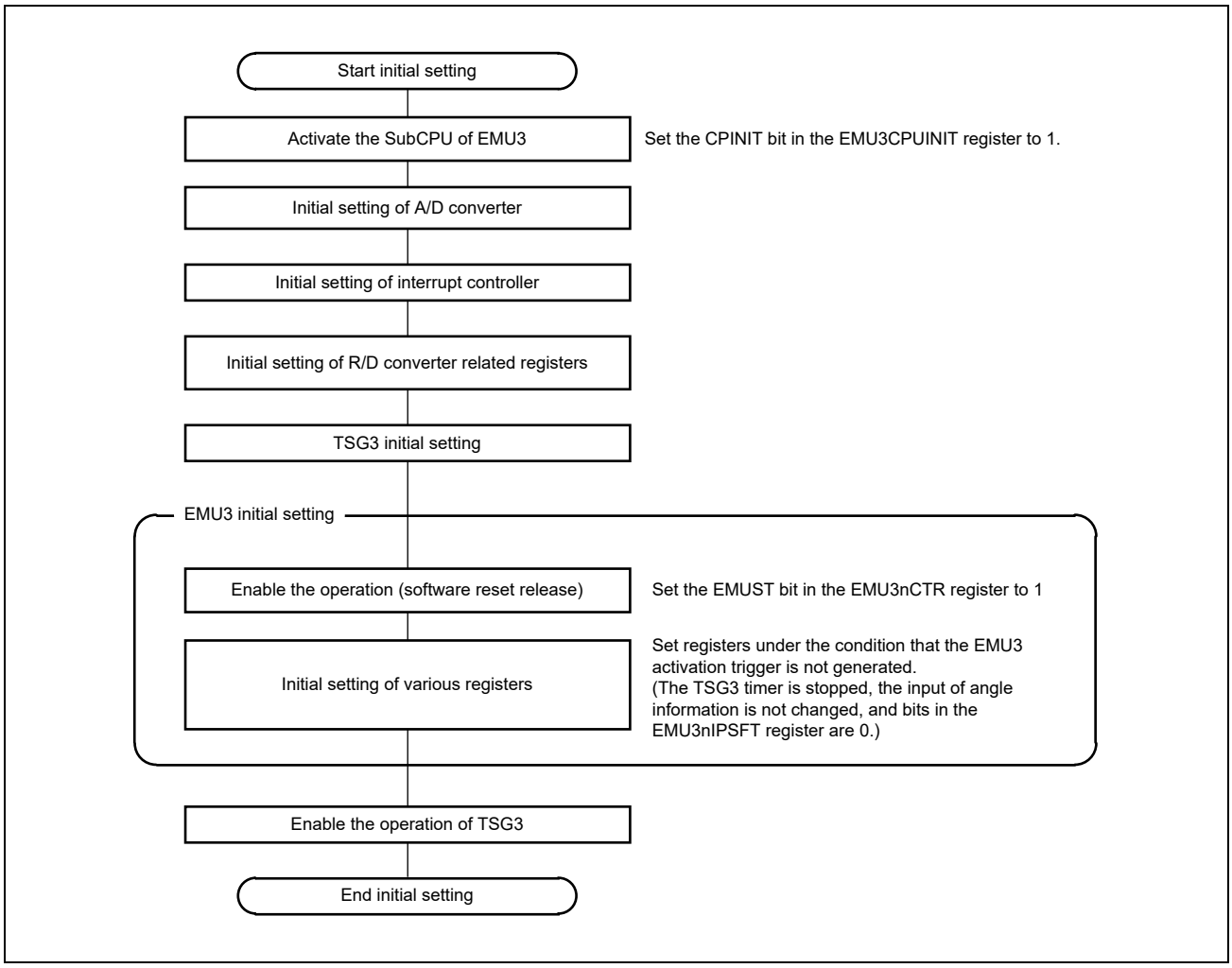


Figure 25.35 Sample Flow for Initializing the H/W Accelerator

25.5.2 Procedures for Initializing and Processing Interrupts of the IPs of the H/W Accelerator

Given below are the examples of procedures for initializing the functions of the H/W accelerator and for processing related interrupts. By combining the CPU interrupts and software executions, it is possible in below methods.

- Correct the processing results of the PI control IP with the CPU program and to implement various execution procedures such as let the PWM IP succeed in processing the rest.
- Exercise PI control from within the CPU program based on the processing results of the input IP and let the PWM IP succeed in processing the rest.

When exercising rectangle wave control under software processing of the batch rectangle IP and rectangle wave control using the independent rectangle IP1 and independent rectangle IP2, it is also possible to use the procedure such as “update the comparison and pattern values under CPU program control on detection of an angle compare 0 match”.

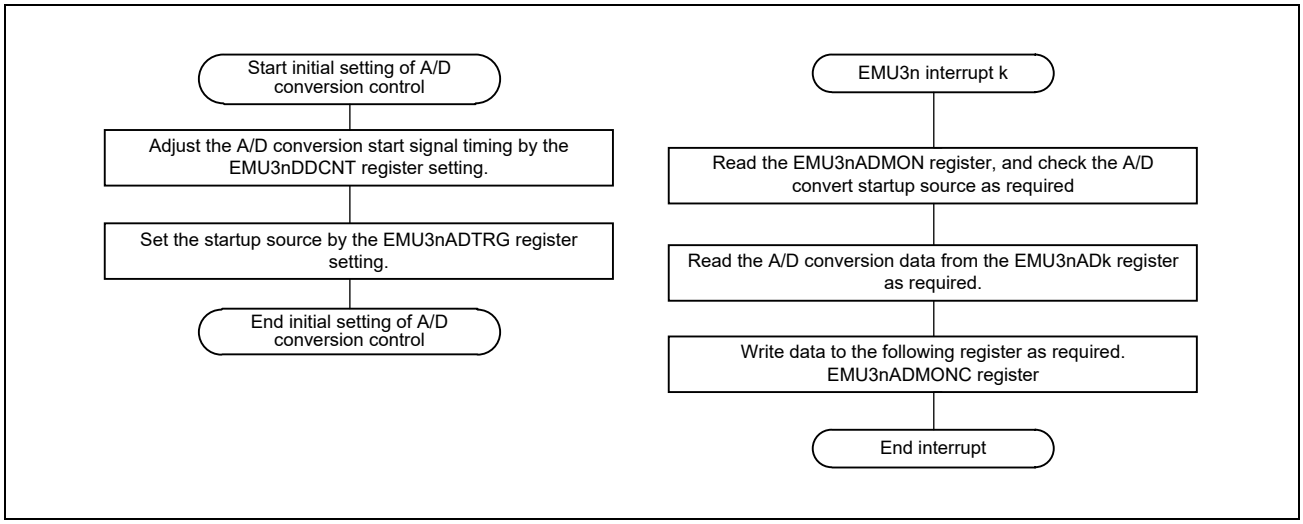


Figure 25.36 Sample Setting Procedure for Making Initialization for A/D Conversion Control (Using Interrupts)

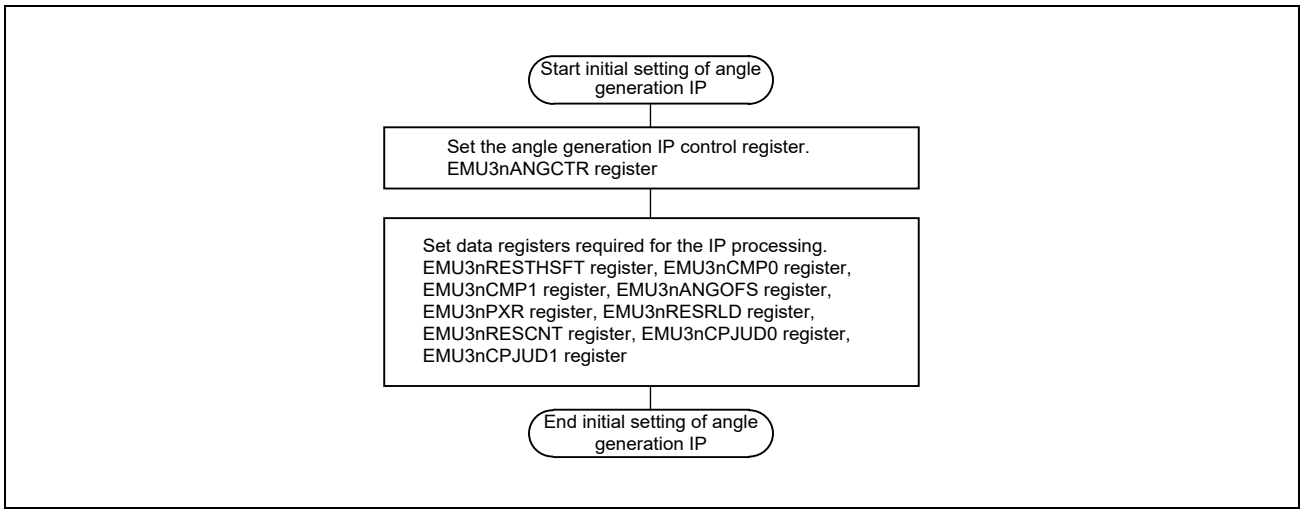


Figure 25.37 Sample Setting Procedure for Initializing the Angle Generation IP

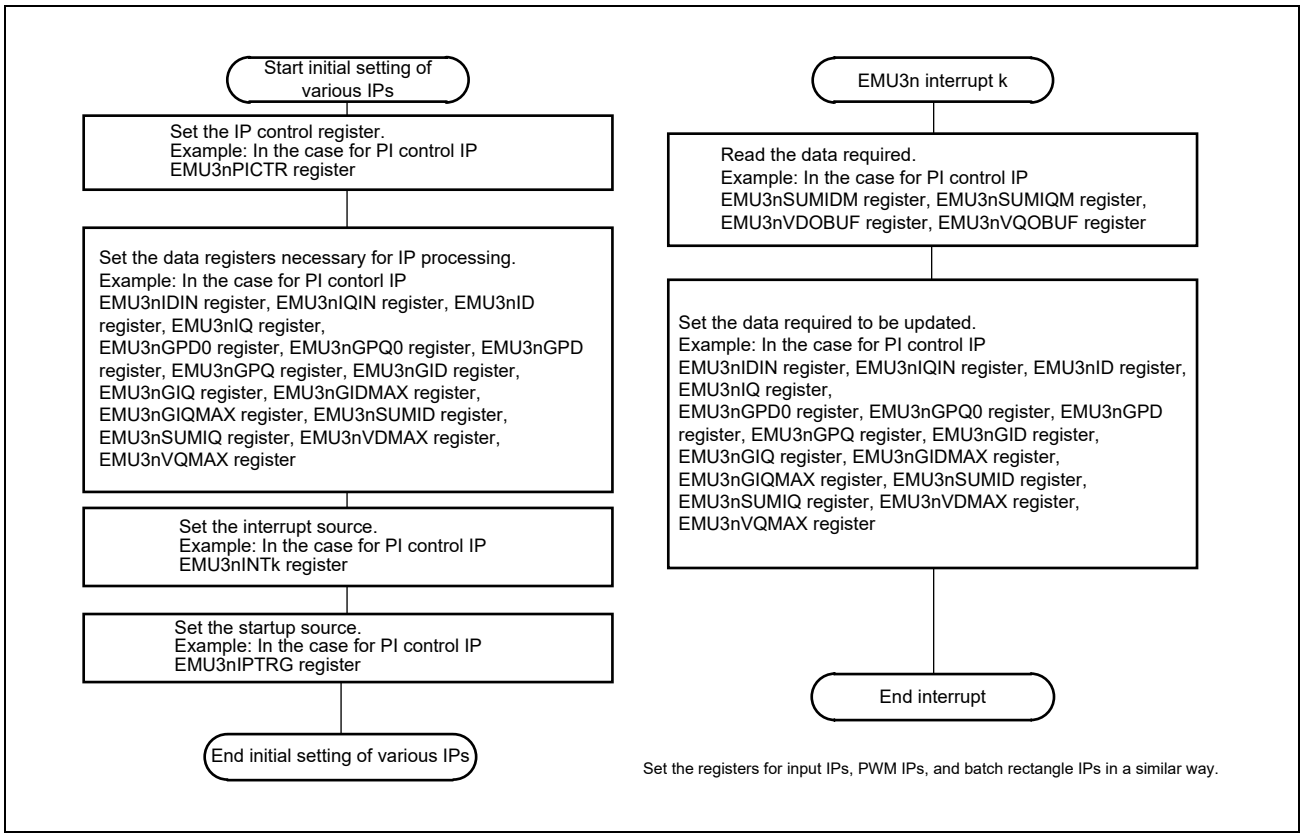


Figure 25.38 Sample Setting Procedure for Initializing the Input IP, PI Control IP, PWM IP, and Batch Rectangle IP (Hardware Processing) (Using Interrupts)

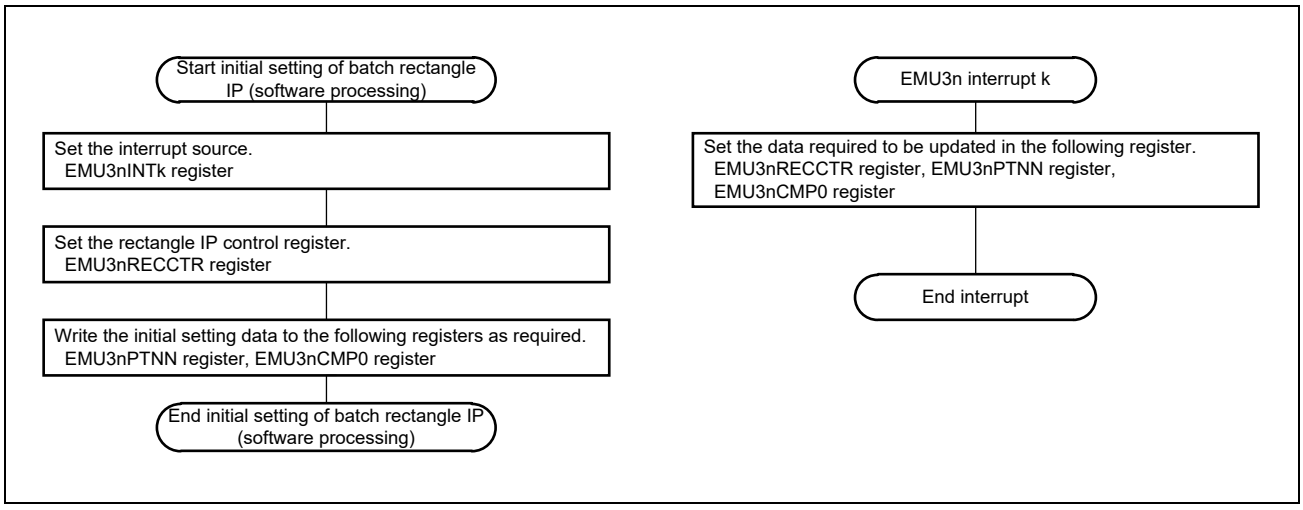


Figure 25.39 Sample Setting Procedure for Initializing the Batch Rectangle IP (Software Processing) (Using Interrupts)

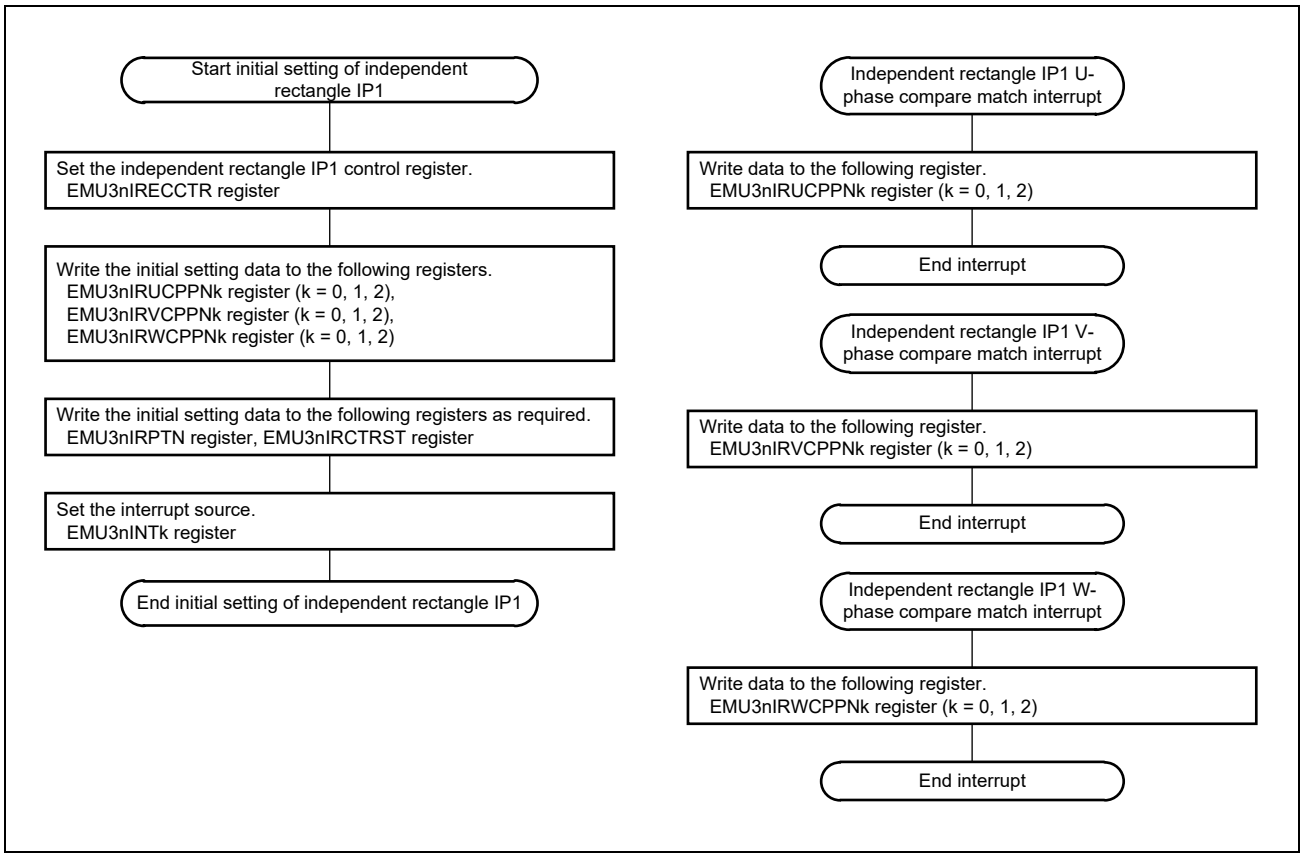


Figure 25.40 Sample Setting Procedure for Initializing the Independent Rectangle IP1 (Using Interrupts)

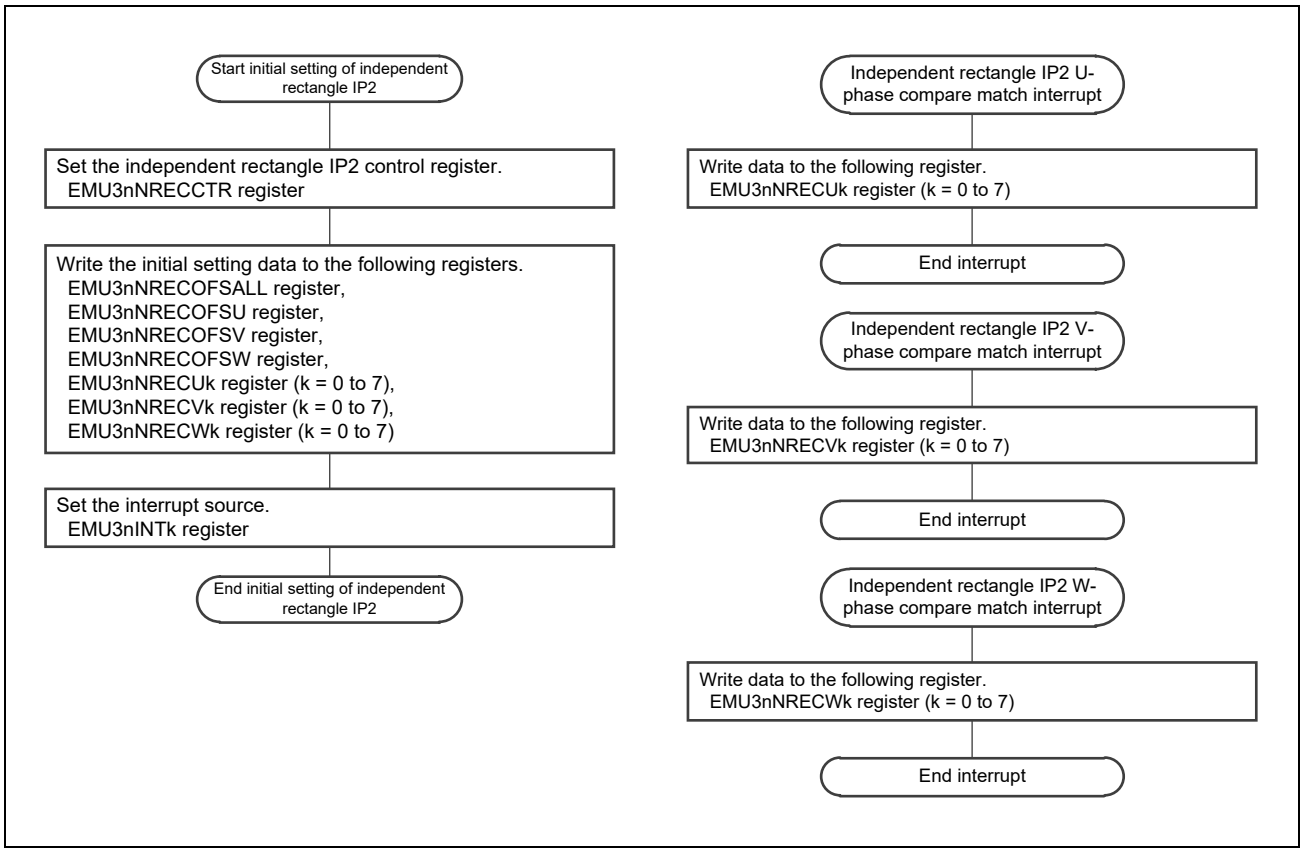


Figure 25.41 Sample Setting Procedure for Initializing the Independent Rectangle IP2 (Using Interrupts)

25.5.3 Collaborative Operation of the CPU and H/W Accelerator (on a H/W Arithmetic Block Basis)

The input IP, PI control IP, PWM IP, and angle generation IP of the H/W accelerator can realize flexible arithmetic processing by inserting CPU-based software processing in between the fine-grained H/W arithmetic blocks Func(*).

The collaboration from a H/W arithmetic block to the CPU is accomplished by the notification of an interrupt to CPU.

The collaboration from the CPU to a H/W arithmetic block is accomplished by writing data into the EMU3nFUNCWAITGRP* register (*: A or B).

Figure 25.42 shows the outline of CPU to H/W accelerator collaboration using the PWM IP as an example.

The state of each IP is represented by the following four terms:

- RUN:

  Indicates that the IP is running.

  This corresponds to the IP bits of the EMU3nFSMSTGRP* register (*: A or B). For example, “11” in the IP bits of the EMU3nFSMSTGRPA mean that the PWM IP is running.

- IDLE:

  Indicates that the IP is stopped.

  This corresponds to the IP bits of the EMU3nFSMSTGRP* register (*: A or B). For example, a value other than “11” in the IP bits of the EMU3nFSMSTGRPA register means that the PWM IP is stopped.

- WAIT:

  Indicates that no H/W arithmetic block Func(*) is being executed while the IP is running.

  This corresponds to the WAIT bit of the EMU3nFSMSTGRP* register (*: A or B).

- BUSY:

  Indicates that a H/W arithmetic block Func(*) is being executed while the IP is running.

  This corresponds to the BUSY bit of the EMU3nFSMSTGRP* register (*: A or B).

(a) in **Figure 25.42** shows an example of running the PWM IP execution in the normal state. After the PWM IP is started, it completes its processing after running all H/W arithmetic blocks.

(b) in **Figure 25.42** shows an example of inserting CPU software processing between PWM IP's H/W arithmetic blocks.

Interrupt are notified to the CPU by switching the H/W arithmetic block into the WAIT state, so that the CPU can execute the interrupt processing. The H/W arithmetic block is restarted by writing the EMU3nFUNCWAITGRP* register (*: A or B).

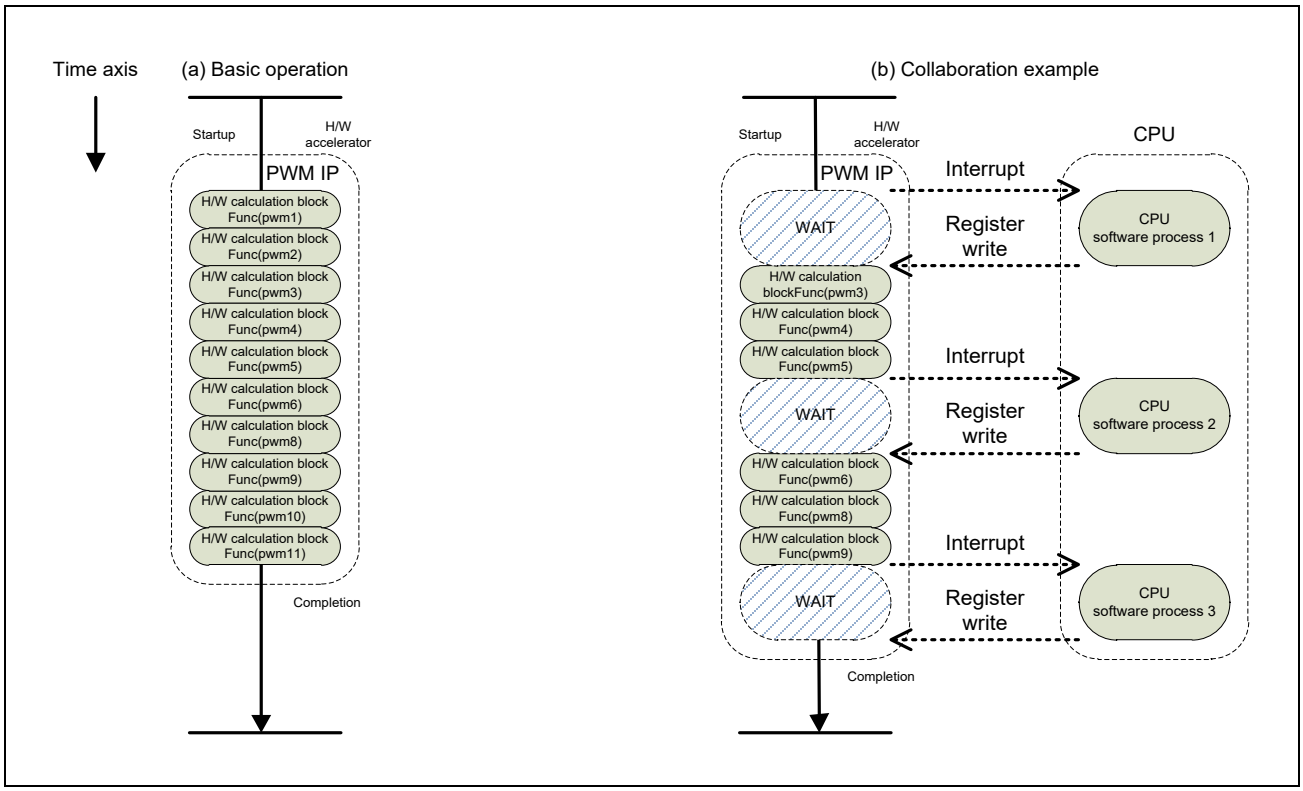


Figure 25.42 Outline of H/W Accelerators' Collaboration

25.5.3.1 Related Registers

Table 25.196 shows a list of registers that are necessary to run the H/W arithmetic block basis.

The registers that are associated with the input IP, PI control IP, PWM IP, and angle generation IP are listed in **Table 25.197**.

Table 25.196 List of Registers Associated with the H/W Accelerator Collaborative Operations (on a H/W Arithmetic Block Basis)

Register	Function
EMU3nFUNCIDLEGRP* (*:A0,A1,A2,B)	Specifies the H/W arithmetic block that is transited if IP is started during “IP stopped state (IDLE)”. If the SFTEN bit of the EMU3nFUNCIDLEGRP* (*A0,A1,A2) register is set to 0, the specified H/W arithmetic block is transited from IDLE at the timing that IP startup source occurs. If the SFTEN bit of the EMU3nFUNCIDLEGRP* (*A0,A1,A2) register is set to 1, the specified H/W arithmetic block is transited from IDLE by write access to the required register. that is, the transition is software startup.
EMU3nFUNCFLGRP* (*:A0,A1,A2,B)	Specifies the type of transition upon completion of the execution of a H/W arithmetic block selected from “Next block”, “IDLE”, and “WAIT”.
EMU3nFUNCWAITGRP* (*:A,B)	Specifies the next H/W arithmetic block to transition from WAIT state. By write access to this register, the specified H/W arithmetic block is startup from WAIT state.
EMU3nFUNCFINGRP* (*:A,B)	Indicates the latest H/W arithmetic block that is transited to IDLE or WAIT at completion of H/W arithmetic block execution if setting each symbol of the EMU3nFUNCFLGRP* register (*:A0,A1,A2,B) is “01” or “10”.
EMU3nFSMSTGRP* (*:A,B)	It can monitor that the state of each IP becomes any of RUN, IDLE, WAIT or BUSY.
EMU3nINTk (k = 0 to 7) EMU3nINTSD EMU3nINTSDC	There are interrupt source select bits of “WAIT transition detection of each IP”. By outputting the interrupt into CPU at the timing of WAIT transition after H/W arithmetic block completion, it is noticed the timing of WAIT transition into CPU. By checking the value of of EMU3nFUNCFINGRP* register, it can identify the H/W arithmetic block that was executed before the WAIT transition.

Table 25.197 List of Registers Associated with the IPs

Register	input IP	PI control IP	PWM IP	Angle generation IP
EMU3nFUNCIDLEGRP* (*:A0,A1,A2,B)	*=A0	*=A1	*=A2	*=B
EMU3nFUNCFLGRP* (*:A0,A1,A2,B)	*=A0	*=A1	*=A2	*=B
EMU3nFUNCWAITGRP* (*:A,B)	*=A	*=A	*=A	*=B
EMU3nFUNCFINGRP* (*:A,B)	*=A	*=A	*=A	*=B
EMU3nFSMSTGRP* (*:A,B)	*=A	*=A	*=A	*=B
EMU3nINTk.FWG* (k = 0 to 7) (*:A,B) EMU3nINTSD.FWG*IF (*:A,B) EMU3nINTSDC.FWG*IFC (*:A,B)	*=A	*=A	*=A	*=B

25.5.3.2 Fine-Grained IP State Transitions

Figure 25.43 shows the relationship between the fine-grained IP state transitions and the above-mentioned registers using the H/W arithmetic block Func(pwm3) of the PWM IP. The similar mechanism also applies to the input IP, PI control IP, and angle generation IP.

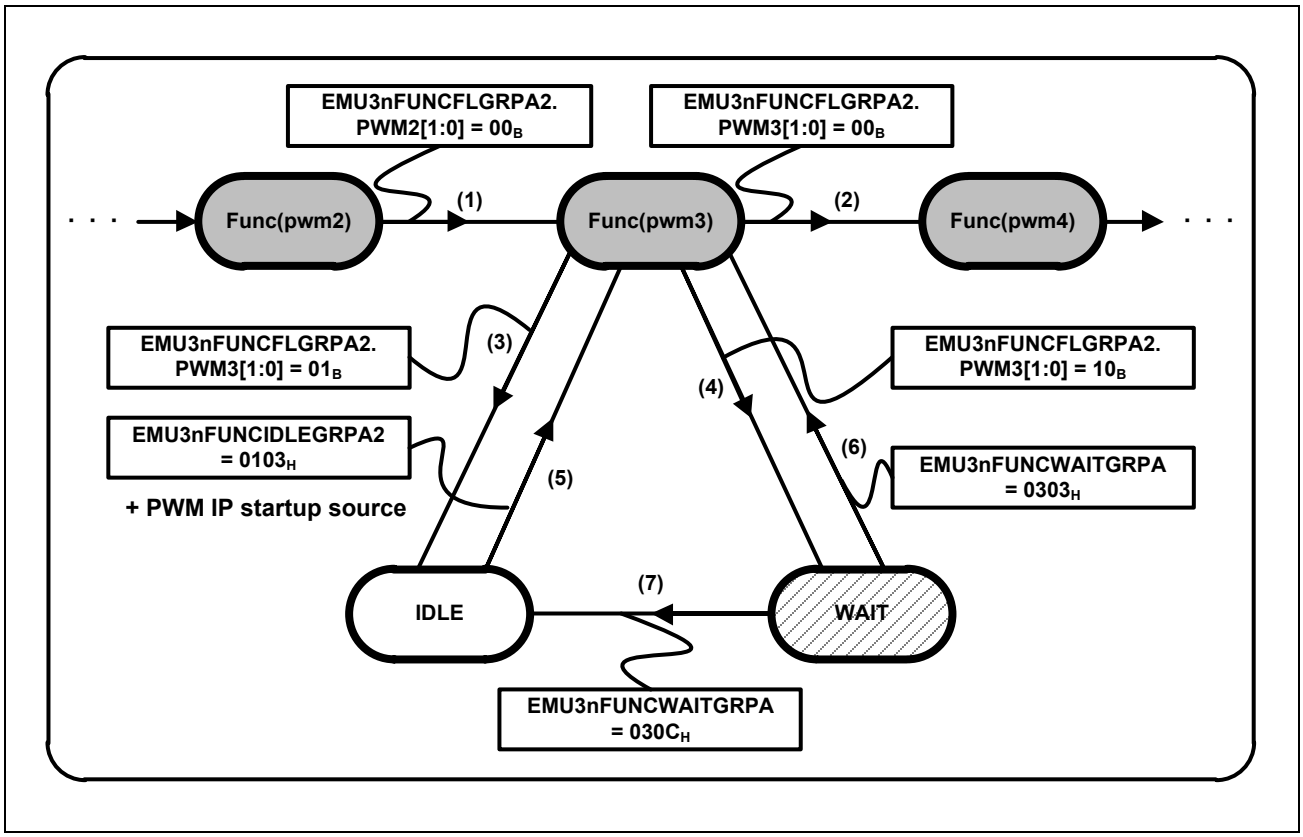


Figure 25.43 PWM IP State Transitions

(1) Func(pwm2)→Func(pwm3)

Setting the PWM2 bits of the EMU3nFUNCFLGRPA2 register to “00” causes a transition to Func(pwm3) which is the next H/W arithmetic block.

(2) Func(pwm3)→Func(pwm4)

Setting the PWM3 bits of the EMU3nFUNCFLGRPA2 register to “00” causes a transition to Func(pwm4) which is the next H/W arithmetic block.

(3) Func(pwm3)→IDLE

Setting the PWM3 bits of the EMU3nFUNCFLGRPA2 register to “01” causes a transition to IDLE.

(4) Func(pwm3)→WAIT

Setting the PWM3 bits of the EMU3nFUNCFLGRPA2 register to “10” causes a transition to WAIT.

(5) IDLE→Func(pwm3)

Setting the EMU3nFUNCIDLEGRPA2 register with a value of "0103_H" causes a transition not to ordinary Func(pwm1) but to Func(pwm3) to occur when the PWM IP is started. If a source of starting up the PWM IP occurs with this register setting, a transition to Func(pwm3) occurs when the PWM IP is started.

In addition to this, it is possible to start the PWM IP at the timing when the EMU3nFUNCIDLEGRPA2 register is written while the PWM IP is stopped (IDLE). Writing a value of "1103_H" into the EMU3nFUNCIDLEGRPA2 register when the PWM IP is stopped ((IDLE) causes the PWM IP to start and a transition to Func(pwm3) at this write timing.

(6) WAIT→Func(pwm3)

Setting the EMU3nFUNCWAITGRPA register with a value of "0303_H" causes a transition from the WAIT state to Func(pwm3).

(7) WAIT→IDLE

Setting the EMU3nFUNCWAITGRPA register with a value of "$030C_H$" causes a transition from the WAIT state to PWM IP complete (IDLE).

25.5.3.3 Setting Examples

Figure 25.44 is a use case of collaborative processing with the CPU based on the collaborative operations of the H/W accelerator.

In the use case shown in **Figure 25.44**, the register settings and states for inserting CPU software processing in between the H/W arithmetic blocks of the PWM IP are explained in chronological order in paragraphs (1) to (9) that follow.

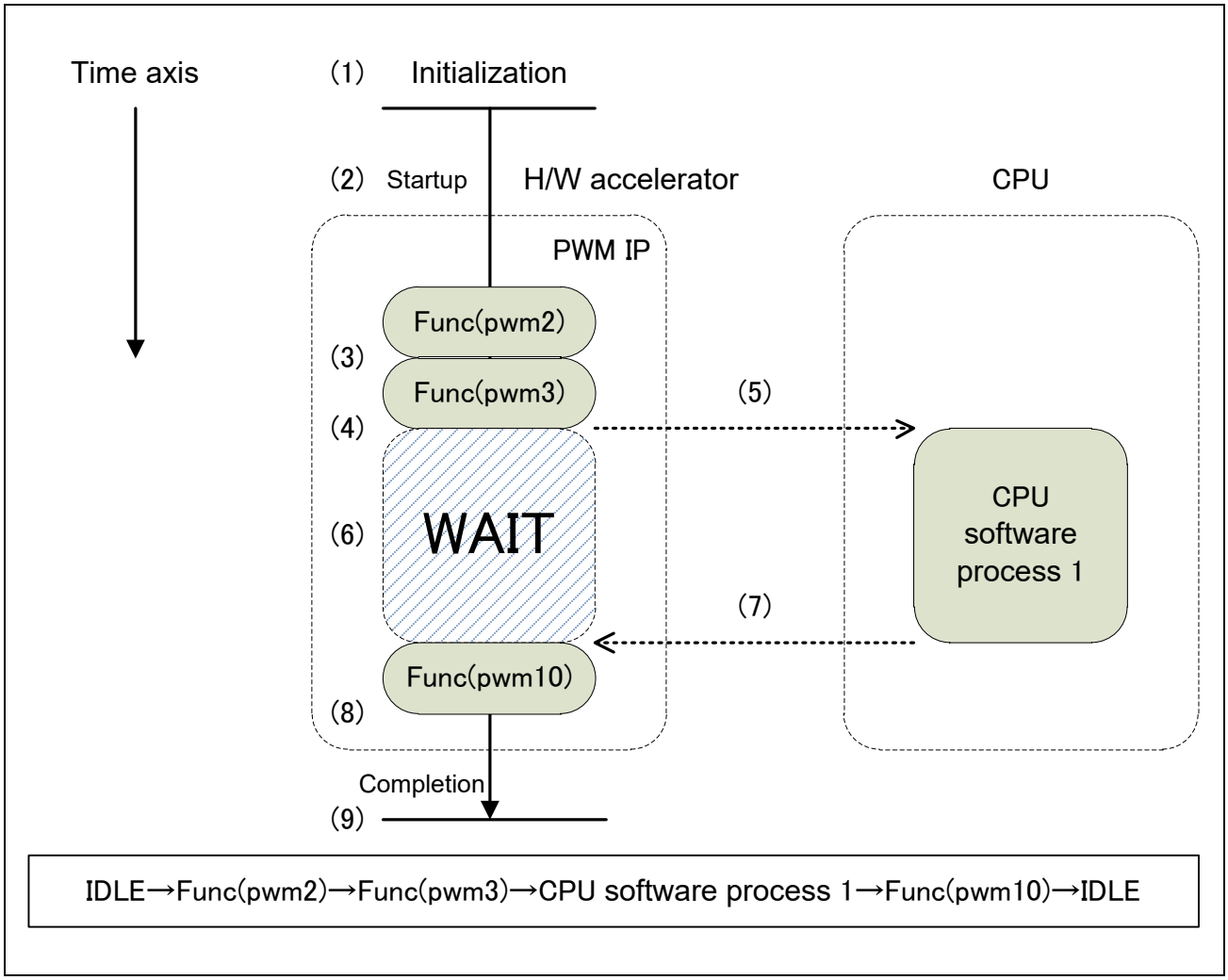


Figure 25.44 Use Case of Setting for H/W Accelerator's Collaborative Operations

(1) Initialization

The initialization is performed for implementing the use case shown in **Figure 25.46**.

In this example, the EMU3nIPTRG register and EMU3nFUNCIDLEGRPA2 register are set up so that the PWM IP can be started by a software trigger.

The EMU3nINTk registers (k = 0 to 7) are set up so that an interrupt to CPU can occur when the PWM IP transits to the WAIT state.

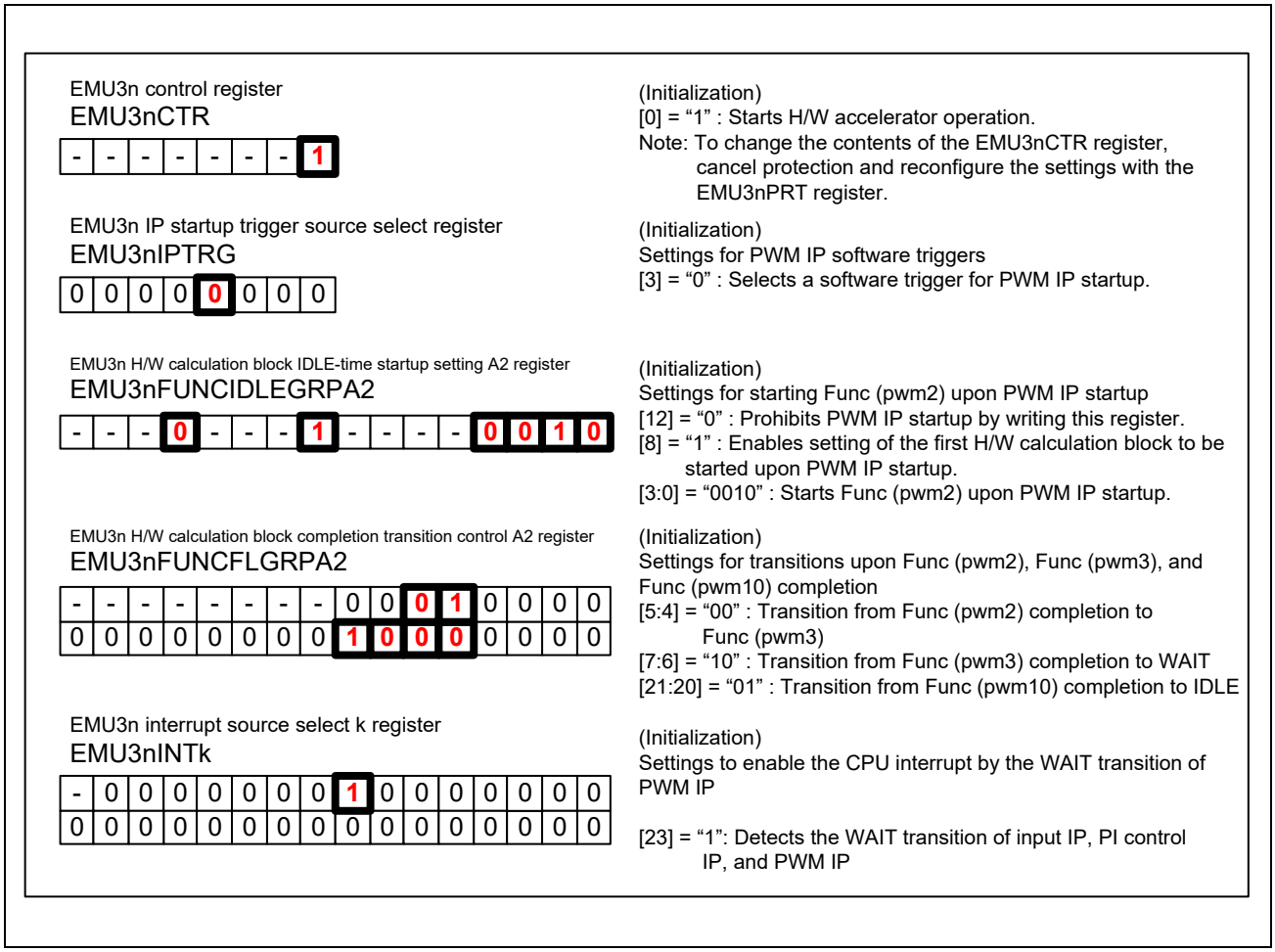


Figure 25.45 H/W Accelerator's Collaborative Operations (1) Initialization

(2) PWM IP startup to Func(pwm2) transition

The CPU manipulates the EMU3nIPSFT register to make setting for software startup of the PWM IP.

According to the initial settings of the EMU3nFUNCIDLEGRPA2 register, processing starts from H/W arithmetic block Func(pwm2) when the PWM IP is started.

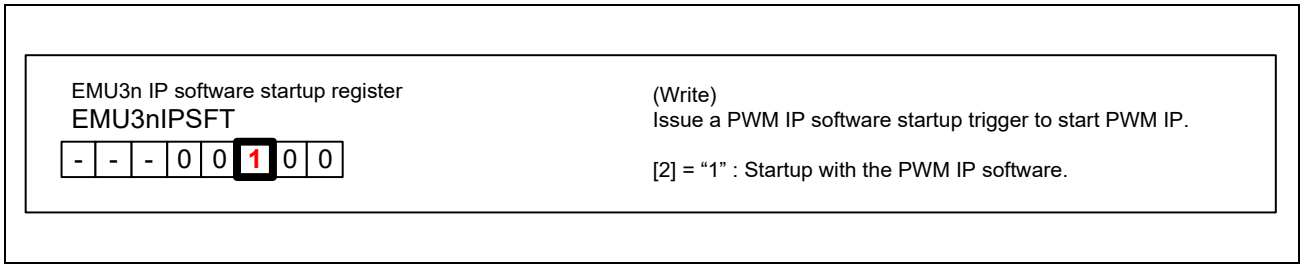


Figure 25.46 H/W Accelerator's Collaborative Operations (2) PWM IP startup to Func(pwm2)

(3) Func(pwm2) to Func(pwm3) transition

According to the initialization value ("00") of EMU3nFUNCFLGRPA2[5:4], a transition to Func(pwm3), which is the next H/W arithmetic block, occurs upon completion of the H/W arithmetic block Func(pwm2).

(4) Func(pwm3) to WAIT transition

According to the initialization value ("10") of EMU3nFUNCFLGRPA2[7:6], a transition to the WAIT state occurs in which the PWM IP remains active upon completion of the H/W arithmetic block Func(pwm3).

(5) Func(pwm3) to CPU interrupt

According to the initialization values of the (EMU3nINTk registers (k = 0 to 7), an interrupt is notified to the CPU at the timing of Func(pwm3) completion. The interrupt source can be determined by checking the EMU3nINTSD register. The H/W arithmetic block in which processing is stopped can be determined by checking the EMU3nFSMSTGRPA register. After the interrupt source is determined, the interrupt state is cleared using the EMU3nINTSDC register.

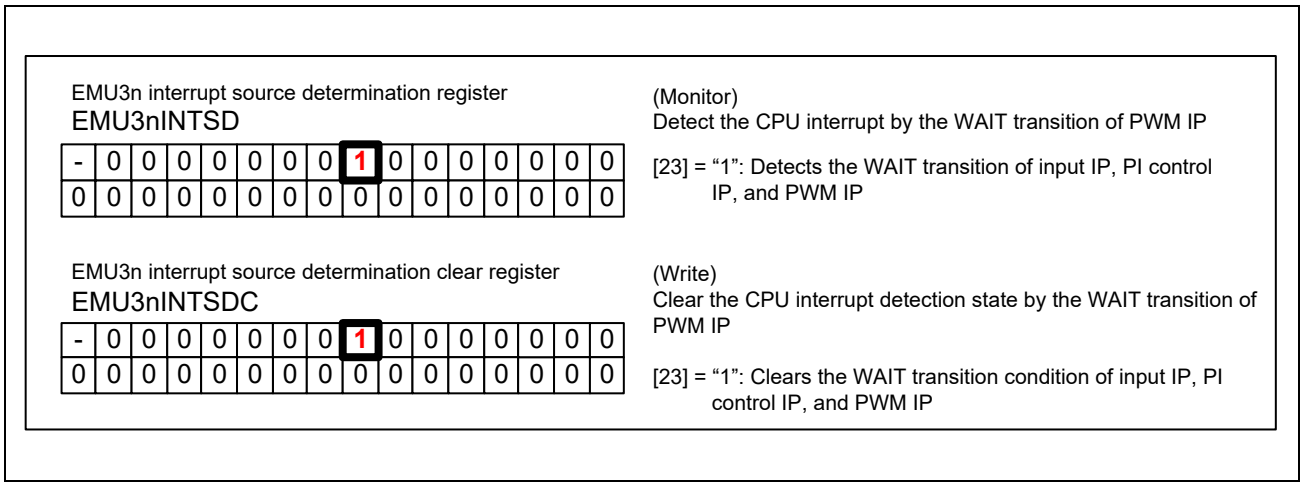


Figure 25.47 H/W Accelerator's Collaborative Operations (5) Func(pwm3) to CPU Interrupt

(6) WAIT state

The PWM IP being in the WAIT state can also be confirmed by checking the monitor registers shown in **Figure 25.48**.

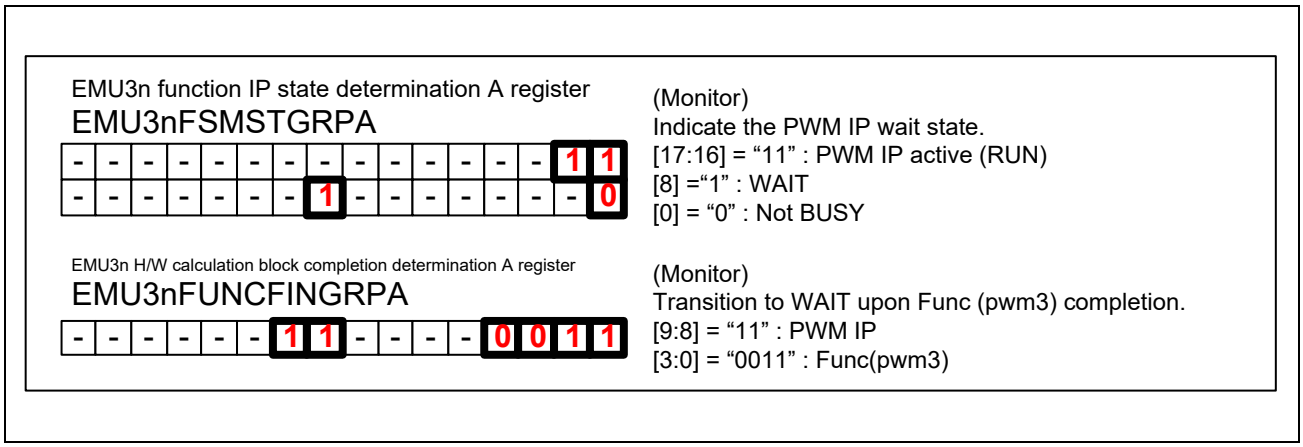


Figure 25.48 H/W Accelerator's Collaborative Operations (6) WAIT State

(7) CPU processing completion to Func(pwm10) transition

Writing the EMU3nFUNCWAITGRPA register resumes the processing at H/W arithmetic block Func(pwm10).

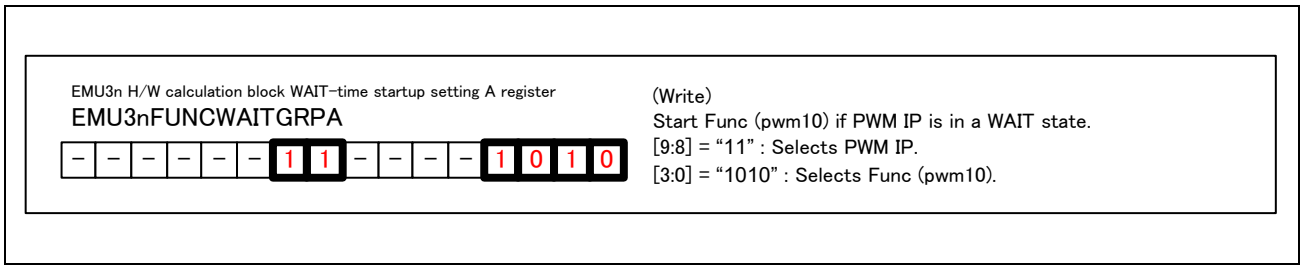


Figure 25.49 H/W Accelerator's Collaborative Operations (7) CPU Processing Completion to Func(pwm10) Transition

(8) Func(pwm10) to PWM IP completion

According to the initialized value (“01”) of the EMU3nFUNCFLGRPA2[21:20], a transition from Func(pwm10) to IDLE, i.e., PWM IP completion, occurs.

(9) PWM IP stopped

The states of the monitor registers established after the PWM IP is stopped are shown in **Figure 25.50**.

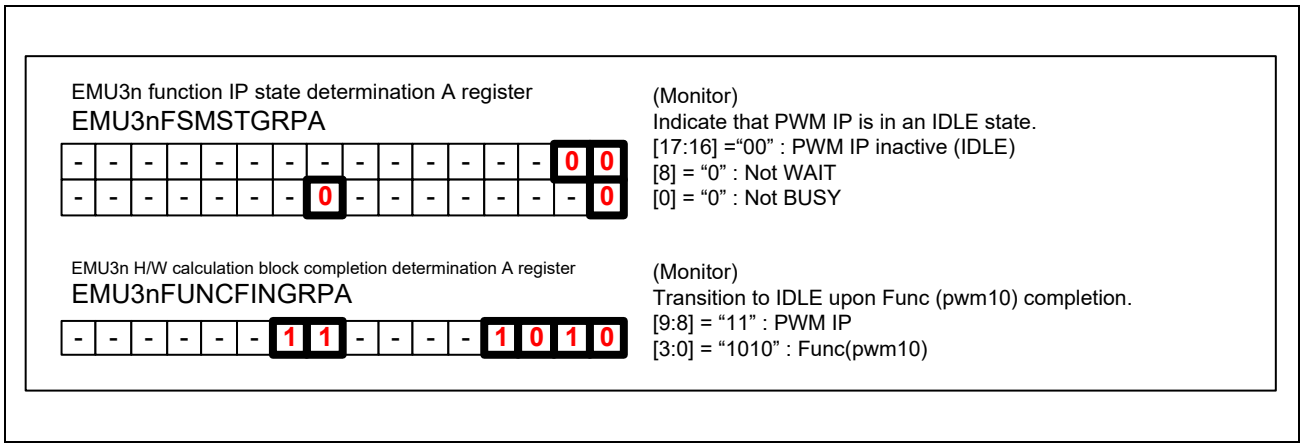


Figure 25.50 H/W Accelerator's Collaborative Operations (9) PWM IP Stopped

25.5.4 3-Phase PWM Waveform Output Control

Examples of 3-phase PWM control operation using the H/W accelerator are given below.

For details on the IIR filtering function, see **Section 25.4.9, IIR Filters.**

For details on the input IP function, see **Section 25.4.3, Input IP**.

For details on the PI control IP function, see **Section, 25.4.4, PI Control IP**.

For details on the PWM IP function, see **Section 25.4.5, PWM IP**.

25.5.4.1 Exercising 3-Phase PWM Control with the H/W Accelerator

- Startup conditions:
  - The IIR filter is started upon completion of an A/D conversion.
  - The input IP is started upon completion of IIR filtering.
  - The PI control IP is started upon completion of input IP processing.
  - The PWM IP is started upon completion of PI control IP processing.
- Operation example:

  (1) An A/D conversion is started at a trough of the carrier as a trigger.

  (2) The IIR filter is started upon completion of the A/D conversion.

  (3) The input IP is started upon completion of IIR filtering.

  (4) The PI control IP is started upon completion of input IP processing.

  (5) The PWM IP is started upon completion of PI control IP processing and calculates the PWM compare values for the next carrier.

  (6) The PWM compare values are stored in the required registers of the TSG3.

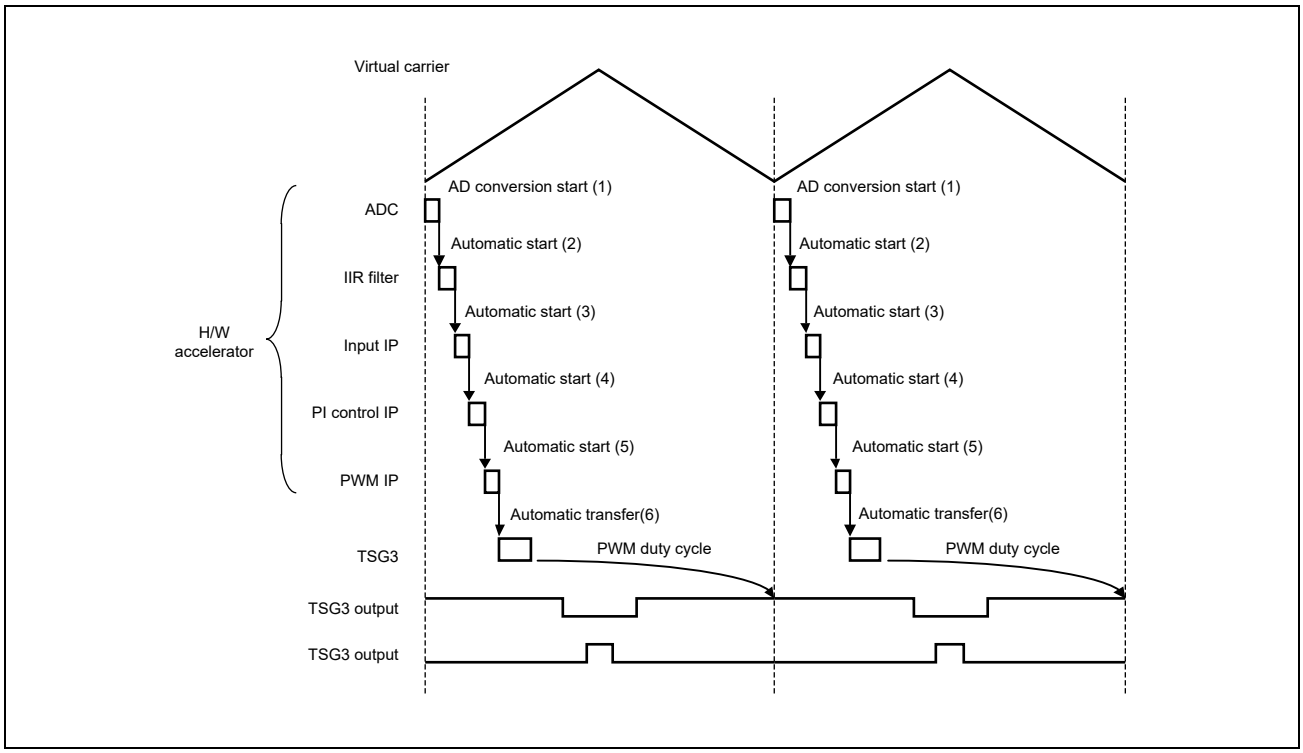


Figure 25.51 Example of 3-Phase PWM Control by the H/W Accelerator

25.5.4.2 Exercising 3-Phase PWM Control with the H/W Accelerator and CPU

- Startup conditions:
  - The IIR filter is started upon completion of an A/D conversion.
  - The input IP is started upon completion of IIR filtering.
  - The PWM IP is started on a software trigger.
- Operation example:

(1) An A/D conversion is started at a trough of the carrier as a trigger.

(2) The IIR filter is started upon completion of the A/D conversion.

(3) The input IP is started upon completion of IIR filtering.

(4) Interrupt program processing is started upon completion of input IP processing.

(5) The PWM IP is started by the interrupt program processing and calculates the PWM compare values for the next carrier.

(6) The PWM compare values are stored in the required registers of theTSG3.

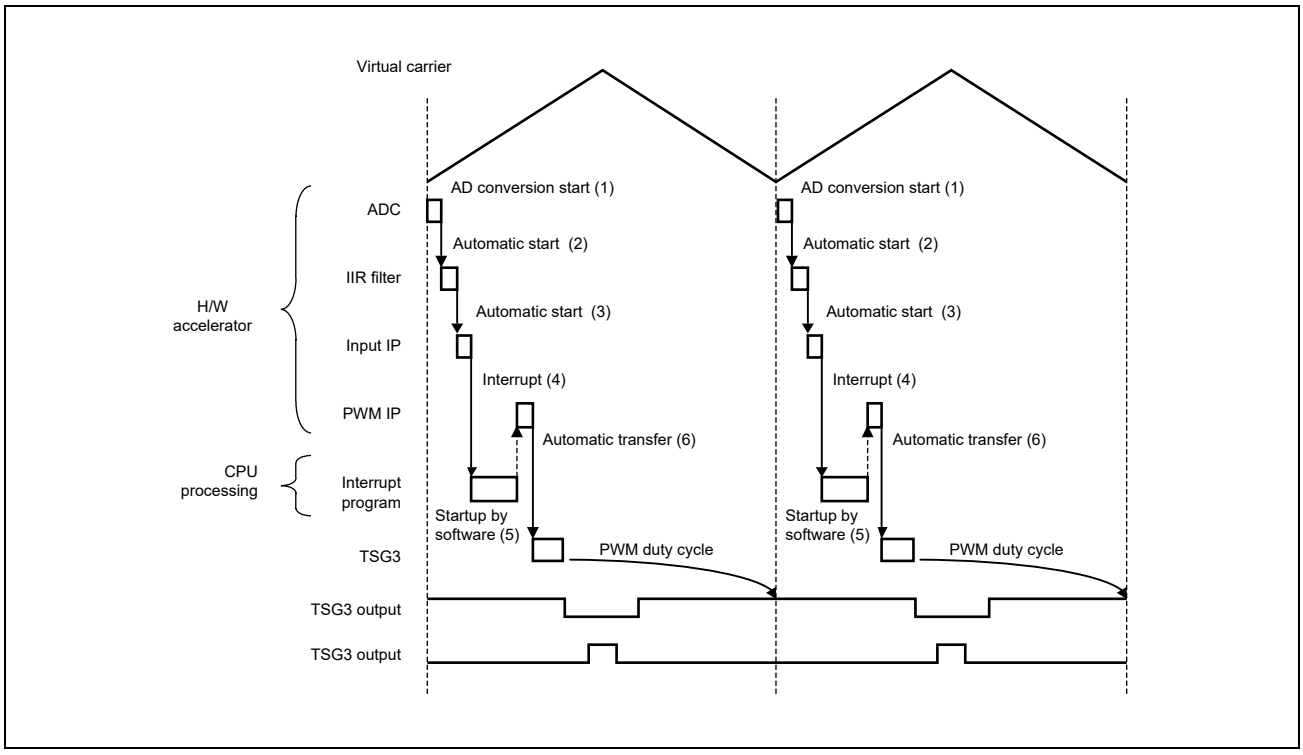


Figure 25.52 Example of 3-phase PWM Control by the H/W Accelerator and CPU

25.5.5 Rectangle Wave Output Control

An operation example of rectangle wave control using the H/W accelerator is given below.

For details on the IIR filter function, see **Section 25.4.9, IIR Filters**.

For details on the input IP function, see **Section 25.4.3, Input IP**.

For details on the rectangle IP function, see **Section 25.4.6, Rectangle IP**.

- Startup conditions:
  - The IIR filter is started upon completion of an A/D conversion.
  - The input IP is started upon completion of IIR filtering.
  - The rectangle IP is started on a software trigger.
- Operation example:

  (1) The A/D converter starts conversion and the rectangle wave output patterns are switched on a compare 0 match as a trigger.

  (2) The IIR filter is started upon completion of the A/D conversion.

  (3) The input IP is started upon completion of IIR filtering.

  (4) Interrupt program processing is started upon completion of input IP processing.

  (5) The rectangle IP is started by the interrupt program processing and makes the next EMU3nCMP0 configuration and sets up the output patterns.

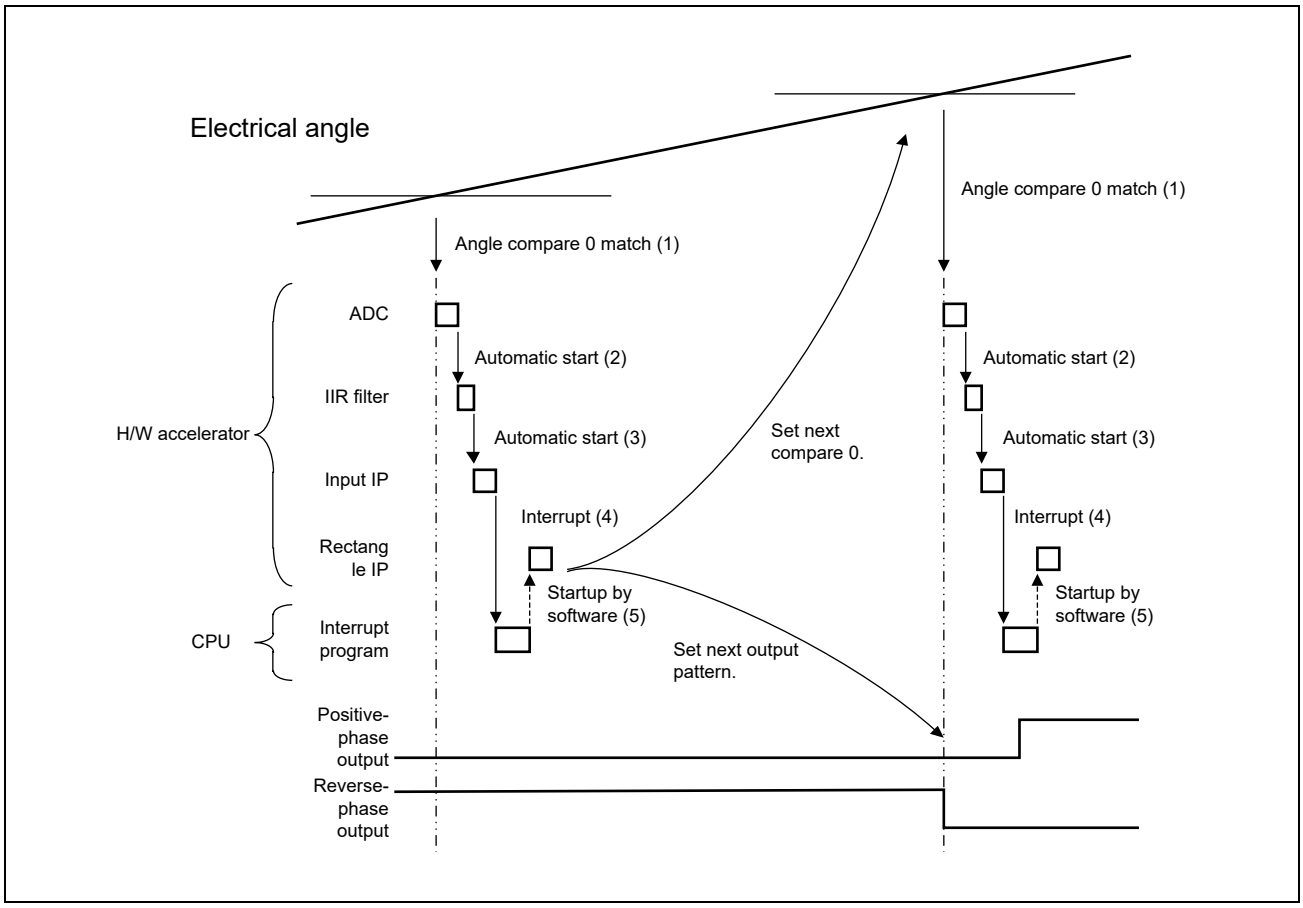


Figure 25.53 Example of Rectangle Wave Output Control

Section 26 R/D Converter (RDC3A)

This section contains a generic description of the R/D (resolver-to-digital) converter 3A (RDC3A).

The first part of this section describes specific properties in RH850/C1M-A products, such as the numbers of units and register base addresses, etc. The remainder of the section describes the functions and registers of the RDC3A.

26.1 Features of the RH850/C1M-A RDC3A

26.1.1 Numbers of Units

These microcontrollers have the following numbers of RDC3A units.

Table 26.1 Numbers of Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	2	1
Name	RDC3An (n = 0, 1)	RDC3An (n = 0)

Table 26.2 Index

Index	Meaning
n	Throughout this section, the individual RDC3A units are identified by the index “n” (n = 0, 1), for example, RDC3AnRDSTP for the RDC3An stop register.

26.1.2 Register Base Addresses

RDC3A base addresses are listed in the table below.

In general, RDC3A register addresses are given as offsets from these base addresses.

Table 26.3 Register Base Address

Base Address Name	Base Address
<RDC3A0_base>	FF78 0000$_H$
<RDC3A1_base>	FF78 1000$_H$

26.1.3 Clock Supply

The RDC3A clock supply is shown in the following table.

Table 26.4 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
RDC3An	PCLK	CLK_HSB (High speed peripheral clock)
	CCLK	CLKC_LSB (Unmodulated low speed peripheral clock)

26.1.4 Interrupt Requests

RDC3A interrupt requests are listed in the following table.

Table 26.5 Interrupt Requests

Interrupt Name	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
		1st	2nd	1st	2nd
RDC3A0					
RDC3A0 Z phase interrupt	76	67		67	
RDC3A0 RDC error interrupt	77	—	—	—	—
RDC3A0 compare match interrupt 0	78	68	—	68	—
RDC3A0 compare match interrupt 1	79	69	—	69	—
RDC3A0 compare match interrupt 2	80	70	—	70	—
RDC3A0 excitation timer (ET) interrupt	81	—	—	—	—
RDC3A0 excitation timer (ET) DMA request	—	71	—	71	—
RDC3A0 BIST completion interrupt	82	—	—	—	—
RDC3A1					
RDC3A1 Z phase interrupt	83	72	—	72	—
RDC3A1 RDC error interrupt	84	—	—	—	—
RDC3A1 compare match interrupt 0	85	73	—	73	—
RDC3A1 compare match interrupt 1	86	74	—	74	—
RDC3A1 compare match interrupt 2	87	75	—	75	—
RDC3A1 excitation timer (ET) interrupt	88	—	—	—	—
RDC3A1 excitation timer (ET) DMA request	—	76	—	76	—
RDC3A1 BIST completion interrupt	89	—	—	—	—

Note: —: No number is assigned.

Note 1. 1st: Primary channel, 2nd: Secondary channel

26.1.5 Reset Source

RDC3A reset sources are listed in the following table. RDC3A is initialized by the following reset sources.

Table 26.6 Reset Sources

Unit Name	Reset Source
RDC3An	All reset sources

26.1.6 External Input/Output Signals

External input and output signals of the RDC3A are listed in the following table.

Table 26.7 External Input and Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal Name	Products Supported	
			C1M-A2	C1M-A1
Common to RDC3A0 and RDC3A1				
RVCC	RDC3A analog power supply (5V)	RVCC	√	√
RVSS	RDC3A analog ground (0V)	RVSS	√	√
RDC3A0				
RDC3A0SINMNT*1	Input amplifier monitor output	RDC3A0SINMNT	√	√
RDC3A0S4	Resolver signal input	RDC3A0S4	√	√
RDC3A0S2	Resolver signal input	RDC3A0S2	√	√
RDC3A0S1	Resolver signal input	RDC3A0S1	√	√
RDC3A0S3	Resolver signal input	RDC3A0S3	√	√
RDC3A0COSMNT*1	Input amplifier monitor output	RDC3A0COSMNT	√	√
RDC3A0RSO	Input/output for input/output excitation signal	RDC3A0RSO	√	√
RDC3A0COM	Input/output of common voltage for input/output excitation signal	RDC3A0COM	√	√
RD0_OUT_U	Encoder pulse output	RDC3A0_OUT_U	√	√
RD0_OUT_V	Encoder pulse output	RDC3A0_OUT_V	√	√
RD0_OUT_W	Encoder pulse output	RDC3A0_OUT_W	√	√
RDC3A1				
RDC3A1SINMNT*1	Input amplifier monitor output	RDC3A1SINMNT	√	—
RDC3A1S4	Resolver signal input	RDC3A1S4	√	—
RDC3A1S2	Resolver signal input	RDC3A1S2	√	—
RDC3A1S1	Resolver signal input	RDC3A1S1	√	—
RDC3A1S3	Resolver signal input	RDC3A1S3	√	—
RDC3A1COSMNT*1	Input amplifier monitor output	RDC3A1COSMNT	√	—
RDC3A1RSO	Input/output for input/output excitation signal	RDC3A1RSO	√	—
RDC3A1COM	Input/output of common voltage for input/output excitation signal	RDC3A1COM	√	—
RD1_OUT_U	Encoder pulse output	RDC3A1_OUT_U	√	—
RD1_OUT_V	Encoder pulse output	RDC3A1_OUT_V	√	—
RD1_OUT_W	Encoder pulse output	RDC3A1_OUT_W	√	—

Note 1. Input amplifier monitor pins (RDC3AnSINMNT and RDC3AnCOSMNT) also act as the A/D converter analog input pins. Do not use these pins as input amplifier monitor output pin and A/D conversion pin at the same time.

26.2 Overview

26.2.1 Functional Overview

The R/D (resolver-to-digital) converter 3A converts the analog value (angle information) indicating the rotor angle of the resolver into a 16-bit (at maximum) digital value.

There are two conversion methods;

- Angular conversion mode 0:
  The analog value is converted to the digital value by using the analog calculation block and the digital tracking loop.
- Angular conversion mode 1:
  The analog signal is obtained by the A/D converter and converted by the digital tracking loop.

In addition to the function of converting the analog angle signal from the resolver into a digital angle signal, the RDC3A provides excitation signal output function, error detection function, and self-diagnosis function.

Table 26.8 lists the specifications for the RDC3A.

Table 26.8 RDC3A Specifications (1/2)

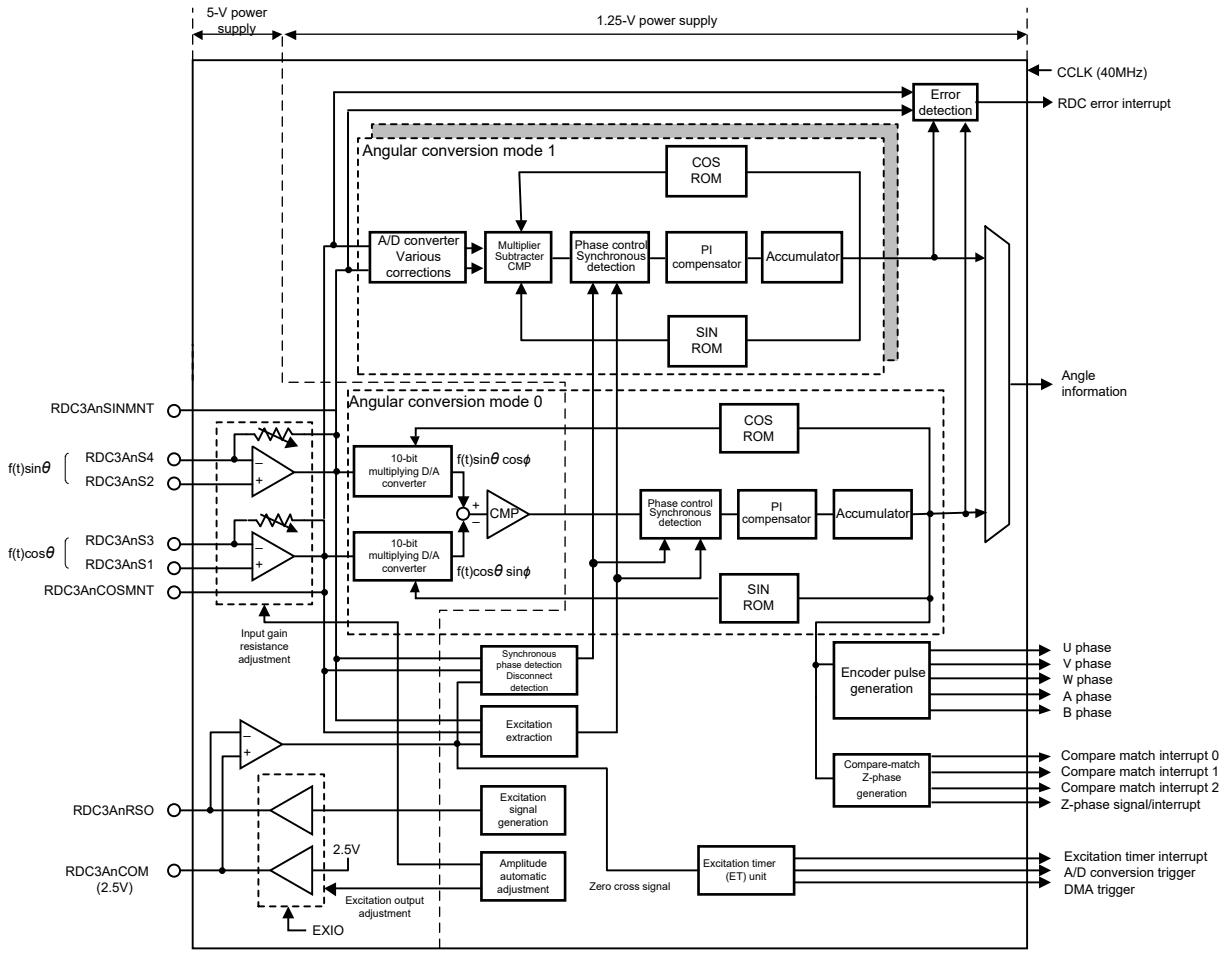
Item	Function	Description	Applicable mode	
			Angular Conversion Mode 0	Angular Conversion Mode 1
Tracking loop	Excitation signal source selection	Selects the excitation signal generated in the RDC3A (RDC3AnRSO, RDC3AnCOM) or that input from outside.	√	√
	Required sensor selection	Selects VR resolver or DC resolver.	√	√
	Excitation component extraction	Uses the excitation components extracted from the resolver signal for R/D conversion.	√	√
	PI compensator bandwidth setting	Selects from six bandwidths (five fixed and one auto-adjusted).	√	√
	Forced gain control	Improves the tracking performance when the resolver angle deviates significantly from the R/D converted angle.	√	√
	Maximum angular velocity setting	Sets a maximum angular velocity (resolution) in the range from 960,000 rpm (10 bits) to 15,000 rpm (16 bits).	√	√
	PHI angular velocity information reading	Reads the angular velocity from the PHI angle output register.	√	√
	Monitor	Reads angle information (°), angular velocity information (rpm), and control variation values (%) directly from a register.	√	√
	Angle compare	When the angle that is set in the angle compare registers 0 to 2 and the R/D converted angle match, a compare match interrupt request is generated.	√	√
	Encoder pulse output	Outputs U-phase, V-phase, W-phase, A-phase, B-phase, and Z-phase signals (4096 Edge/Revolution).	√	√

Table 26.8 RDC3A Specifications (2/2)

Item	Function	Description	Applicable mode	
			Angular Conversion Mode 0	Angular Conversion Mode 1
Sine and cosine correction function	ADC noise elimination	Discards the converted values of the RDC3AnSINMNT and RDC3AnCOSMNT signals acquired by AD conversion if they contain noise.	—	√
	Sine/cosine gain correction	Automatically detects the amplitudes of the sine and cosine signals by comparing them and corrects them.	—	√
	Sine/cosine common offset correction	Automatically detects an offset of the common levels of the sine and cosine signals input from the original common level and corrects them.	—	√
	Sine/cosine phase correction	Automatically detects the phase deviations of the excitation components of the sine and cosine signals to be input and corrects them.	—	√
	Sine/cosine angle correction	Inputs the value for correcting the sine and cosine angles to the SIN-ROM and COS-ROM tables. This function handles the mounting of the resolver at an angle toward the motor shaft by correcting the resulting orthogonality errors of the sine and cosine signals from the resolver.	√	√
Excitation signal output	Excitation signal output function (RDC3AnRSO, RDC3AnCOM)	Generates the excitation signal with the voltage buffer. Outputs the sine-wave voltage signal generated in the 7-bit D/A converter through the RDC3AnRSO pin. Voltage amplitude (Vpp) is 2 V, which can be changed with the register setting. Outputs common voltage, which is RVCC divided by 2, from the RDC3AnCOM pin.	√	√
	Automatic amplitude adjustment function	Automatically adjusts the amplitude of the input monitor signals (RDC3AnSINMNT, RDC3AnCOSMNT) into appropriate values. Targets to be adjusted: Input gain resistance and excitation signal output amplitudes	√	√
Error detection	Error detection function	Detects resolver signal error, resolver signal disconnect error, R/D conversion error, two paths comparison conversion error, resolver signal power/ground short error, and sum-of-squares amplitude error.	√	√
Self-diagnosis	Built-in self-test function	ADBIST (checks if AD conversions are successful) Angle conversion BIST (0, 45, 270°) Error detection BIST (resolver signal error, resolver signal disconnect error, conversion error, power/ground short error, and sum-of-squares amplitude error)	√	√
	ADC software BIST function	A software BIST used to determine the result of AD conversion by the CPU whose DAC code has been written to the register. Diagnosis of all 4096 codes is possible.	√	√
Others	Excitation timer (ET) function	Measures period of excitation signal Generates event signals (AD trigger, DMA request).	√	√
	PGA inversion function	In angle conversion using the ADC, this function inverts the signals output from the PGA (input amplifier) with the common potential as a center depending on their angles to reduce angle conversion errors.	—	√
Interrupts		Compare match interrupts 0 to 2 Z-phase interrupt RDC error interrupt Excitation timer interrupt BIST completion interrupt	√	√

26.2.2 Block Diagram

Figure 26.1 is a block diagram of the entire RDC3A module.



Note: Self-test circuits are omitted from this block diagram.

Note: n = 0, 1
f(t): excitation signal
RDC3AnS1 to RDC3AnS4: resolver signals
EXIO: A bit in the RDC3AnREF register
SIN ROM: Sine wave ROM
COS ROM: Cosine wave ROM
RDC3AnCOSMNT, RDC3AnSINMNT: Monitor signals output from the input amplifier

Figure 26.1 Block Diagram of the Entire RDC3A Module

26.2.3 Operating Principle

The following describes the operating principles of the RDC3A module.

This R/D converter module uses the tracking method to convert analog resolver signals to digital signals (R/D conversion).

The tracking loop runs at 20 MHz.

When the excitation signal f(t) is input to the excitation coil, f(t) • sinθ and f(t) • cosθ are output from the resolver according to the angle (θ) of the resolver rotor. These signals are input to the RDC3AnS2-RDC3AnS4 and RDC3AnS1-RDC3AnS3 pins, respectively.

These signals are amplified and then input to the multiplying D/A converter. At the same time, cos ϕ (or sin ϕ) is generated by passing the accumulator output through COS ROM (or SIN ROM) and is fed back to the corresponding D/A converter. Then, the subtraction is performed on the outputs from both D/A converters.

$$f(t) \cdot (\sin\theta \cdot \cos\phi - \cos\theta \cdot \sin\phi)$$

$$= f(t) \cdot \sin(\theta - \phi)$$

$$\approx f(t) \cdot (\theta - \phi)$$

The result is converted to 1-bit digital value by the comparator (CMP) and then passed to the digital block.

The excitation component f(t) is removed in the synchronous detection circuit to obtain the control variation $\varepsilon = \theta - \phi$.

The negative feedback control over the entire analog and digital circuits provides feedback so that the control variation becomes zero. When $\theta = \phi$, the analog angle information from the resolver has been converted to digital angle ϕ (16-bit wide). **Figure 26.2** describes the PI compensator and accumulator.

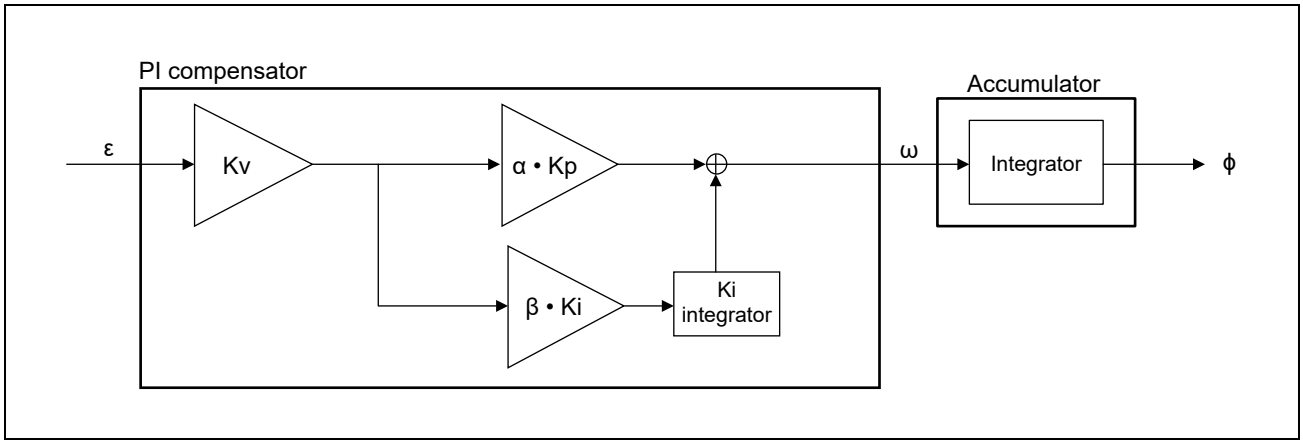


Figure 26.2 PI Compensator and Accumulator

The PI compensator converts the control variation according to the following formula, and passes the result to the accumulator circuit

$\omega = (\alpha \bullet Kp + (\beta \bullet Ki)/(s \bullet T)) \bullet Kv \bullet \varepsilon$

Kv, Kp, Ki: control coefficients that can be set in a register

α, β: fixed values

s: Laplace variable

T: Integration time constant

ω: PI compensator output, angular velocity information

The accumulator circuit calculates the angle ϕ based on the angular velocity information ω.

26.3 Register

26.3.1 List of Registers

The following table lists the RDC3A registers.

For <RDC3An_base>, see **Section 26.1.2, Register Base Addresses**.

Table 26.9 List of Registers

Module Name	Register Name	Symbol	Address
RDC3An	RDC Stop register	RDC3AnRDSTP	<RDC3An_base> + 0000_H
RDC3An	Control gain select register 0	RDC3AnPI0	<RDC3An_base> + 0004_H
RDC3An	Control gain select register 1	RDC3AnPI1	<RDC3An_base> + 0008_H
RDC3An	PHI compare setting register 0	RDC3AnPHICP0	<RDC3An_base> + $000C_H$
RDC3An	PHI compare setting register 1	RDC3AnPHICP1	<RDC3An_base> + 0010_H
RDC3An	PHI compare setting register 2	RDC3AnPHICP2	<RDC3An_base> + 0014_H
RDC3An	Sine/cosine angle correction register	RDC3AnSCCOR0	<RDC3An_base> + 0018_H
RDC3An	Sine/cosine correction register 0	RDC3AnSCCOR1	<RDC3An_base> + $001C_H$
RDC3An	Sine/cosine correction register 1	RDC3AnSCCOR2	<RDC3An_base> + 0020_H
RDC3An	Sine/cosine correction register 2	RDC3AnSCCOR3	<RDC3An_base> + 0024_H
RDC3An	Automatic amplitude adjustment register 0	RDC3AnATMNT0	<RDC3An_base> + 0028_H
RDC3An	Automatic amplitude adjustment register 1	RDC3AnATMNT1	<RDC3An_base> + $002C_H$
RDC3An	Error detection register 0	RDC3AnDIAG0	<RDC3An_base> + 0030_H
RDC3An	Error detection register 1	RDC3AnDIAG1	<RDC3An_base> + 0034_H
RDC3An	Error detection register 2	RDC3AnDIAG2	<RDC3An_base> + 0038_H
RDC3An	Error detection output register 0	RDC3AnDGOUT0	<RDC3An_base> + $003C_H$
RDC3An	Error detection output register 1	RDC3AnDGOUT1	<RDC3An_base> + 0040_H
RDC3An	BIST register 0	RDC3AnBIST0	<RDC3An_base> + 0044_H
RDC3An	BIST register 1	RDC3AnBIST1	<RDC3An_base> + 0048_H
RDC3An	Excitation setting register	RDC3AnREF	<RDC3An_base> + $004C_H$
RDC3An	Encoder register 0	RDC3AnENC0	<RDC3An_base> + 0050_H
RDC3An	Encoder register 1	RDC3AnENC1	<RDC3An_base> + 0054_H
RDC3An	Encoder register 2	RDC3AnENC2	<RDC3An_base> + 0058_H
RDC3An	Angular velocity register	RDC3AnOMG	<RDC3An_base> + $005C_H$
RDC3An	Test bus register	RDC3AnTBUS	<RDC3An_base> + 0064_H
RDC3An	Angular conversion mode select register	RDC3AnADRD	<RDC3An_base> + 0068_H
RDC3An	ET control register	RDC3AnETEN	<RDC3An_base> + $006C_H$
RDC3An	ET capture register	RDC3AnETCAP	<RDC3An_base> + 0070_H
RDC3An	ET zero-crossing counter register	RDC3AnETMCNT	<RDC3An_base> + 0074_H
RDC3An	Digital operation register 0	RDC3AnDCUR0	<RDC3An_base> + $007C_H$
RDC3An	Digital operation register 1	RDC3AnDCUR1	<RDC3An_base> + 0080_H
RDC3An	BIST end setting register 0	RDC3AnBISTFX0	<RDC3An_base> + 0088_H
RDC3An	BIST end setting register 1	RDC3AnBISTFX1	<RDC3An_base> + 0094_H
RDC3An	12-bit SAR-ADC digital circuit block setting register 1	RDC3AnADSTD1	<RDC3An_base> + $00AC_H$
RDC3An	Error detection register 3	RDC3AnDIAG3	<RDC3An_base> + $00B0_H$
RDC3An	Error detection register 4	RDC3AnDIAG4	<RDC3An_base> + $00B4_H$

26.3.2 RDC3AnRDSTP – RDC Stop Register

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0000_H

Value after reset $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MNTC	—	—	—	—	—	—	—	ANSTP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Table 26.10 RDC3AnRDSTP Register Contents

Bit Position	Bit Name	Description
31 to 9	Reserved	These bits are read as 0. The write value should be 0.
8	MNTC	Sinmnt/Cosmnt External Pin Setting The sinmnt and cosmnt external pins are set as follows. 0: The external output pins for sinmnt and cosmnt are left open-circuit. 1: The sinmnt and cosmnt signals are output through the external output pins.
7 to 1	Reserved	These bits are read as 0. The write value should be 0.
0	ANSTP	R/D Converter Stop 0: Analog circuits are running. 1: All the power supply circuits running at 5V enter the following state. – All of them are disabled. – All of the pins (SINMNT, COSMNT, S1, S2, S3, S4, RSO, COM) are in the high-impedance state.

26.3.3 RDC3AnPI0 ─ Control Gain Select Register 0

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0004_H

Value after reset $0002\ 0010_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	KVMS[1:0]		DVW[1:0]		—	KPF	KPS[1:0]		—	KIS[2:0]			—	DEVCK[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LKVS[3:0]				HKVS[3:0]				—	—	—	BWCS	—	LPGS[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Table 26.11 RDC3AnPI0 Register Contents (1/3)

Bit Position	Bit Name	Description
31, 30	KVMS[1:0]	Kv Gain Method Select Selects the Kv gain method. b31 b30 0 0: 1-step-shifted 12-level AGC (default) 0 1: 1-step-shifted 7-level AGC 1 0: 1-step-shifted 2-level AGC 1 1: Fixed Kv
29, 28	DVW[1:0]	ERR Deviation Weighting Selects weighting of ERR deviation. b29 b28 0 0: ×1 (default) 0 1: ×3 1 0: ×5 1 1: ×7
27	Reserved	This bit is read as 0. The write value should be 0.
26	KPF	Kp Gain Quadruple Selects whether or not to quadruple the gain on the Kp side of the PI compensator.*1 0: Kp gain is not quadrupled. (×1) 1: Kp gain is quadrupled.
25, 24	KPS[1:0]	Kp Gain Select Selects the Kp gain in the PI compensator*1 b25 b24 0 0: ×1 (default) 0 1: ×0.25 1 0: ×0.5 1 1: ×2
23	Reserved	This bit is read as 0. The write value should be 0.

Table 26.11 RDC3AnPI0 Register Contents (2/3)

Bit Position	Bit Name	Description
22 to 20	KIS[2:0]	Ki Gain Select Sets the Ki gain in the PI compensator. b22 b21 b20 0 0 0: ×1 (default) 0 0 1: ×0.125 0 1 0: ×0.25 0 1 1: ×0.5 1 0 0: ×2 1 0 1: ×4 1 1 0: ×8 1 1 1: ×16
19	Reserved	This bit is read as 0. The write value should be 0.
18 to 16	DEVCK[2:0]	Control Variation Determination Clock Select*2 Selects the clock period used for determining the control variation. 0 0 0: 50-µs* clock period 0 0 1: 100-µs* clock period 0 1 0: 200-µs* clock period (default) 0 1 1: 25-µs* clock period 1 0 0: 400-µs* clock period 1 0 1: 800-µs* clock period 1 1 0: 50-µs* clock period 1 1 1: 50-µs* clock period
15 to 12	LKVS[3:0]	Low Kv Gain Select Sets the gain in the lower Kv side when the fixed Kv method or 2-level AGC method is selected. b15 b14 b13 b12 0 0 0 0: ×1 (default) 0 0 0 1: ×0.0625 0 0 1 0: ×0.125 0 0 1 1: ×0.25 0 1 0 0: ×0.5 0 1 0 1: ×1 0 1 1 0: ×2 0 1 1 1: ×4 1 0 0 0: ×8 1 0 0 1: ×16 1 0 1 0: ×32 1 0 1 1: ×64 1 1 0 0: ×128 1 1 0 1: ×1 1 1 1 0: ×1 1 1 1 1: ×1

Table 26.11 RDC3AnPI0 Register Contents (3/3)

Bit Position	Bit Name	Description
11 to 8	HKVS[3:0]	High Kv Gain Select Sets the gain in the higher Kv side when the fixed Kv method or 2-level AGC method is selected. b11 b10 b9 b8 0 0 0 0: ×32 (default) 0 0 0 1: ×0.0625 0 0 1 0: ×0.125 0 0 1 1: ×0.25 0 1 0 0: ×0.5 0 1 0 1: ×1 0 1 1 0: ×2 0 1 1 1: ×4 1 0 0 0: ×8 1 0 0 1: ×16 1 0 1 0: ×32 1 0 1 1: ×64 1 1 0 0: ×128 1 1 0 1: ×32 1 1 1 0: ×32 1 1 1 1: ×32
7 to 5	Reserved	These bits are read as 0. The write value should be 0.
4	BWCS	Bandwidth Setting Selects the PI compensator setting method. 0: Sets the coefficients by using the RDC3AnPI0 register. 1: Using the LPGS[2:0] bits in this register (see the description in the LPGS[2:0] for the set value.)
3	Reserved	This bit is read as 0. The write value should be 0.
2 to 0	LPGS[2:0]	Loop Gain Select By setting the BWCS bit in this register to 1, the LPGS[2:0] bits are enabled and the PI compensator can be set by using them. See **Table 26.17**

Note: The timing value is when the CCLK is running at 40 MHz.

Note 1. Do not set the DVW[1:0], KPS[1:0], KPF, and HKVS[3:0] bits in the RDC3AnPI0 register to values listed in **Table 26.12** and **Table 26.13**. In case of using the combinations listed in these tables, the P component in the PI compensator overflows, resulting in unexpected value if the Kv gain is large.

Note 2. Set the control variation determination clock period to be longer than the excitation signal period currently used. For example, when a 10-kHz (100-µs period) excitation signal is used, set the control variation determination clock period to 100, 200, 400 or 800 µs. Do not select 50 or 25 µs.

Table 26.12 Prohibited Combinations of DVW[1:0], KPS[1:0], and KPF
• When KVMS[1:0] = 00 or 01 (at 12- or 7- level AGC)

DVW[1:0]	KPS[1:0]	KPF
00 (×1)	11 (×2)	1 (×4)
01 (×3)	11 (×2)	1 (×4)
01 (×3)	00 (×1)	1 (×4)
10 (×5)	11 (×2)	1 (×4)
10 (×5)	11 (×2)	0 (×1)
10 (×5)	00 (×1)	1 (×4)
10 (×5)	10 (×0.5)	1 (×4)
11 (×7)	11 (×2)	1 (×4)
11 (×7)	11 (×2)	0 (×1)
11 (×7)	00 (×1)	1 (×4)
11 (×7)	10 (×0.5)	1 (×4)

Table 26.13 Prohibited Combinations of DVW[1:0], KPS[1:0], KPF, and HKVS[3:0]
• When KVMS[1:0] = 10 or 11 (at 2-level AGC with fixed Kv)

DVW[1:0]	KPS[1:0]	KPF	HKVS[3:0]
00 (×1)	11 (×2)	1 (×4)	1100 (×128)
01 (×3)	11 (×2)	1 (×4)	1100 (×128)
01 (×3)	11 (×2)	1 (×4)	1011 (×64)
01 (×3)	00 (×1)	1 (×4)	1100 (×128)
10 (×5)	11 (×2)	1 (×4)	1100 (×128)
10 (×5)	11 (×2)	1 (×4)	1011 (×64)
10 (×5)	11 (×2)	1 (×4)	1010 (×32)
10 (×5)	11 (×2)	0 (×1)	1100 (×128)
10 (×5)	00 (×1)	1 (×4)	1100 (×128)
10 (×5)	00 (×1)	1 (×4)	1011 (×64)
10 (×5)	10 (×0.5)	1 (×4)	1100 (×128)
11 (×7)	11 (×2)	1 (×4)	1100 (×128)
11 (×7)	11 (×2)	1 (×4)	1011 (×64)
11 (×7)	11 (×2)	1 (×4)	1010 (×32)
11 (×7)	11 (×2)	0 (×1)	1100 (×128)
11 (×7)	00 (×1)	1 (×4)	1100 (×128)
11 (×7)	00 (×1)	1 (×4)	1011 (×64)
11 (×7)	10 (×0.5)	1 (×4)	1100 (×128)

Kv Gain Method Select Bits

These bits select the Kv gain method in the PI compensator. When the AGC (Auto Gain Control) method is selected, the Kv gain is automatically selected according to the amount of control variation. It is recommended to use the value after reset (12-level AGC method).

- 12-level AGC method (default)

Table 26.14 lists the control variation amount and the value of the selected Kv gain for the 12-level AGC. The control variation amount indicates a bias of control variation ε (High or Low) within the determination clock period. With

regard to the resolver angle signal θ and the R/D converter output angle signal ϕ, the High and the Low appear in equal proportions if θ and ϕ are equal. In such a case, the control variation amount is ± 0%. If ϕ is completely behind θ, the ε always becomes high. In this case, the control variation amount is +100%. If ϕ is completely ahead of θ, ε always becomes Low. In this case, the control variation amount is –100%. The determination clock period can be selected by the DEVCK[2:0] bits.

Table 26.14 Control Variation Amount and Kv Gain for 12-Level AGC

Control Variation Amount (Absolute Value)	Kv Gain
After released from a reset	×128
76.8% to 100%	×64
64.0% to 76.8%	×32
57.6% to 64.0%	×16
51.2% to 57.6%	×8
44.8% to 51.2%	×4
38.4% to 44.8%	×2
32.0% to 38.4%	×1
25.6% to 32.0%	×0.5
19.2% to 25.6%	×0.25
12.8% to 19.2%	×0.125
0.0% to 12.8%	×0.0625

If the Kv gain falles below ×128 after a reset, the Kv returns to ×128 on any of the following conditions:

1. The KIRST bit is set to 1 when the AGDS bit is 0 (Ki reset)
2. Recovery from the resolver signal error state when the AGDS bit is 0.

The Kv gain does not return to ×128 on the above conditions when the AGDS bit is 1.

- 7-level AGC method

Table 26.15 lists the control variation amount and the value of the selected Kv gain for the 7-level AGC.

Table 26.15 Control Variation Amount and Kv Gain for 7-Level AGC

Control Variation Amount (Absolute Value)	Kv Gain
After released from a reset	×128
76.8% to 100 %	×64
51.2% to 76.8%	×16
38.4% to 51.2%	×4
25.6% to 38.4%	×1
12.8% to 25.6%	×0.25
0.0 % to 12.8%	×0.0625

If the Kv gain falls below × 128 after a reset, the Kv returns to × 128 on the following conditions:

1. The KIRST bit is set to 1 when the AGDS bit is 0 (Ki reset)
2. Recovery from the resolver signal error state when the AGDS bit is 0.

The Kv gain does not return to ×128 on the above conditions when the AGDS bit is 1.

- 2-level AGC method

Table 26.16 lists the control variation amount and the value of the selected Kv gain for the 2-level AGC. The low and high gains can be set using bits LKVS[3:0] and HKVS[3:0], respectively.

Table 26.16 Control Variation Amount and Kv Gain for 2-Level AGC

Control Variation Amount (Absolute Value)	Kv Gain
76.8% to 100%	Makes transition to the high Kv gain
25.6% to 76.8%	Kv gain maintained (no transition)
0.0 % to 25.6%	Makes transition to the low Kv gain

If the Kv gain falles below ×128 after a reset, the Kv returns to ×128 on any of the following conditions:

1. The KIRST bit is set to 1 when the AGDS bit is 0 (Ki reset)
2. Recovery from the resolver signal error state when the AGDS bit is 0.

The Kv gain does not return to ×128 on the above conditions when the AGDS bit is 1.

- Fixed Kv method

If the fixed Kv method is selected, the module operates with the low Kv gain irrespective of the control variation amount. The module, however, makes a transition to the high Kv gain only when an error is detected and then the erorr signal makes a transition to 0 (recovery from the error). After that, the module makes a transition to the low Kv gain side when the absolute value of the control variation amount falls below 25.6%.

Loop Gain Select Bits

Combinations of values in the LPGS[2:0] bits and corresponding settings in the PI compsensator when the BWCS bit is 1 are shown in **Table 26.17**.

Table 26.17 LPGS[2:0] and Corresponding Settings in the PI Compensator

	LPGS[2:0]			Settings in PI Compensator
BWCS	b2	b1	b0	Bandwidth
0	X	X	X	Set by the RDC3AnPI0 register
1	0	0	0	800 Hz
1	0	1	1	1500 Hz
1	1	0	0	1000 Hz
1	1	0	1	500 Hz
1	1	1	0	200 Hz
1	1	1	1	Adjusted automatically.

Note: The default setting is b2:b0 = 000 which sets the bandwidth as 800 Hz.

Note: Settings other than those listed above are prohibited.

26.3.4 RDC3AnPI1 ─ Control Gain Select Register 1

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0008_H

Value after reset $0001\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SAGD	—	—	—	AGCD	—	—	—	—	—	—	—	AGDS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	AGST[3:0]				—	—	—	—	—	MAXV[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 26.18 RDC3AnPI1 Register Contents (1/2)

Bit Position	Bit Name	Description
31 to 29	Reserved	These bits are read as 0. The write value should be 0.
28	SAGD	Short-Period BIST Recovery Forced Gain Control Function Disable Controls whether the RD converter makes a transition to the forced gain control state on completion of a short-period BIST (ADBIST, resolver signal error BIST, disconnect error BIST, power/ground short error BIST). 0: Use the forced control gain function. 1: Do not use the forced gain control function.
27 to 25	Reserved	These bits are read as 0. The write value should be 0.
24	AGCD	Forced Gain Control Function Disable Selects whether or not to use the forced gain control function*1. 0: Use the forced gain control function. 1: Do not use the forced gain control function.
23 to 17	Reserved	These bits are read as 0. The write value should be 0.
16	AGDS	AGC Kv High Gain Transition Restriction Restricts the AGC Kv gain from going high. 0: The Kv goes high in response to excitation errors or after a Ki reset. 12-level AGC is selected: Kv is fixed to ×128 7-level AGC is selected: Kv is fixed to ×128 2-level AGC is selected: Kv is fixed to the higher side 1: The Kv does not go high in response to excitation errors or after a Ki reset.
15 to 12	Reserved	These bits are read as 0. The write value should be 0.

Table 26.18 RDC3AnPI1 Register Contents (2/2)

Bit Position	Bit Name	Description
11 to 8	AGST[3:0]	Short-Period BIST Recovery Initial AGC Gain Sets the initial value for the Kv gain on transitions to the forced gain control state after completion of a short-period BIST (ADBIST, resolver signal error BIST, disconnect error BIST, power/ground short error BIST) or ending on detection of power/ground short error. b11 b10 b9 b8 Initial value for the Kv gain 0 0 0 0: ×4 (default) 0 0 0 1: ×0.0625 0 0 1 0: ×0.125 0 0 1 1: ×0.25 0 1 0 0: ×0.5 0 1 0 1: ×1 0 1 1 0: ×2 0 1 1 1: ×4 1 0 0 0: ×8 1 0 0 1: ×16 1 0 1 0: ×32 1 0 1 1: ×64 1 1 0 0: ×128 1 1 0 1: ×4 1 1 1 0: ×4 1 1 1 1: ×4
7 to 3	Reserved	These bits are read as 0. The write value should be 0.
2 to 0	MAXV[2:0]	Maximum Angular Velocity Select Sets the maximum angular velocity. See **Table 26.19**.

Note 1. To change the setting of AGCD, write 1 to the KIRST bit after the setting has been changed. Otherwise, the phi angle values are output under the free-running condition.
The setting of AGCD should be changed while the resolver is not running.

Maximum Angular Velocity Setting Function

Table 26.19 lists the maximum angular velocity selected by the MAXV bits and corresponding R/D conversion resolutions.

Table 26.19 Maximum Angular Velocity Select Bit Settings

b2	b1	b0	Maximum Angular Velocity (min^{-1})	Resolution (bits)
0	0	0	120000*1	13
0	0	1	240000*1	12 (default)
0	1	0	480000*1	11
0	1	1	960000*1	10
1	0	0	15000*1	16
1	0	1	60000*1	14
1	1	0	240000*1	12
1	1	1	240000*1	12

Note 1. The timing value is when the CCLK is running at 40 MHz.

26.3.5 RDC3AnPHICP0 ─ PHI Compare Setting Register 0

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $000C_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	INTCLR[2:0]			—	—	—	—	—	—	—	IRS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	INTFLG[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.20 RDC3AnPHICP0 Register Contents

Bit Position	Bit Name	Description
31 to 27	Reserved	These bits are read as 0. The write value should be 0.
26	INTCLR[2]	Compare Match Interrupt 2 Clear Writing 1 to this bit clears the INTFLG2 bit to 0. This bit becomes 0 after two clock cycles have elapsed following it being set to 1.
25	INTCLR[1]	Compare Match Interrupt 1 Clear Writing 1 to this bit clears the INTFLG1 bit to 0. This bit becomes 0 after two clock cycles have elapsed following it being set to 1.
24	INTCLR[0]	Compare Match Interrupt 0 Clear Writing 1 to this bit clears the INTFLG0 bit to 0. This bit becomes 0 after two clock cycles have elapsed following it being set to 1.
23 to 17	Reserved	These bits are read as 0. The write value should be 0.
16	IRS	Compare Match Interrupt Request Signal Select 0: Compare match signal 1: Signal latching compare match signal
15 to 3	Reserved	These bits are read as 0. The write value should be 0.
2	INTFLG[2]	Compare Match Interrupt 2 Flag 0: Interrupt request has not been generated. 1: Interrupt request has been generated.
1	INTFLG[1]	Compare Match Interrupt 1 Flag 0: Interrupt request has not been generated. 1: Interrupt request has been generated.
0	INTFLG[0]	Compare Match Interrupt 0 Flag 0: Interrupt request has not been generated. 1: Interrupt request has been generated.

26.3.6 RDC3AnPHICP1 – PHI Compare Setting Register 1

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0010_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMP1[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMP0[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.21 RDC3AnPHICP1 Register Contents

Bit Position	Bit Name	Description
31 to 16	CMP1[15:0]	Phi Compare Value 1 Sets the angle compare value with a 16-bit width.
15 to 0	CMP0[15:0]	Phi Compare Value 0 Sets the angle compare value with a 16-bit width.

26.3.7 RDC3AnPHICP2 – PHI Compare Setting Register 2

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0014_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMP2[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.22 RDC3AnPHICP2 Register Contents

Bit Position	Bit Name	Description
31 to 16	Reserved	These bits are read as 0. The write value should be 0.
15 to 0	CMP2[15:0]	Phi Compare Value 2 Sets the phi to be compared for matching at a width of 16 bits.

26.3.8 RDC3AnSCCOR0 – Sine/Cosine Angle Correction Register

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0018_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	COSPO[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SINPO[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.23 RDC3AnSCCOR0 Register Contents

Bit Position	Bit Name	Description
31 to 28	Reserved	These bits are read as 0. The write value should be 0.
27 to 16	COSPO[11:0]	Cosine Phase Correction Specifies the angular correction value for the phi to be input to the COS ROM table as a 12-bit singed integer.*1 b27-b16 000_H: 0° (no correction) $7FF_H$: +180° (maximum correction in the positive direction) 800_H: – 180° (maximum correction in the negative direction)
15 to 12	Reserved	These bits are read as 0. The write value should be 0.
11 to 0	SINPO[11:0]	Sine Phase Correction Specifies the angular correction value for the phi to be input to the SIN ROM table as a 12-bit signed integer.*1 b11-b0 000_H: 0° (no correction) $7FF_H$: +180° (maximum correction in the positive direction) 800_H: – 180° (maximum correction in the negative direction)

Note 1. Set 000_H (0°) in these bits when executing an angle conversion BIST.

26.3.9 RDC3AnSCCOR1 ─ Sine/Cosine Correction Register 0

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $001C_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	PHCST	—	GNCST	CMCLT[1:0]		CMCSL[1:0]		GNCLT[1:0]		GNCSL[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PHCLT[1:0]		PHCSL[1:0]		—	NSRSL	GNCND	SGLMD	GNJSP	GNCNS[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.24 RDC3AnSCCOR1 Register Contents (1/3)

Bit Position	Bit Name	Description
31 to 27	Reserved	These bits are read as 0. The write value should be 0.
26	PHCST	Phase Correction Start Writing 1 to this bit starts phase correction. Set this bit to 1 after the RDC has entered the angular tracking state. This bit returns to 0 after two clock cycles has elapsed following it being set to 1. This bit is effective only for once after a reset.
25	Reserved	This bit is read as 0. The write value should be 0.
24	GNCST	Gain Correction Start Writing 1 to this bit starts gain correction or common offset correction. Set this bit to 1 after the RDC has entered the angular tracking state and the resolver has rotated through an electrical angle of at least 180°. This bit returns to 0 after two clock cycles has elapsed following it being set to 1.
23, 22	CMCLT[1:0]	Common Offset Correction Limit Sets limitation for the range of correcting the common offset value. If the value for correction obtained from the input signal exceeds the set limitation, the set value is used for correction. b23 b22 Limitation on the value for correction 0 0: ± 0.0312 × RVCC (approx. ± 150 mV) (default) 0 1: ± 0.0156 × RVCC (approx. ± 78 m V) 1 0: ± 0.125 × RVCC (approx. ± 625 mV) 1 1: ± 0 × RVCC (± 0 mV) (no correction applied)
21, 20	CMCSL[1:0]	Common Offset Correction Type Select Selects the type of correcting by the common offset value. b21 b20 Type of correction 0 0: No correction (fixed to 00_H, this is the default setting) 0 1: Always use the fixed offset correction value set by CSOSN[9:0] and CCOSN[9:0]. 1 0: Use the offset correction value obtained from the calculation. The timing to apply this value depends on the setting of the GNCSL bit. 1 1: Setting prohibited

Table 26.24 RDC3AnSCCOR1 Register Contents (2/3)

Bit Position	Bit Name	Description
19, 18	GNCLT[1:0]	Gain Correction Range Limit Sets limitation for the range of correcting the gain value. If the value for correction obtained from the input signal exceeds the set limitation, the set value is used for correction. b19 b18 Limitation on the value for correction 0 0: ± 20% (default) 0 1: ± 10% 1 0: ± 40% 1 1: ± 0% (no correction applied)
17, 16	GNCSL[1:0]	Gain Correction Type Select Selects the type of correcting by the gain value. b17 b16 Type of correction 0 0: No correction (fixed to 1024, this is the default setting) 0 1: Always use the fixed correction value set by GNCNM[10:0] 1 0: Use the correction value obtained from the calculation at the time of GNCST being set 1 1: Use the correction value obtained from the calculation at the time of z-phase output following GNCST being set (correct the value on every single rotation).
15 to 12	Reserved	These bits are read as 0. The write value should be 0.
11, 10	PHCLT[1:0]	Phase Correction Limit Sets limitation for the range of correcting the sine and cosine phases. If the value for delay obtained from the input signal exceeds the set limitation, the set value is used for delay compensation. b11 b10 Limitation of delay compensation 0 0: 3 µs (default) 0 1: 1 µs 1 0: 6 µs 1 1: 0 µs (no correction applied)
9, 8	PHCSL[1:0]	Sine/Cosine Phase Correction Type Select Selects the type of correction for the sine and cosine phases. b9 b8 Type of correction 0 0: No correction (default) 0 1: Obtain the average of phase deviations on the sixteen crossings of the excitation signals after the PHCST bit is set to 1. The value for correction is updated every time 1 is written to the PHCST bit. 1 0: Obtain the average of phase deviations on the sixteen crossings of the excitation signals after the PHCST bit is set to 1. The value for correction obtained in the first round is retained. 1 1: Use the fixed correction value set by PHCNM[5:0] and PHSNM[5:0].
7	Reserved	This bit is read as 0. The write value should be 0.
6	NSRSL	AD Noise Elimination Target Select 0: Apply the AD noise filter to the sine, cosine, gain, and common offset compensation calculating circuits (default). 1: Apply the AD noise filter to the sine, cosine, gain, and common offset compensation calculating circuits and multiplication and subtraction circuits.
5	GNCND	AD Noise Elimination Disable Disables the AD noise eliminator. 0: Noise eliminator is enabled (default) 1: Noise eliminator is disabled
4	SGLMD	AD Stucking Detection Circuit Disable Disables the AD stucking detector. 0: Stucking detector is enabled (default) 1: Stucking detector is disabled (default)

Table 26.24 RDC3AnSCCOR1 Register Contents (3/3)

Bit Position	Bit Name	Description
3	GNJSP	Noise Elimination Comparison Sessions Select Selects the sessions of AD conversion whose results are compared. 0: The results of AD conversion from the current session and previous sessions are compared. 1: The results of AD conversion from the current session are compared with those from the sessions immediately before the previous session.
2 to 0	GNCNS[2:0]	AD Noise Elimination Setting Sets the threshold for judging the value of AD conversion to be noise. If the value obtained as the result of AD conversion exceeds the value of these bits, it is rejected. b2 b1 b0 Threshold for noise 0 0 0: 128(approx. 156 mV) (default) 0 0 1: 32 (approx. 39 mV) 0 1 0: 64 (approx. 78 mV) 0 1 1: 128 (approx. 156 mV) 1 0 0: 256 (approx. 313 mV) 1 0 1: 512 (approx. 625 mV) 1 1 0: 1024 (approx. 1250 mV) 1 1 1: 2048 (approx. 2500 mV)

26.3.10 RDC3AnSCCOR2 ─ Sine/Cosine Correction Register 1

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0020_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	GNCNM[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PHCNM[5:0]						—	—	PHSNM[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.25 RDC3AnSCCOR2 Register Contents

Bit Position	Bit Name	Description
31 to 28	Reserved	These bits are read as 0. The write value should be 0.
27 to 16	GNCNM[11:0]	Gain Correction Fixed Value Sets the fixed value for use in gain correction. The value obtained from the calculation below is multiplied by the gain value for the cosine side. – Value for correction of the cosine side = d'(GNCNM[11:0])/1024 Set these bits to 1024 (multiplication by 1) for no correction. If 1178($49A_H$) is set , for example, the value for the cosine side is corrected by +15%.
15, 14	Reserved	These bits are read as 0. The write value should be 0.
13 to 8	PHCNM[5:0]	Cosine Side Phase Correction Fixed Value Sets the fixed value for use in correction of the phase (amount of delay) of the cosine side. The value for delay obtained from the calculation below is inserted on the cosine side. – Value for delay of the cosine side = d'(PHCNM[5:0]) × 200 ns Setting of $1F_H$ is not allowed. Allowed values are between 0 µs to 6 µs.
7, 6	Reserved	These bits are read as 0. The write value should be 0.
5 to 0	PHSNM[5:0]	Sine Side Phase Correction Fixed Value Sets the fixed value for use in correction of the phase (amount of delay) of the sine side. The value for delay obtained from the calculation below is inserted on the sine side. – Value for delay of the sine side = d'(PHSNM[5:0]) × 200 ns Setting of $1F_H$ is not allowed. Allowed values are between 0 µs to 6 µs.

26.3.11 RDC3AnSCCOR3 ─ Sine/Cosine Correction Register 2

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0024_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CCOSN[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CSOSN[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.26 RDC3AnSCCOR3 Register Contents

Bit Position	Bit Name	Description
31 to 26	Reserved	These bits are read as 0. The write value should be 0.
25 to 16	CCOSN[9:0]	Cosine Side Common Offset Correction Fixed Value Setting Sets the fixed value (two's complement) for use in correction of the cosine side value. The value obtained from the calculation below is subtracted from the result of AD conversion. – Common offset value of the cosine side = d'(CCOSN[9:0]) × RVCC/4096 (V)
15 to 10	Reserved	These bits are read as 0. The write value should be 0.
9 to 0	CSOSN[9:0]	Sine Side Common Offset Correction Fixed Value Setting Sets the fixed value (two's complement) for use in correction of the sine side value. The value obtained from the calculation below is subtracted from the result of AD conversion. – Common offset value of the sine side = d'(CSOSN[9:0]) × RVCC/4096 (V)

26.3.12 RDC3AnATMNT0 ─ Automatic Amplitude Adjustment Register 0

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0028_H

Value after reset XX24 0200_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	IRSS0	IGRT	IGRM[3:0]				—	—	EXOC[1:0]		IRSC[3:0]			
Value after reset	0	0	0	-	-	-	-	-	0	0	1	0	0	1	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	EAAOD	IRSS1	EXOS	—	—	EAATSP	SQJGT	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Table 26.27 RDC3AnATMNT0 Register Contents (1/3)

Bit Position	Bit Name	Description
31, 30	Reserved	These bits are read as 0. The write value should be 0.
29	IRSS0	Input Gain Resistance Value Flash Trimming Value Invalidate Selects the value for use in the input gain resistance to be output from the excitation amplitude automatic adjustment circuit as any among the following: input gain resistance automatic adjustment value, the value set in the relevant register, or the value for trimming set in the flash memory. The selection is made by the combination of the values of this bit, the IRSS1 bit, and the IGRT signal from the flash memory.
28	IGRT	Flash Input Gain Resistance Value Invalid Confirmation This bit is for conforming the level of the input signal (IGRT) from the flash memory, which invalidates the trimming value for the input gain resistance. 0: The trimming value set in the flash memory is valid. 1: The trimming value set in the flash memory is invalid.
27 to 24	IGRM[3:0]	Flash Input Gain Resistance Value Confirmation These bits reflect the levels of the input signal from the flash memory which hold the value for the input gain resistance (IGRM[3:0]). b27 b26 b25 b24 0 0 0 0: input gain resistance value TYP – 40% 0 0 0 1: input gain resistance value TYP – 30% 0 0 1 0: input gain resistance value TYP – 20% 0 0 1 1: input gain resistance value TYP – 10% 0 1 0 0: input gain resistance value TYP± 0% 0 1 0 1: input gain resistance value TYP + 10% 0 1 1 0: input gain resistance value TYP + 20% 0 1 1 1: input gain resistance value TYP + 30% 1 0 0 0: input gain resistance value TYP + 40% 1 0 0 1: input gain resistance value TYP + 40% 1 0 1 0: input gain resistance value TYP + 40% 1 0 1 1: input gain resistance value TYP + 40% 1 1 0 0: input gain resistance value TYP + 40% 1 1 0 1: input gain resistance value TYP + 40% 1 1 1 0: input gain resistance value TYP + 40% 1 1 1 1: input gain resistance value TYP + 40%

Table 26.27 RDC3AnATMNT0 Register Contents (2/3)

Bit Position	Bit Name	Description
23, 22	Reserved	These bits are read as 0. The write value should be 0.
21, 20	EXOC[1:0]	Excitation Output Value Setting These bits make settings of the excitation output. The value in these bits are output from the excitation amplitude automatic adjustment circuit as the excitation output amplitude setting when the EXOS bit is 0. b21 b20 0 0: excitation output value TYP – 40% 0 1: excitation output value TYP – 20% 1 0: excitation output value TYP± 0% (default) 1 1: excitation output value TYP + 20%
19 to 16	IRSC[3:0]	Input Gain Resistance Value Select These bits specify the input gain resistance value. The value in these bits are output from the excitation amplitude automatic adjustment circuit as the input gain resistance value setting when the IRSS1 bit is 0 and the IRSS0 bit is 1, or when the IRSS1 bit is 0 and the signal from the flash memory is 1. b19 b18 b17 b16 0 0 0 0 : input gain resistance value TYP – 40% 0 0 0 1 : input gain resistance value TYP – 30% 0 0 1 0 : input gain resistance value TYP – 20% 0 0 1 1 : input gain resistance value TYP – 10% 0 1 0 0 : input gain resistance value TYP ± 0% (default) 0 1 0 1 : input gain resistance value TYP + 10% 0 1 1 0 : input gain resistance value TYP + 20% 0 1 1 1 : input gain resistance value TYP + 30% 1 0 0 0 : input gain resistance value TYP + 40% 1 0 0 1 : input gain resistance value TYP + 40% 1 0 1 0 : input gain resistance value TYP + 40% 1 0 1 1 : input gain resistance value TYP + 40% 1 1 0 0 : input gain resistance value TYP + 40% 1 1 0 1 : input gain resistance value TYP + 40% 1 1 1 0 : input gain resistance value TYP + 40% 1 1 1 1 : input gain resistance value TYP + 40%
15	Reserved	This bit is read as 0. The write value should be 0.
14	EAAOD	Excitation Amplitude Automatic Adjustment Order Select Selects which amplitude adjustment factor to apply first to the excitation amplitude automatic adjustment circuit as correction by the input gain resistance value or by the excitation output. 0: Correction by the excitation output and then the input gain resistance. 1: Correction by the input gain resistance value and then the excitation output.
13	IRSS1	Input Gain Resistance Value Adjustment Output Select Selects the value for use in the input gain resistance to be output from the excitation amplitude automatic adjustment circuit as any among the following: input gain resistance automatic adjustment value, the value set in the relevant register, or the value for trimming set in the flash memory. The selection is made by the combination of the values of this bit, the IRSS0 bit, and the IGRT signal from the flash memory.
12	EXOS	Excitation Output Amplitude Setting Select Selects the value for use in the excitation amplitude to be output from the excitation amplitude automatic adjustment circuit as either of the following: the value set in the relevant register or the value set for excitation output automatic adjustment. 0: The value set in the EXOC[1:0] bits. 1: The value set for the excitation output automatic adjustment.
11, 10	Reserved	These bits are read as 0. The write value should be 0.

Table 26.27 RDC3AnATMNT0 Register Contents (3/3)

Bit Position	Bit Name	Description
9	EAATSP	Excitation Amplitude Automatic Adjustment Stop Stops excitation amplitude adjustment by the excitation amplitude automatic adjustment circuit. When this bit is changed from 0 to 1, the adjustment value at that time is retained. 0: Performs automatic adjustment. 1: Stops automatic adjustment.
8	SQJGT	Excitation Amplitude Sum-of-Squares Judgment Time Select Sets the interval for adjusting the amplitude of the excitation waveform generated by the excitation amplitude automatic adjustment circuit. 0: Adjusts the amplitude of excitation waveforms every 1 milliseconds*. 1: Adjusts the amplitude of excitation waveforms every 10 milliseconds*.
7 to 0	Reserved	These bits are read as 0. The write value should be 0.

Note: The timing value is when the CCLK is running at 40 MHz.

Selection of the Input Gain Resistance Value

One of the following three values is selected as the input gain resistance value to be output from the excitation amplitude automatic adjustment circuit.

- Automatically adjusted input gain resistance value
- The value set in the IRSC[3:0] bits
- The value for trimming from the flash memory (which is read from the IGRM[3:0] bits)

The values of the IRSS1 and IRSS0 bits, and of the IGRT signal from the flash memory, are used in combination to select the value to use for input gain resistance as shown in **Table 26.28**.

Table 26.28 Selection of Input Gain Resistance Value Adjustment Code

IRSS1 Bit	IRSS0 Bit	The IGRT Signal from the Flash Memory	Selected Setting of the Input Gain Resistance Value
0	0	0	The value for trimming from the flash memory
0	0	1	The value in the IRSC[3:0] bits
0	1	X	The value in the IRSC[3:0] bits
1	X	X	Automatically adjusted input gain resistance value

26.3.13 RDC3AnATMNT1 ─ Amplitude Automatic Adjustment Register 1

Access Readable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $002C_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ESQULL	ESQOUL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	RLT[17:4]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.29 RDC3AnATMNT1 Register Contents

Bit Position	Bit Name	Description
31 to 18	Reserved	These bits are read as 0. The write value should be 0.
17	ESQULL	Excitation Amplitude Integrated Sum-of-Squares Value Fall Below Lower Limit Determine Excitation amplitude integrated sum-of-squares value fall below lower limit determine signal is stored. 0: Excitation amplitude integrated sum-of-squares value is equal to or larger than the lower limit. 1: Excitation amplitude integrated sum-of-squares value is smaller than the lower limit.
16	ESQOUL	Excitation Amplitude Integrated Sum-of-Squares Value Exceed Upper Limit Determine Excitation amplitude integrated sum-of-squares value exceed upper limit determine signal is stored. 0: Excitation amplitude integrated sum-of-squares value is equal to or smaller than the upper limit. 1: Excitation amplitude integrated sum-of-squares value is larger than the upper limit.
15, 14	Reserved	These bits are read as 0. The write value should be 0.
13 to 0	RLT[17:4]	Excitation Amplitude Integrated Sum-of-Squares Value Upper 14 bits of the 18-bit excitation amplitude integrated sum-of-squares value is stored. The value is updated every 1 ms*.

Note: The timing value is when the CCLK is running at 40 MHz.

26.3.14 RDC3AnDIAG0 ─ Error Detection Register 0

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0030_H

Value after reset $001A\ 2933_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	P2ANT[1:0]		EXCETH[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SGBTH[7:0]								SGBDTH[7:0]							
Value after reset	0	0	1	0	1	0	0	1	0	0	1	1	0	0	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.30 RDC3AnDIAG0 Register Contents

Bit Position	Bit Name	Description
31 to 26	Reserved	These bits are read as 0. The write value should be 0.
25, 24	P2ANT [1:0]	Two Paths Conversions Error Threshold Sets the threshold for use in detecting errors by comparing the results of conversion from the path for the angular conversion mode 0 and that for the angular conversion mode 1. b25 b24 Threshold for compared values 0 0: ± 128LSB (@12-bit resolution) (default) 0 1: ± 64LSB (@12-bit resolution) 1 0: ± 256LSB (@12-bit resolution) 1 1: ± 512LSB (@12-bit resolution)
23 to 16	EXCETH[7:0]	Resolver Signal Error Comparison Threshold Sets the threshold for use in detecting errors in the resolver signal. The threshold is obtained from the formula below. – Threshold value = 2 × d'(EXCETH[7:0]) × 8 × RVCC/4096 (Vpp) Default setting $1A_H$: 0.102 × RVCC (Vpp)
15 to 8	SGBTH[7:0]	Disconnect Error Comparison Threshold (for VR Resolver) Sets the threshold for use in detecting disconnect errors when the VR resolver is used. The threshold is obtained from the formula below. – Threshold value = 0.5 × RVCC + d'(SGBTH[7:0]) × 8×RVCC/4096 (V) Default setting 29_H: 0.58 × RVCC (V)
7 to 0	SGBDTH[7:0]	Disconnect Error Comparison Threshold Setting (for DC Resolver) Sets the threshold for use in detecting disconnect errors when the DC resolver is used. The threshold is obtained from the formula below. – Threshold value = 0.5 × RVCC + (d'(SGBDTH[7:0]) × 8 + 1024) × RVCC/4096 (V) Default setting 33_H: 0.85 × RVCC (V)

26.3.15 RDC3AnDIAG1 ─ Error Detection Register 1

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0034_H

Value after reset $B000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CVEDS	EDPS[1:0]		—	—	VGASL[1:0]		—	—	—	VGST	—	—	—	INIT
Value after reset	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ERDEN	—	SQERST	—	ERRST	—	—	—	—	—	—	—	KIRST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R/W	R	R/W	R	R	R	R	R	R	R	R/W

Table 26.31 RDC3AnDIAG1 Register Contents (1/2)

Bit Position	Bit Name	Description
31	Reserved	This bit is read as 1. The write value should be 1.
30	CVEDS	Conversion Error Detection Circuit Select Selects the conversion error detection circuit. 0: The conversion error detection circuit which supports high-speed rotation of the RD conversion error detection signal.*1 1: The conversion error detection circuit which does not support high-speed rotation of the RD conversion error detection signal.
29, 28	EDPS[1:0]	RD Conversion Error Determination Time Select Selects the error determination time (for preventing sudden acceleration) for RD conversion errors*2 b29 b28 0 0: 95.8* millisecond*3 0 1: 147* millisecond*3 1 0: 4.92* millisecond 1 1: 7.37* millisecond (default)
27, 26	Reserved	These bits are read as 0. The write value should be 0.
25, 24	VGASL[1:0]	Power/Ground Short Error Detection Start Method Select Selects the timing to start detection of a power/ground short error. b25 b24 Starting method 0 0: No detection (default) 0 1: When 1 is written to the VGST bit 1 0: Every 10 ms 1 1: No detection
23 to 21	Reserved	These bits are read as 0. The write value should be 0.
20	VGST	Power/Ground Short Error Detection Start Writing 1 to this bit executes power/ground short error detection a single time over all six analog pins. This bit becomes 0 after two clock cycles have elapsed following it being set to 1.
19 to 17	Reserved	These bits are read as 0. The write value should be 0.

Table 26.31 RDC3AnDIAG1 Register Contents (2/2)

Bit Position	Bit Name	Description
16	INIT	Writing 1 to this bit leads to initialization within RDC3A. After this bit has been set to 1, its value is restored to 0 following the completion of initialization within RDC3A.*4
15 to 13	Reserved	These bits are read as 0. The write value should be 0.
12	ERDEN	Error Detection Start Error detection is enabled when the error detection output mask is released after 26 milliseconds* have elapsed following this bit being set to 1.*5
11	Reserved	This bit is read as 0. The write value should be 0.
10	SQERST	Sum-of-Squares Amplitude Error Excitation Counter Reset Writing 1 to this bit resets the sum-of-squares amplitude error excitation counter to 0. This bit becomes 0 after two clock cycles have elapsed following it being set to 1.
9	Reserved	This bit is read as 0. The write value should be 0.
8	ERRST	Error Signal Reset Bit Writing 1 to this bit resets the register bits listed below to 0. Note that each of these bits remains at 1 if an error is continuous. This bit becomes 0 after two clock cycles have elapsed following it being set to 1. Register bits: ERHD, ERDEXC, ERDSBC, ERDSBS, ERDP2, ERDCNV, ERDR1 to 4V
7 to 1	Reserved	These bits are read as 0. The write value should be 0.
0	KIRST	Ki Component Reset The values of the Ki integrator and accumulator integrator become 0 by writing 1 to this bit. On reset, the forced gain control function is enabled and the Kv gain of the PI compensator goes high. This bit becomes 0 after two clock cycles have elapsed following it being set to 1. Setting this bit to 1 while it is 1, wait for at least 2 µs.

Note: The timing value is when the CCLK is running at 40 MHz.

Note 1. Input the excitation signal whose frequency is below 22 kHz when the RD conversion error detection circuit (which supports high-speed rotation) is used (CVEDS = 0).

Note 2. Do not set EDPS[1:0] to a shorter determination time than the current setting while the R/D converter is operating. In other words, the settings below are prohibited;
When EDPS[1:0] = 01, setting EDPS[1:0] to 00, 10, or 11
When EDPS[1:0] = 00, setting EDPS[1:0] to 10 or 11
When EDPS[1:0] = 11, setting EDPS[1:0] to 10
However, changing from 11b to 10b is allowed if the ERDEN bit in the RDC3AnDIAG1 register is 0 after a reset. After changing the bits, set the ERDEN bit to 1.

Note 3. Do not set this value when the RD conversion error detection circuit (which supports high-speed rotation) is selected (CVEDS = 0).

Note 4. The user must not clear the INIT bit once it has been set.

Note 5. The functionality of error detection is disabled after a reset. It is enabled after 26 milliseconds* since the ERDEN bit has been set to 1. Enabling this functionality by setting this bit to 1 takes effects only once, following a reset (setting this bit to 0 does not disable error detection). To disable this functionality after enabling it, disable interrupts by setting the EINTEN bit to 0.

R/D Conversion Error Determination Time Select Bits

These bits are used to set the time to determine R/D conversion errors. If a deviation (ε) between the input angle and R/D converted angle remains large for more than 50% of the specified determination time, it is considered as the R/D conversion error and the ERCNV bit in the RDC3AnDGOUT0 register is set to 1 if the ERCNVS bit selects detection of a conversion error.

Furthermore, if the EINTEN bit in the RDC3AnENC0 register is 1, the ERR and ERHD bits in the RDC3AnDGOUT0 register are set to 1 and an RDC error interrupt occurs.

26.3.16 RDC3AnDIAG2 – Error Detection Register 2

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0038_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	EREXCS	ERSBCS	ERSBSS	—	ERSQS	ERP2S	ERCNVS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ERR1VS	ERR2VS	ERS1VS	ERS2VS	ERS3VS	ERS4VS	—	—	ERR1GS	ERR2GS	ERS1GS	ERS2GS	ERS3GS	ERS4GS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.32 RDC3AnDIAG2 Register Contents (1/3)

Bit Position	Bit Name	Description
31 to 23	Reserved	These bits are read as 0. The write value should be 0.
22	EREXCS	Resolver Signal Error Select Selects whether or not to output detection of a resolver signal error. 0: Set the bits ERR, ERHD, EREXC, and ERDEXC to 1 on occurrence of a resolver signal error 1: Do not set the bits ERR, ERHD, EREXC, and ERDEXC to 1 on occurrence of a resolver signal error
21	ERSBCS	Disconnect Error (Cosine) Select Selects whether or not to output detection of a disconnect error (cosine side) 0: Set the bits ERR, ERHD, ERSBC, and ERDSBC to 1 on occurrence of a disconnect error (cosine side) 1: Do not set the bits ERR, ERHD, ERSBC, and ERDSBC to 1 on occurrence of a disconnect error (cosine side).
20	ERSBSS	Disconnect Error (Sine) Select Selects whether or not to output detection of a disconnect error (sine side) 0: Set the bits ERR, ERHD, ERSBS, and ERDSBS to 1 on occurrence of a disconnect error (sine side) 1: Do not set the bits ERR, ERHD, ERSBS, and ERDSBS to 1 on occurrence of a disconnect error (sine side).
19	Reserved	This bit is read as 0. The write value should be 0.
18	ERSQS	Sum-of-Squares Amplitude Error Select Selects whether or not to output detection of a sum-of-squares amplitude error. 0: Set the bits ERR, ERHD, ERSQ, and ERDSQ to 1 on occurrence of a sum-of-squares amplitude error 1: Do not set the bits ERR, ERHD, ERSQ, and ERDSQ to 1 on occurrence of a sum-of-squares amplitude error.
17	ERP2S	Two Paths Conversion Error Select Selects whether or not to output detection of a two paths conversion error. 0: Set the bits ERR, ERHD, ERP2, and ERDP2 to 1 on occurrence of a two paths conversion error 1: Do not set the bits ERR, ERHD, ERP2, and ERDP2 to 1 on occurrence of a two paths conversion error

Table 26.32 RDC3AnDIAG2 Register Contents (2/3)

Bit Position	Bit Name	Description
16	ERCNVS	Conversion Error Select Selects whether or not to output detection of a conversion error. 0: Set the bits ERR, ERHD, ERCNV, and ERDCNV to 1 on occurrence of a conversion error 1: Do not set the bits ERR, ERHD, ERCNV, and ERDCNV to 1 on occurrence of a conversion error
15, 14	Reserved	These bits are read as 0. The write value should be 0.
13	ERR1VS	RSO Power Short Error Select Selects whether or not to output detection of an RSO pin power short error. 0: Set the bits ERR, ERHD, ERR1V, and ERDR1V to 1 on occurrence of an RSO pin power short error. 1: Do not set the bits ERR, ERHD, ERR1V, and ERDR1V to 1 on occurrence of an RSO pin power short error.
12	ERR2VS	COM Power Short Error Select Selects whether or not to output detection of a COM pin power short error. 0: Set the bits ERR, ERHD, ERR2V, and ERDR2V to 1 on occurrence of a COM pin power short error. 1: Do not set the bits ERR, ERHD, ERR2V, and ERDR2V to 1 on occurrence of a COM pin power short error.
11	ERS1VS	S1 Power Short Error Select Selects whether or not to output detection of an S1 pin power short error. 0: Set the bits ERR, ERHD, ERCNV, ERS1V, and ERDS1V to 1 on occurrence of an S1 pin power short error. 1: Do not set the bits ERR, ERHD, ERCNV, ERS1V, and ERDS1V to 1 on occurrence of an S1 pin power short error.
10	ERS2VS	S2 Power Short Error Select Selects whether or not to output detection of an S2 pin power short error. 0: Set the bits ERR, ERHD, ERCNV, ERS2V, and ERDS2V to 1 on occurrence of an S2 pin power short error. 1: Do not set the bits ERR, ERHD, ERCNV, ERS2V, and ERDS2V to 1 on occurrence of an S2 pin power short error.
9	ERS3VS	S3 Power Short Error Select Selects whether or not to output detection of an S3 pin power short error. 0: Set the bits ERR, ERHD, ERCNV, ERS3V, and ERDS3V to 1 on occurrence of an S3 pin power short error. 1: Do not set the bits ERR, ERHD, ERCNV, ERS3V, and ERDS3V to 1 on occurrence of an S3 pin power short error.
8	ERS4VS	S4 Power Short Error Select Selects whether or not to output detection of an S4 pin power short error. 0: Set the bits ERR, ERHD, ERCNV, ERS4V, and ERDS4V to 1 on occurrence of an S4 pin power short error. 1: Do not set the bits ERR, ERHD, ERCNV, ERS4V, and ERDS4V to 1 on occurrence of an S4 pin power short error.
7, 6	Reserved	These bits are read as 0. The write value should be 0.
5	ERR1GS	RSO Ground Short Error Select Selects whether or not to output detection of an RSO pin ground short error. 0: Set the bits ERR, ERHD, ERR1G, and ERDR1G to 1 on occurrence of an RSO pin ground short error. 1: Do not set the bits ERR, ERHD, ERR1G, and ERDR1G to 1 on occurrence of an RSO pin ground short error.

Table 26.32 RDC3AnDIAG2 Register Contents (3/3)

Bit Position	Bit Name	Description
4	ERR2GS	COM Ground Short Error Select Selects whether or not to output detection of a COM pin ground short error. 0: Set the bits ERR, ERHD, ERR2G, and ERDR2G to 1 on occurrence of a COM pin ground short error. 1: Do not set the bits ERR, ERHD, ERR2G, and ERDR2G to 1 on occurrence of a COM pin ground short error.
3	ERS1GS	S1 Ground Short Error Select Selects whether or not to output detection of an S1 pin ground short error. 0: Set the bits ERR, ERHD, ERCNV, ERS1G, and ERDS1G to 1 on occurrence of an S1 pin ground short error. 1: Do not set the bits ERR, ERHD, ERCNV, ERS1G, and ERDS1G to 1 on occurrence of an S1 pin ground short error.
2	ERS2GS	S2 Ground Short Error Select Selects whether or not to output detection of an S2 pin ground short error. 0: Set the bits ERR, ERHD, ERCNV, ERS2G, and ERDS2G to 1 on occurrence of an S2 pin ground short error. 1: Do not set the bits ERR, ERHD, ERCNV, ERS2G, and ERDS2G to 1 on occurrence of an S2 pin ground short error.
1	ERS3GS	S3 Ground Short Error Select Selects whether or not to output detection of an S3 pin ground short error. 0: Set the bits ERR, ERHD, ERCNV, ERS3G, and ERDS3G to 1 on occurrence of an S3 pin ground short error. 1: Do not set the bits ERR, ERHD, ERCNV, ERS3G, and ERDS3G to 1 on occurrence of an S3 pin ground short error.
0	ERS4GS	S4 Ground Short Error Select Selects whether or not to output detection of an S4 pin ground short error. 0: Set the bits ERR, ERHD, ERCNV, ERS4G, and ERDS4G to 1 on occurrence of an S4 pin ground short error. 1: Do not set the bits ERR, ERHD, ERCNV, ERS4G, and ERDS4G to 1 on occurrence of an S4 pin ground short error.

26.3.17 RDC3AnDGOUT0 ─ Error Detection Output Register 0

Access Readable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $003C_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VGFLG	—	—	ERR	—	—	—	ERHD	—	EREXC	ERSBC	ERSBS	—	ERSQ	ERP2	ERCNV
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ERR1V	ERR2V	ERS1V	ERS2V	ERS3V	ERS4V	—	—	ERR1G	ERR2G	ERS1G	ERS2G	ERS3G	ERS4G
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.33 RDC3AnDGOUT0 Register Contents (1/2)

Bit Position	Bit Name	Description
31	VGFLG	Power/Ground Short Error Detection Running This flag indicates whether or not the power/ground short error detection BIST is running or not. The BIST is running if the flag is 1.
30, 29	Reserved	These bits are read as 0. The write value should be 0.
28	ERR	Error This flag is set to 1 if the active level of the detection signal for any of the errors indicated by flags in the RDC3AnDIAG2 register is being output. This signal returns to 0 when recovered from an error. This flag is not set to 1 if the EINTEN bit of the RDC3AnENC0 register is 0.
27 to 25	Reserved	These bits are read as 0. The write value should be 0.
24	ERHD	Error Hold This flag is set to 1 if the active level of the detection signal for any of the errors indicated by flags in the RDC3AnDIAG2 register is being output. This signal is reset to 0 by setting the ERRST bit to 1. Note that this bit remains at 1 if errors occur continuously. This flag is not set to 1 if the EINTEN bit of the RDC3AnENC0 register is 0.
23	Reserved	This bit is read as 0. The write value should be 0.
22	EREXC	Resolver Signal Error The value of this flag becomes 1 when a resolver signal error occurs and returns to 0 on recovery from the error.
21	ERSBC	Disconnect Error (Cosine Side) The value of this flag becomes 1 when a disconnect error (cosine side) occurs and returns to 0 on recovery from the error.
20	ERSBS	Disconnect Error (Sine Side) The value of this flag becomes 1 when a disconnect error (sine side) occurs and returns to 0 on recovery from the error.
19	Reserved	This bit is read as 0. The write value should be 0.
18	ERSQ	Sum-of-Squares Amplitude Error The value of this flag becomes 1 when a sum-of-squares amplitude error occurs and returns to 0 on recovery from the error.

Table 26.33 RDC3AnDGOUT0 Register Contents (2/2)

Bit Position	Bit Name	Description
17	ERP2	Two Paths Conversion Error The value of this flag becomes 1 when a two path conversion error occurs and returns to 0 on recovery from the error.
16	ERCNV	Conversion Error The value of this flag becomes 1 when a conversion error occurs and returns to 0 on recovery from the error.
15, 14	Reserved	These bits are read as 0. The write value should be 0.
13	ERR1V	RSO Power Short Error The value of this flag becomes 1 when an RSO pin power short error occurs and returns to 0 on recovery from the error.
12	ERR2V	COM Power Short Error The value of this flag becomes 1 when a COM pin power short error occurs and returns to 0 on recovery from the error.
11	ERS1V	S1 Power Short Error The value of this flag becomes 1 when a S1 pin power short error occurs and returns to 0 on recovery from the error.
10	ERS2V	S2 Power Short Error The value of this flag becomes 1 when a S2 pin power short error occurs and returns to 0 on recovery from the error.
9	ERS3V	S3 Power Short Error The value of this flag becomes 1 when a S3 pin power short error occurs and returns to 0 on recovery from the error.
8	ERS4V	S4 Power Short Error The value of this flag becomes 1 when a S4 pin power short error occurs and returns to 0 on recovery from the error.
7, 6	Reserved	These bits are read as 0. The write value should be 0.
5	ERR1G	RSO Ground Short Error The value of this flag becomes 1 when an RSO pin ground short error occurs and returns to 0 on recovery from the error.
4	ERR2G	COM Ground Short Error The value of this flag becomes 1 when a COM pin ground short error occurs and returns to 0 on recovery from the error.
3	ERS1G	S1 Ground Short Error The value of this flag becomes 1 when a S1 pin ground short error occurs and returns to 0 on recovery from the error.
2	ERS2G	S2 Ground Short Error The value of this flag becomes 1 when a S2 pin ground short error occurs and returns to 0 on recovery from the error.
1	ERS3G	S3 Ground Short Error The value of this flag becomes 1 when a S3 pin ground short error occurs and returns to 0 on recovery from the error.
0	ERS4G	S4 Ground Short Error The value of this flag becomes 1 when a S4 pin ground short error occurs and returns to 0 on recovery from the error.

26.3.18 RDC3AnDGOUT1 – Error Detection Output Register 1

Access Readable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0040_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	ERR	—	—	—	ERHD	—	ERDEXC	ERDSBC	ERDSBS	—	ERDSQ	ERDP2	ERDCNV
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ERDR1V	ERDR2V	ERDS1V	ERDS2V	ERDS3V	ERDS4V	—	—	ERDR1G	ERDR2G	ERDS1G	ERDS2G	ERDS3G	ERDS4G
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.34 RDC3AnDGOUT1 Register Contents (1/2)

Bit Position	Bit Name	Description
31 to 29	Reserved	These bits are read as 0. The write value should be 0.
28	ERR	Error This flag is set to 1 if the active level of the detection signal for any of the errors indicated in the RDC3AnDIAG2 register is being output. This signal returns to 0 when recovered from an error. This flag is not set to 1 if the EINTEN bit of the RDC3AnENC0 register is 0.
27 to 25	Reserved	These bits are read as 0. The write value should be 0.
24	ERHD	Error Hold This flag is set to 1 if the active level of the detection signal for any of the errors indicated in the RDC3AnDIAG2 register is being output. This signal is reset to 0 by setting the ERRST bit to 1. Note that this bit remains at 1 if an error is continuous. This flag is not set to 1 if the EINTEN bit of the RDC3AnENC0 register is 0.
23	Reserved	This bit is read as 0. The write value should be 0.
22	ERDEXC	Resolver Signal Error Hold The value of this flag becomes 1 when a resolver signal error occurs. This bit remains at 1 until it is reset by the ERRST bit.
21	ERDSBC	Disconnect Error (Cosine Side) Hold The value of this flag becomes 1 when a disconnect error (cosine side) occurs. This bit remains at 1 until it is reset by the ERRST bit.
20	ERDSBS	Disconnect Error (Sine Side) Hold The value of this flag becomes 1 when a disconnect error (sine side) occurs. This bit remains at 1 until it is reset by the ERRST bit.
19	Reserved	This bit is read as 0. The write value should be 0.
18	ERDSQ	Sum-of-Squares Amplitude Error Hold The value of this flag becomes 1 when a sum-of-squares amplitude error occurs. This bit remains at 1 until it is reset by the ERRST bit.
17	ERDP2	Two Paths Conversion Error Hold The value of this flag becomes 1 when a two path conversion error occurs. This bit remains at 1 until it is reset by the ERRST bit.
16	ERDCNV	Conversion Error Hold The value of this flag becomes 1 when a conversion error occurs. This bit remains at 1 until it is reset by the ERRST bit.
15, 14	Reserved	These bits are read as 0. The write value should be 0.

Table 26.34 RDC3AnDGOUT1 Register Contents (2/2)

Bit Position	Bit Name	Description
13	ERDR1V	RSO Power Short Error Hold The value of this flag becomes 1 when an RSO pin power short error occurs. This bit remains at 1 until it is reset by the ERRST bit.
12	ERDR2V	COM Power Short Error Hold The value of this flag becomes 1 when a COM pin power short error occurs. This bit remains at 1 until it is reset by the ERRST bit.
11	ERDS1V	S1 Power Short Error Hold The value of this flag becomes 1 when a S1 pin power short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
10	ERDS2V	S2 Power Short Error Hold The value of this flag becomes 1 when a S2 pin power short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
9	ERDS3V	S3 Power Short Error Hold The value of this flag becomes 1 when a S3 pin power short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
8	ERDS4V	S4 Power Short Error Hold The value of this flag becomes 1 when a S4 pin power short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
7, 6	Reserved	These bits are read as 0. The write value should be 0.
5	ERDR1G	RSO Ground Short Error Hold The value of this flag becomes 1 when an RSO pin ground short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
4	ERDR2G	COM Ground Short Error Hold The value of this flag becomes 1 when a COM pin ground short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
3	ERDS1G	S1 Ground Short Error Hold The value of this flag becomes 1 when a S1 pin ground short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
2	ERDS2G	S2 Ground Short Error Hold The value of this flag becomes 1 when a S2 pin ground short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
1	ERDS3G	S3 Ground Short Error Hold The value of this flag becomes 1 when a S3 pin ground short error occurred. This bit remains at 1 until it is reset by the ERRST bit.
0	ERDS4G	S4 Ground Short Error Hold The value of this flag becomes 1 when a S4 pin ground short error occurred. This bit remains at 1 until it is reset by the ERRST bit.

26.3.19 RDC3AnBIST0 ─ BIST Register 0

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0044_H

Value after reset $0200\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	ADB3TH[1:0]		—	—	—	ERCVP2D	—	—	—	CBSP2D
Value after reset	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	BSTF	—	—	—	—	BISTCD[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.35 RDC3AnBIST0 Register Contents

Bit Position	Bit Name	Description
31 to 26	Reserved	These bits are read as 0. The write value should be 0.
25, 24	ADB3TH[1:0]	ADBIST Threshold Setting Sets the threshold for ADBIST determination. b25 b24 Threshold for determination 0 0: ± 16LSB 0 1: ± 8LSB 1 0: ± 32LSB (default) 1 1: ± 64LSB
23 to 21	Reserved	These bits are read as 0. The write value should be 0.
20	ERCVP2D	Two Paths Conversion Error BIST Disable Selects whether or not to include two paths conversion error in the conversion error BIST. 0: Two paths conversion error is included in the conversion error BIST (default). 1: Two paths conversion error is not included in the conversion error BIST
19 to 17	Reserved	These bits are read as 0. The write value should be 0.
16	CBSP2D	Two Paths Conversion BIST Disable Selects whether or not to include two paths conversion in the conversion BIST. 0: Two paths conversion is included in the conversion BIST (default). 1: Two paths conversion is not included in the conversion BIST
15 to 9	Reserved	These bits are read as 0. The write value should be 0.
8	BSTF	BIST Flag This flag indicates whether a BIST is running or not. A BIST is running if the flag is 1 and another BIST cannot be executed while the flag remains 1.
7 to 4	Reserved	These bits are read as 0. The write value should be 0.
3 to 0	BISTCD[3:0]	BIST Result Store The result of BIST is stored in these bits (see **Table 26.36**).

BIST Result Store Bits

The result of BIST executed by the BCON [3:0] bits is stored in the BISTCD [3:0] bits. The relation between the BCON[3:0] and BISTCD[3:0] are shown in **Table 26.36**.

Table 26.36 Contents of BCON[3:0] and BISTCD[3:0]

Bits for Selecting the BIST to be Executed BCON[3:0]				Bits where the result of BIST is Stored BISTCD[3:0]				
b3	b2	b1	b0	b7	b6	b5	b4	Content indicated by BISTCD[3:0]
0	0	0	0	0	0	0	0	BEXE is disabled. BISTCD[3:0] = 0000 is stored without any determination.
0	0	0	1	X	X	X	X	This combination is not allowed.
0	0	1	0	0	0	1	0	The result of sum-of-squares amplitude error detection BIST (low side) was passed.
				1	1	1	1	The result of sum-of-squares amplitude error detection BIST (low side) was failure.
0	0	1	1	0	0	1	1	The result of sum-of-squares amplitude error detection BIST (high side) was passed.
				1	1	1	1	The result of sum-of-squares amplitude error detection BIST (high side) was failure.
0	1	0	0	0	1	0	0	The result of the ADBIST was passed.
				1	1	1	1	The result of the ADBIST was failure.
0	1	0	1	0	1	0	1	The result of the angle conversion BIST1 (0°) was passed.
				1	1	1	1	The result of the angle conversion BIST1 (0°) was failure.
0	1	1	0	0	1	1	0	The result of the angle conversion BIST2 (45°) was passed.
				1	1	1	1	The result of the angle conversion BIST2 (45°) was failure.
0	1	1	1	0	1	1	1	The result of the angle conversion BIST3 (270°) was passed.
				1	1	1	1	The result of the angle conversion BIST3 (270°) was failure.
1	0	0	0	X	X	X	X	This combination is not allowed.
1	0	0	1	1	0	0	1	The result of the resolver signal error detection BIST was passed.
				1	1	1	1	The result of the resolver signal error detection BIST was failure.
1	0	1	0	1	0	1	0	The result of the resolver signal disconnect detection BIST (cosine side) was passed.
				1	1	1	1	The result of the resolver signal disconnect detection BIST (cosine side) was failure.
1	0	1	1	1	0	1	1	The result of the resolver signal disconnect detection BIST (sine side) was passed.
				1	1	1	1	The result of the resolver signal disconnect detection BIST (sine side) was failure.
1	1	0	0	1	1	0	0	The result of the conversion error BIST was passed.
				1	1	1	1	The result of the conversion error BIST was failure.
1	1	0	1	1	1	0	1	The result of the power short error BIST was passed.
				1	1	1	1	The result of the power short error BIST was failure.
1	1	1	0	1	1	1	0	The result of the ground short error BIST was passed.
				1	1	1	1	The result of the ground short error BIST was failure.
1	1	1	1	X	X	X	X	This combination is not allowed.

26.3.20 RDC3AnBIST1 – BIST Register 1

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0048_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BISTCL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	BEXE	—	—	—	—	BCON[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 26.37 RDC3AnBIST1 Register Contents

Bit Position	Bit Name	Description
31 to 17	Reserved	These bits are read as 0. The write value should be 0.
16	BISTCL	BIST Result Clear Writing 1 to this bit clears the result of the BIST (BISTCD[3:0]) to 0. This bit is returned to 0 after two clock cycles have elapsed following it being set to 1.
15 to 9	Reserved	These bits are read as 0. The write value should be 0.
8	BEXE	BIST Execution This signal is used for starting a BIST. Setting this bit to 1 leads to execution of a BIST. This bit is returned to 0 after two clock cycles have elapsed after having been set to 1. Select the BIST to be executed by the combination of values in BCON[3:0] (see **Table 26.38**).
7 to 4	Reserved	These bits are read as 0. The write value should be 0.
3 to 0	BCON[3:0]	Executing BIST Setting Selects the BIST to be executed (see **Table 26.38**).

Executing BIST Setting

The BIST to be executed is selected by the combination of values in the BCON[3:0]. The combination of values in BCON[3:0] and corresponding BIST to be executed are shown in **Table 26.38**.

Table 26.38 BIST Selected by the Combination of Values in BCON[3:0]

b3	b2	b1	b0	BIST to be Executed
0	0	0	0	BEXE is disabled.
0	0	0	1	This combination is not allowed.
0	0	1	0	Error detection BIST: sum-of-squares amplitude error detection BIST (low side)
0	0	1	1	Error detection BIST: sum-of-squares amplitude error detection BIST (high side)
0	1	0	0	ADBIST
0	1	0	1	Angle conversion BIST: target angle 1 (0°)
0	1	1	0	Angle conversion BIST: target angle 2 (45°)
0	1	1	1	Angle conversion BIST: target angle 3 (270°)
1	0	0	0	This combination is not allowed.
1	0	0	1	Error detection BIST: resolver signal error detection BIST
1	0	1	0	Error detection BIST: resolver signal disconnect detection BIST (cosine side)
1	0	1	1	Error detection BIST: resolver signal disconnect detection BIST (sine side)
1	1	0	0	Error detection BIST: conversion error BIST
1	1	0	1	Error detection BIST: power short error BIST
1	1	1	0	Error detection BIST: ground short error BIST
1	1	1	1	This combination is not allowed.

26.3.21 RDC3AnREF ─ Excitation Setting Register

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $004C_H$

Value after reset $0A0F\ 0400_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EXFS2[3:0]				RFEXS	—	SENS	EXIO	—	—	EXF15	EXFS[4:0]				
Value after reset	0	0	0	0	1	0	1	0	0	0	0	0	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	COMSTS[2:0]			—	—	—	—	—	—	—	PLSNFS	—	—	—	—
Value after reset	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W	R	R	R	R

Table 26.39 RDC3AnREF Register Contents (1/2)

Bit Position	Bit Name	Description
31 to 28	EXFS2[3:0]	Excitation Frequency Select 2 The frequency of the excitation signal is determined by the combination of values in the EXFS2 and EXFS bits. Note that the frequency is fixed to 15 kHz when EXF15 = 1. See **Table 26.40** for the frequencies to be output*2.
27	RFEXS	Excitation Component Extraction Function 0: Excitation component extraction function is disabled. 1: Excitation component extraction function is enabled (default)
26	Reserved	This bit is read as 0. The write value should be 0.
25	SENS	Required Sensor Selection Function Selects the required sensor. 0: Use the DC resolver*1 1: Use the VR resolver (default)
24	EXIO	RSO/COM Input/Output Switching Switches input and output through the RSO and COM pins. 0: Excitation signal input from outside (default) 1: Excitation voltage output
23, 22	Reserved	These bits are read as 0. The write value should be 0.
21	EXF15	Excitation Signal 15 kHz Setting Selects whether to set the frequency of the excitation signal to 15 kHz or not. 0: The frequency of the excitation signal is the value set by the EXFS[4:0] bits. 1: The frequency of the excitation signal is 15 kHz.
20 to 16	EXFS[4:0]	Excitation Signal Frequency Select The frequency of the excitation signal is determined by the combination of values in the EXFS2 and EXFS bits. Note that the frequency is fixed to 15 kHz when EXF15 = 1. See **Table 26.40** for the frequencies to be output*2.
15	Reserved	This bit is read as 0. The write value should be 0.

Table 26.39 RDC3AnREF Register Contents (2/2)

Bit Position	Bit Name	Description
14 to 12	COMSTS[2:0]	AD Comparator Stop Function Select Selects how stopping the AD comparator at the time of switching of the excitation buffer code. b14 b13 b12 0 0 0: Does not stop. 0 0 1: Stops when all of the DA[6:0] bits are changed. 0 1 0: Stops when the DA[6:3] bits are changed. 0 1 1: Stops when the DA[6:4] bits are changed. 1 0 0: Stops when the DA[6:5] bits are changed. 1 0 1: Stops when the DA[6] bit is changed. 1 1 0: Does not stop. 1 1 1: Does not stop. DA: excitation sine wave control signal*3
11	Reserved	This bit is read as 0. The write value should be 0.
10	Reserved	This bit is read as 1. The write value should be 1.
9 to 5	Reserved	These bits are read as 0. The write value should be 0.
4	PLSNFS	Excitation Extraction Noise Filter Selects the noise filter for the excitation component extraction circuit. b4 0: The noise filter is not used. 1: The noise filter is used.
3 to 0	Reserved	This bit is read as 0. The write value should be 0.

Note 1. DC resolver is selected when EXIO = 1 and SENS = 0.

Note 2. When the frequency of the excitation signal is 22 kHz or above, do not set the CVEDS bit to 0. When the user sets the frequency by using the EXFS[4:0] and EXFS2[3:0] bits, set the EXF15 bit to 0.

Note 3. When the frequency of the excitation signal is set as 20 kHz or more (the values within the thick frame in **Table 26.40)**, set COMSTS[2:0] = 000 (AD comparatore does not stop).

Table 26.40 Excitation Frequency Setting

EXFS [4:0] \ EXFS2 [3:0]	Frequency of the Excitation Signal Output (kHz)															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	Setting prohibited															
1																
2																
3	40.00	20.00	26.67	30.00	32.00	33.33	34.29	35.00	35.56	36.00	36.36	36.37	36.92	37.14	37.33	37.50
4	32.00	16.00	21.33	24.00	25.60	26.67	27.43	28.00	28.44	28.80	29.09	29.33	29.54	29.71	29.87	30.00
5	26.67	13.33	17.78	20.00	21.33	22.22	22.86	23.33	23.70	24.00	24.24	24.44	24.62	24.76	24.89	25.00
6	22.86	11.43	15.24	17.14	18.29	19.05	19.59	20.00	20.32	20.57	20.78	20.95	21.10	21.22	21.33	21.43
7	20.00	10.00	13.33	15.00	16.00	16.67	17.14	17.50	17.78	18.00	18.18	18.33	18.46	18.57	18.67	18.75
8	17.78	8.89	11.85	13.33	14.22	14.81	15.24	15.56	15.80	16.00	16.16	16.30	16.41	16.51	16.59	16.67
9	16.00	8.00	10.67	12.00	12.80	13.33	13.71	14.00	14.22	14.40	14.55	14.67	14.77	14.86	14.93	15.00
10	14.55	7.27	9.70	10.91	11.64	12.12	12.47	12.73	12.93	13.09	13.22	13.33	13.43	13.51	13.58	13.64
11	13.33	6.67	8.89	10.00	10.67	11.11	11.43	11.67	11.85	12.00	12.12	12.22	12.31	12.38	12.44	12.50
12	12.31	6.15	8.21	9.23	9.85	10.26	10.55	10.77	10.94	11.08	11.19	11.28	11.36	11.43	11.49	11.54
13	11.43	5.71	7.62	8.57	9.14	9.52	9.80	10.00	10.16	10.29	10.39	10.48	10.55	10.61	10.67	10.71
14	10.67	5.33	7.11	8.00	8.53	8.89	9.14	9.33	9.48	9.60	9.70	9.78	9.85	9.90	9.96	10.00
15	10.00	5.00	6.67	7.50	8.00	8.33	8.57	8.75	8.89	9.00	9.09	9.17	9.23	9.29	9.33	9.38
16	9.41	4.71	6.27	7.06	7.53	7.84	8.07	8.24	8.37	8.47	8.56	8.63	8.69	8.74	8.78	8.82
17	8.89	4.44	5.93	6.67	7.11	7.41	7.62	7.78	7.90	8.00	8.08	8.15	8.21	8.25	8.30	8.33
18	8.42	4.21	5.61	6.32	6.74	7.02	7.22	7.37	7.49	7.58	7.66	7.72	7.77	7.82	7.86	7.89
19	8.00	4.00	5.33	6.00	6.40	6.67	6.86	7.00	7.11	7.20	7.27	7.33	7.38	7.43	7.47	7.50
20	7.62	3.81	5.08	5.71	6.10	6.35	6.53	6.67	6.77	6.86	6.93	6.98	7.03	7.07	7.11	7.14
21	7.27	3.64	4.85	5.45	5.82	6.06	6.23	6.36	6.46	6.55	6.61	6.67	6.71	6.75	6.79	6.82
22	6.96	3.48	4.64	5.22	5.57	5.80	5.96	6.09	6.18	6.26	6.32	6.38	6.42	6.46	6.49	6.52
23	6.67	3.33	4.44	5.00	5.33	5.56	5.71	5.83	5.93	6.00	6.06	6.11	6.15	6.19	6.22	6.25
24	6.40	3.20	4.27	4.80	5.12	5.33	5.49	5.60	5.69	5.76	5.82	5.87	5.91	5.94	5.97	6.00
25	6.15	3.08	4.10	4.62	4.92	5.13	5.27	5.38	5.47	5.54	5.59	5.64	5.68	5.71	5.74	5.77
26	5.93	2.96	3.95	4.44	4.74	4.94	5.08	5.19	5.27	5.33	5.39	5.43	5.47	5.50	5.53	5.56
27	5.71	2.86	3.81	4.29	4.57	4.76	4.90	5.00	5.08	5.14	5.19	5.24	5.27	5.31	5.33	5.36
28	5.52	2.76	3.68	4.14	4.41	4.60	4.73	4.83	4.90	4.97	5.02	5.06	5.09	5.12	5.15	5.17
29	5.33	Prohibeted Setting														
30	5.16															
31	5.00															

26.3.22 RDC3AnENC0 ─ Encoder Register 0

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0050_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	XUVW[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HYSS	—	—	—	CINTEN	ABEN	UVWEN	ZEN	EINTEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 26.41 RDC3AnENC0 Register Contents (1/2)

Bit Position	Bit Name	Description
31 to 20	Reserved	These bits are read as 0. The write value should be 0.
19 to 16	XUVW	Encoder Pulse Pole Selection Selects the number of poles for the output of the encoder pulses. b19-b16 U phase V phase W phase 1111: ×1 ×1 ×1 1110: ×1 ×1 ×1 1101: ×1 ×1 ×1 1100: ×1 ×1 ×1 1011: ×1 ×1 ×1 1010: ×10 ×1 ×1 1001: ×9 ×1 ×1 1000: ×8 ×1 ×1 0111: ×1 ×1 ×1 0110: ×6 ×1 ×1 0101: ×5 ×1 ×1 0100: ×4 ×4 ×4 0011: ×3 ×3 ×3 0010: ×2 ×2 ×2 0001: ×1 ×1 ×1 0000: ×1 ×1 ×1
15 to 9	Reserved	These bits are read as 0. The write value should be 0.
8	HYSS	Hysteresis Output Select Selects whether to output the encoder pulse signal and compare flag signal through the hysteresis circuit or without going through it.*1 0: Outputs the encoder pulse signal or compare flag signal through the hysteresis circuit. 1: Outputs the encoder pulse signal or compare flag signal without going through the hysteresis circuit.
7 to 5	Reserved	These bits are read as 0. The write value should be 0.
4	CINTEN	Angle Compare Interrupt Enable 0: Interrupt is disabled. 1: Interrupt is enabled.

Table 26.41 RDC3AnENC0 Register Contents (2/2)

Bit Position	Bit Name	Description
3	ABEN	A/B Phase Output Enable 0: Output is disabled. 1: Output is enabled.
2	UVWEN	Encoder Pulse U/V/W Phase Output Enable Controls output of the U, V, and W phases of the encoder pulse signal. 0: Output of the encoder pulse signals as the external output signals RD0_OUT_U, RD0_OUT_V, or RD0_OUT_W is disabled and they are fixed to 0. 1: Output of the encoder pulse signals as the external output signals RD0_OUT_U, RD0_OUT_V, or RD0_OUT_W is enabled.
1	ZEN	Z Phase Output and Z Phase Signal Interrupt Enable 0: Output is disabled. 1: Output is enabled.
0	EINTEN	RDC Error Interrupt Enable 0: Interrupt is disabled. 1: Interrupt is enabled.

Note 1. When output through the hysteresis circuit is selected (HYSS = 0), the RD angle conversion resolution should be 12 bits (MAXV = 001).

26.3.23 RDC3AnENC1 ─ Encoder Register 1

Access Readable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0054_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PHI[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	ENCA	ENCB	ENCZ	ENCU	ENCV	ENCW
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.42 RDC3AnENC1 Register Contents

Bit Position	Bit Name	Description
31 to 16	PHI[15:0]	PHI[15:0] The digitally output PHI angle is stored in these bits with a 16-bit width. PHI[15](MSB) = 180°, PHI[0] (LSB)
15 to 6	Reserved	These bits are read as 0. The write value should be 0.
5	ENCA	Encoder Pulse A Phase This bit indicates the current state of the A-phase encoder pulses.
4	ENCB	Encoder Pulse B Phase This bit indicates the current state of the B-phase encoder pulses.
3	ENCZ	Encoder Pulse Z Phase This bit indicates the current state of the Z-phase encoder pulses.
2	ENCU	Encoder Pulse U Phase This bit indicates the current state of the U-phase encoder pulses.
1	ENCV	Encoder Pulse V Phase This bit indicates the current state of the V-phase encoder pulses.
0	ENCW	Encoder Pulse W Phase This bit indicates the current state of the W-phase encoder pulses.

26.3.24 RDC3AnENC2 – Encoder Register 2

Access Readable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0058_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PHIAD0[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.43 RDC3AnENC2 Register Contents

Bit Position	Bit Name	Description
31 to 16	Reserved	The read value is undefined. The write value should be 0.
15 to 0	PHIAD0[15:0]	Angular Conversion Mode 1 Output PHI[15:0] The digitally output PHI angle from angular conversion mode 1 is stored in these bits with a 16-bit width. PHIAD0[15] (MSB) =180°, PHIAD0[0] (LSB)

26.3.25 RDC3AnOMG – Angular Velocity Register

Access Readable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $005C_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OMG[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OMG[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.44 RDC3AnOMG Register Contents

Bit Position	Bit Name	Description
31 to 0	OMG[31:0]	Angle Velocity[31:0] Indicates the amount of change in phi within the measuring period (selected by OMGPTC[1:0]). The resolution is 25-bits. • The notation is two's complement [0] (the LSB) corresponds to $0.07min^{-1}$ and [24] corresponds to $1171875min^{-1}$. The value in [31:25] represents the sign.

26.3.26 RDC3AnTBUS ─ Test Bus Register

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0064_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	OMGPTC[1:0]		—	—	—	—	—	—	DATSEL[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.45 RDC3AnTBUS Register Contents

Bit Position	Bit Name	Description
31, 30	Reserved	These bits are read as 0. The write value should be 0.
29, 28	OMGPTC[1:0]	Angular Velocity Measuring Period Select Selects the period for measuring the amount of change in angle by the angular velocity measuring circuit. b29 b28 0 0: 12.8 µs (default) 0 1: 51.2 µs 1 0: 102.4 µs 1 1: 204.8 µs
27 to 22	Reserved	These bits are read as 0. The write value should be 0.
21 to 16	DATSEL[5:0]	RDC Data Select Outputs the RDC data to DATA[15:0] (see **Table 26.46**).
15 to 0	DATA[15:0]	Data The data selected by the DATSEL[5:0] bits of the RDC3AnTBUS register are stored in these bits.

RDC Data Select Bits

Table 26.46 Data Selection

DATASEL[5:0]	Output Signal	Output Destination DATA[X:X]
11.1011	12-bit AD output code [11:0]	[11:0]
10.1111	Control variation value [7:0] in angular conversion mode 1	[7:0]
00.0101	Control variation value [7:0] in angular conversion mode 0	[7:0]

The selected data are read from the DATA[15:0] bits.

26.3.27 RDC3AnADRD – Angular Conversion Mode Select Register

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0068_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADRD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 26.47 RDC3AnADRD Register Contents

Bit Position	Bit Name	Description
31 to 1	Reserved	These bits are read as 0. The write value should be 0.
0	ADRD	Angular Conversion Mode Select Selects the angular conversion mode. 0: Angle conversion using the angular conversion mode 0. 1: Angle conversion using the angular conversion mode 1. The ADC is operating for error detection and amplitude automatic adjustment even though the angular conversion mode 0 is selected. On the other hand, the multiplier, subtractor, and comparator of the analog circuit are stopped when the angular conversion mode 1 is selected.

26.3.28 RDC3AnETEN ─ ET Control Register

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $006C_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	ZCSTRG	—	—	CMPEN	IREN	DREN	ADTEN	ZCES	CNTEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNT[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.48 RDC3AnETEN Register Contents (1/2)

Bit Position	Bit Name	Description
31 to 25	Reserved	These bits are read as 0. The write value should be 0.
24	ZCSTRG	Software Trigger This is the software trigger for output of the zero-crossing signal from the excitation timer. 0: No operation. 1: Outputs a trigger. **CAUTION:** Writing 1 to this bit outputs a pulse as trigger signal for one cycle. Reading this bit always returns 0.
23 to 22	Reserved	These bits are read as 0. The write value should be 0.
21	CMPEN	Compare Match Function Enable Enables and disables the compare match function. 0: Disables the compare match function. 1: Enables the compare match function.
20	IREN	Interrupt Request Enable Enables and disables interrupt requests. 0: Disables the output of interrupt. 1: Enables the output of interrupt.
19	DREN	DMA Request Enable Enables and disables DMA requests. 0: Disables the DMA requests. 1: Enables the DMA requests.
18	ADTEN	A/D Conversion Start Trigger Enable Enables and disables the AD conversion start trigger. 0: Disables the A/D conversion start trigger. 1: Enables the A/D conversion start trigger.
17	ZCES	Zero-Crossing Signal Edge Select Selects the edge on the zero-crossing signal to be detected. 0: Detects the rising edge on the zero-crossing signal. 1: Detects the falling edge on the zero-crossing signal.

Table 26.48 RDC3AnETEN Register Contents (2/2)

Bit Position	Bit Name	Description
16	CNTEN	Counter Operation Enable This is the count enable signal for the period measurement timer and the event timer. 0: Operation of the period measurement timer and the event timer is stopped. The value of the reload register is read into the ET counter. 1: The period measurement timer and the event timer are operated.
15 to 0	CNT[15:0]	ET Counter Register The data from the ET counter is stored in these bits.

26.3.29 RDC3AnETCAP – ET Capture Register

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0070_H

Value after reset $0000\ FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMP[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.49 RDC3AnETCAP Register Contents

Bit Position	Bit Name	Description
31 to 16	CAP[15:0]	ET Capture [1] This register captures the data from the ET zero-crossing period measurement counter on detection of a zero-crossing signal. [2] This register captures the data from the ET zero-crossing period measurement counter on a match of the values in the counter and that in the compare match register, when compare match is enabled. [3] This register holds the value captured the last time in the conditions other than above.
15 to 0	CMP[15:0]	ET Compare [1] This register captures the value from the ET zero-crossing period measurement counter into CAP[15:0] on a match of the values in the counter and that in this register, and initializes the counter to 0000_H. [2] An excitation timer interrupt request is issued in the cycle after one in which the value in the ET zero-crossing period measurement counter matches that of this register.

26.3.30 RDC3AnETMCNT ─ ET Zero-Crossing Counter Register

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0074_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNT[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RLD[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.50 RDC3AnETMCNT Register Contents

Bit Position	Bit Name	Description
31 to 16	CNT[15:0]	Zero-Crossing Period Measurement Counter The value in the zero-crossing period measurement counter is stored in these bits.
15 to 0	RLD[15:0]	ET Reload The initial value for the ET counter is set in these bits in 16 bits. The value should be 0002_H or more (setting of 0000_H and 0001_H are prohibited).

26.3.31 RDC3AnDCUR0 – Digital Operation Register 0

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $007C_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PGAIVSL[1:0]		—	—	—	—	SYNCSL	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R	R	R	R	R

Table 26.51 RDC3AnDCUR0 Register Contents

Bit Position	Bit Name	Description
31 to 12	Reserved	These bits are read as 0. The write value should be 0.
11, 10	PGAIVSL[1:0]	PGA Inversion Function Setting Selects the timing of PGA inversion.*1 b11 b10 Inversion timing 0 0: No PGA inversion (default) 0 1: When an excitation zero-crossing occurs in response to the phi angle output exceeding the threshold value for switching to inversion. 1 0: When the AD input signal is switches between sine and cosine waves in response to the phi angle output exceeding the threshold value for switching to inversion. 1 1: When an excitation zero-crossing occurs in response to the phi angle output exceeding the threshold value for switching to inversion and the AD input signal is then switched between sine and cosine waves. In angular conversion mode 0 (ADRD = 0), set these bits to 00. To further improve the accuracy of angular conversion in angular conversion mode 1 (ADRD = 1), set these bits to a value other than 00.
9 to 6	Reserved	These bits are read as 0. The write value should be 0.
5	SYNCSL	Synchronous Detection Setting in Angular Conversion Mode 1 (ADRD = 1) b5 0: Synchronous detection setting 0 1: Synchronous detection setting 1
4 to 0	Reserved	These bits are read as 0. The write value should be 0.

Note 1. When writing 1 to the KIRST bit, set the PGAIVSL bit to 00 in advance.
After 1 has been written to the KIRST bit, wait for at least the settling time before setting the PGAIVSL bits to the desired value.

26.3.32 RDC3AnDCUR1 – Digital Operation Register 1

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0080_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DCCRSTP	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.52 RDC3AnDCUR1 Register Contents

Bit Position	Bit Name	Description
31 to 14	Reserved	These bits are read as 0. The write value should be 0.
13	DCCRSTP	DC Error Correction Setting in Angular Conversion Mode 1 (ADRD = 1) b13 0: DC correction enabled 1: DC correction disabled Be sure to set this bit to 1 when the DC resolver is used.
12 to 0	Reserved	These bits are read as 0. The write value should be 0.

26.3.33 RDC3AnBISTFX0 ─ Setting Register 0 after BIST Ends

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0088_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	BISTFX0	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.53 RDC3AnBISTFX0 Register Contents

Bit Position	Bit Name	Description
31 to 19	Reserved	These bits are read as 0. The write value should be 0.
18	BISTFX0	Setting after Execution of the Conversion BIST in Angular Conversion Mode 1 Follow the instructions for settings in **Figure 26.19**.
17 to 0	Reserved	These bits are read as 0. The write value should be 0.

26.3.34 RDC3AnBISTFX1 ─ Setting Register 1 after BIST Ends

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + 0094_H

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BISTFX1[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.54 RDC3AnBISTFX1 Register Contents

Bit Position	Bit Name	Description
31 to 16	Reserved	These bits are read as 0. The write value should be 0.
15 to 0	BISTFX1[15:0]	Setting after Execution of the Conversion BIST in Angular Conversion Mode 1 Follow the instructions for settings in **Figure 26.19**.

26.3.35 RDC3AnADSTD1 ─ 12-Bit SAR-ADC Digital Circuit Block Setting Register 1

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $00AC_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	ADCALFG	—	—	—	—	—	—	—	ADCALST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ADSFBMD	ADSFBIN[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.55 RDC3AnADSTD1 Register Contents

Bit Position	Bit Name	Description
31 to 25	Reserved	These bits are read as 0. The write value should be 0.
24	ADCALFG	Calibration Processing Flag This flag indicates that an ADC calibration is running. It returns 0 on completion of the calibration.
23 to 21	Reserved	These bits are read as 0. The write value should be 0.
20	Reserved	These bits are read as 0. The write value should be 0.
19 to 17	Reserved	These bits are read as 0. The write value should be 0.
16	ADCALST	ADC Calibration Start This bit returns to 0 after two cycles have elapsed following it being set to 1. Setting 1 to this bit starts ADC calibration within 2 milliseconds.
15 to 13	Reserved	These bits are read as 0. The write value should be 0.
12	ADSFBMD	Software BIST Execution Signal This signal indicates whether the ADC software BIST is processing or not. Set this bit when running the BIST.
11 to 0	ADSFBIN[11:0]	Software BIST Value Setting value at the time of ADC software BIST (12-bit width)

26.3.36 RDC3AnDIAG3 ─ Error Detection Register 3

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $00B0_H$

Value after reset $2000\ 0200_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SQHTH[15:0]															
Value after reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SQLTH[15:0]															
Value after reset	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.56 RDC3AnDIAG3 Register Contents

Bit Position	Bit Name	Description
31 to 16	SQHTH[15:0]	Sum-of-Squares Amplitude Upper Threshold Sets the upper threshold for values of the integrated sum-of-squares amplitude. A value exceeding this is detected as an error. The user can set this value freely based on **Table 26.57** and **Table 26.58**.
15 to 0	SQLTH[15:0]	Sum-of-Squares Amplitude Lower Threshold Sets the lower threshold for the values of the integrated sum-of-squares amplitude. A value exceeding this is detected as an error. The user can set this value freely based on **Table 26.57** and **Table 26.58**.

Table 26.57 Relationship between the Resolver Signal (MNT Signal) Amplitude, Excitation Frequency, and Values of the Integrated Sum-of-Squares Amplitude (when the ADRD bit = 1)

Resolver Signal Amplitude \ Excitation Frequency	5 kHz	7.5 kHz	10 kHz	12.5 kHz	15 kHz	17.5 kHz	20 kHz
0.5 Vpp	254	169	128	104	83	72	63
1.0 Vpp	1032	692	520	417	341	296	255
1.5 Vpp	2304	1533	1156	921	771	649	582
2.0 Vpp	4104	2719	2052	1638	1364	1169	1031
2.5 Vpp	6435	4296	3222	2583	2152	1847	1601
3.0 Vpp	9230	6127	4595	3685	3067	2633	2294
3.5 Vpp	12567	8379	6282	5041	4177	3577	3144
4.0 Vpp	16420	10942	8167	6574	5471	4684	4096
4.5 Vpp	20761	13855	10355	8281	6909	5918	5212

Decimal number

Table 26.58 Relationship between the Resolver Signal (MNT Signal) Amplitude, Excitation Frequency, and Values of the Integrated Sum-of-Squares Amplitude (when the ADRD bit = 0)

Resolver Signal Amplitude \ Excitation Frequency	5 kHz	7.5 kHz	10 kHz	12.5 kHz	15 kHz	17.5 kHz	20 kHz
0.5 Vpp	102	68	51	42	33	29	25
1.0 Vpp	413	277	208	167	136	118	102
1.5 Vpp	922	613	462	368	308	260	233
2.0 Vpp	1642	1088	821	655	546	468	412
2.5 Vpp	2574	1718	1289	1033	861	739	640
3.0 Vpp	3692	2451	1838	1474	1227	1053	918
3.5 Vpp	5027	3352	2513	2016	1671	1431	1258
4.0 Vpp	6568	4377	3267	2630	2188	1874	1638
4.5 Vpp	8304	5542	4142	3312	2764	2367	2085

Decimal number

26.3.37 RDC3AnDIAG4 – Error Detection Register 4

Access Readable/writable in 8-, 16-, and 32-bit units.

Address <RDC3An_base> + $00B4_H$

Value after reset $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	SQCTH[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 26.59 RDC3AnDIAG4 Register Content

Bit Position	Bit Name	Description
31 to 3	Reserved	These bits are read as 0. The write value should be 0.
2 to 0	SQCTH[2:0]	Sum-of-Squares Amplitude Error Excitation Counts Threshold Selects the number of excitation periods in which abnormal amplitudes may be generated in the judgment of integrated sum-of-squares amplitude errors. The number of times set in these bits being exceeded is judged to represent an integrated sum-of-squares amplitude error. b2 b1 b0 0 0 0: 8 times 0 0 1: 1 times 0 1 0: 2 times 0 1 1: 3 times 1 0 0: 4 times 1 0 1: 16 times 1 1 0: 32 times 1 1 1: 64 times

26.4 Functional Description

26.4.1 Tracking Loop

26.4.1.1 PI Compensator Bandwidth Setting Function

This RDC can be set from six different bandwidths (five fixed and one auto-adjusted) using the registers. In addition, coefficients in the PI compensator can be configured in detail by setting the BWCS bit in the RDC3AnPI0 register to 0. For further details, refer to the section on the RDC3AnPI0 register.

26.4.1.2 Forced Gain Control Function

The forced gain control function is designed to improve the tracking performance when the resolver angle deviates significantly from the R/D converted angle in situations such as after a reset.

The forced gain control state is entered when any of the following conditions is met:

[Conditions for starting the forced gain control function]

- Release from the reset state
- The recovery from a resolver signal error
- The recovery from a resolver signal disconnect error
- The start of angular conversion BIST or conversion error BIST
- The end of sum-of-squares amplitude error detection BIST, ADBIST, angular conversion BIST, resolver signal error BIST, disconnection error BIST, conversion error BIST, or power/ground short error BIST
- The end of detection of a power/ground short error
- 1 being written to the KIRST bit of the RDC3AnDIAG1 register

Forced gain control function is applied for approximately 5 ms*[1]. If one of the above conditions is met again during the execution, the forced gain control state is entered again from this point and the period of its execution is extended by approximately 5 ms.
The value of the Kv gain becomes greater on entry to the forced gain state, so even if the resolver rotation angle signal θ and the R/D converter output angle signal ϕ have matched, ϕ may fluctuate significantly for approximately 1 ms.
When the AGCD bit is set to 1, the forced gain state will not be entered even if a condition for starting forced gain control is met, with the exception of following release from the reset state. The initial Kv gain value depends on the type of BIST.

- Angular conversion BIST and conversion error BIST: Initial gain value = × 128 (the maximum value)
- When a short-period BIST (sum-of-squares amplitude error detection BIST, ADBIST, resolver signal error BIST, disconnection error BIST, or power/ground short error BIST) ends: Initial gain value = gain value specified by the AGST[3:0] bits
- When detection of a power/ground short error ends: Initial gain value = gain value specified by the AGST[3:0] bits

A short-period BIST or detection of a power/ground short error can also proceed during angle conversion since both end in a short period of time and tracking of the angle continues even while the test is running. This function has a mechanism to adjust the initial gain by using a register since fluctuations of phi increase if the initial Kv gain value at the end of BIST or detection of a power/ground short error is too large.

Forced gain control is subject to the following restrictions.

1. When changing the setting of the AGCD bit of the register for enabling or disabling forced gain control, do so while the resolver is not running.
   After that, write 1 to the KIRST bit. Not setting the KIRST bit to 1 may result in entry to the free-running state.
2. Forced gain control is subject to restrictions on the angular velocity that allows operation. The upper-limits according to the maximum angular velocity/resolution setting (MAXV[2:0] bit setting) are as listed below.

   When the resolution is equal to or less than 12 bits: 120000 min^{-1}
   When the resolution is 13 bits: 60000 min^{-1}
   When the resolution is 14 bits: 30000 min^{-1}
   When the resolution is 16 bits: 7500 min^{-1}

Note 1. The timing value is when the CCLK is running at 40 MHz.

26.4.1.3 Excitation Signal Source Selection Function

Without the excitation signal output from the RDC3AnRSO pin and common voltage output from the RDC3AnCOM pin, the R/D conversion can be performed using externally generated excitation signal input to pins RDC3AnRSO and RDC3AnCOM pins. When an externally input signal is used, set the EXIO bit in the RDC3AnREF register to 0.

26.4.1.4 Required Sensor Selection Function

The DC resolver signal (E • sinθ, E • cosθ) which does not contain excitation component can be used by setting the SENS bit in the RDC3AnREF register to 0. When the DC resolver signal is used, the excitation component extraction function is disabled.

26.4.1.5 Excitation Component Extraction Function

The excitation signals (RDC3AnRSO, RDC3AnCOM) and resolver signals (RDC3AnS1 to RDC3AnS4) are analog signals input to the RDC. If there is a phase difference between the excitation component (sine wave component) in the excitation signal line and that in the resolver signal line, it can cause an error in the angle conversion result in proportion to the phase difference. The phase difference between the resolver signal and excitation signal can be reduced by using the excitation component contained in the resolver signal line for an angle conversion.

When using an external excitation signal by setting the EXIO bit in the RDC3AnREF register to 0, extracted excitation components cannot be used if the difference between the electrical angle of the resolver and the RDC converted angle is large (such as during power-up or occurrence of error). Therefore, when using an external excitation signal, be sure to enter the external excitation signal into the RDC3AnRSO, RDC3AnCOM pins.

The excitation component extraction function automatically performs the following process; the excitation signal (RDC3AnRSO, RDC3AnCOM) is used when the difference between the resolver electrical angle and the R/D converted angle is large, and the extracted excitation component is used when the difference is small. The exact adjustment of the phase difference between excitation and resolver signals is not required because of this function.

For successful operation of the excitation component extraction circuit, the excitation component phase deviation between the excitation component of the excitation signal (RDC3AnRSO, RDC3AnCOM) and that of the resolver signal (RDC3AnS1 to RDC3AnS4) should be within 45°.

26.4.1.6 Maximum Angular Velocity Setting Function

This function can set the maximum angular velocity (resolution) that is capable of tracking operation by using the MAXV [2:0] bits in the RDC3AnPI1 register. Discrete values do not occur in the selected resolution.

26.4.1.7 Compare Match Interrupt

When the angle set in the CMPj register (j = 0 to 2) and the R/D converted angle match, a compare match interrupt request signal is generated. The bit width that is compared for a match is set using the MAXV [2:0] bits (maximum angular velocity selection) in the RDC3AnPI1 register.

The compare match interrupt request signal can be selected from either a compare match signal or a signal latching a compare match signal by using the IRS bit in the RDC3AnPHICP0 register. When the IRS bit is set to 0, the compare match interrupt request signal becomes high when the R/D converted angle and the angle set in the CMPj register match, and becomes low when the values do not match.

When the IRS bit is set to 1, the compare match interrupt request signal becomes high when the R/D converted angle and the angle set in the CMPj register match, and also the INTFLG[2:0] flag in the RDC3AnPHICP0 register becomes 1. In this case, the request signal retains the high level even when the values do not match.

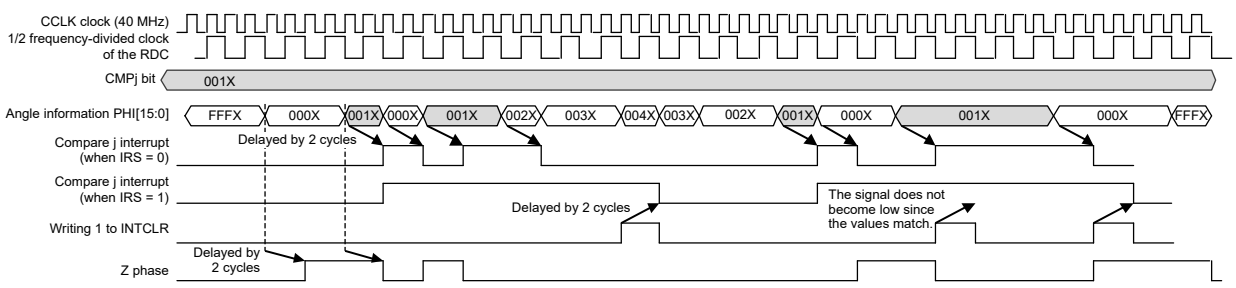
When 1 is written to the INTCLR[2:0] bit in the RDC3AnPHICP0 register while the R/D converted angle and the angle set in the CMPj register do not match, the request signal becomes low and the INTFLG[2:0] flag becomes 0.

If 1 is written to the INTCLR[2:0] bit while the R/D converted angle and the angle set in the CMPj register match, the request signal remains high and the INTFLG[2:0] flag does not become 0.

Turning hysteresis on by setting the HYSS bit in the RDC3AnENC0 register to 0 prevents chattering of the output of the compare match interrupt signal and Z output signal when the angle output near the target bit for comparison is not stable.

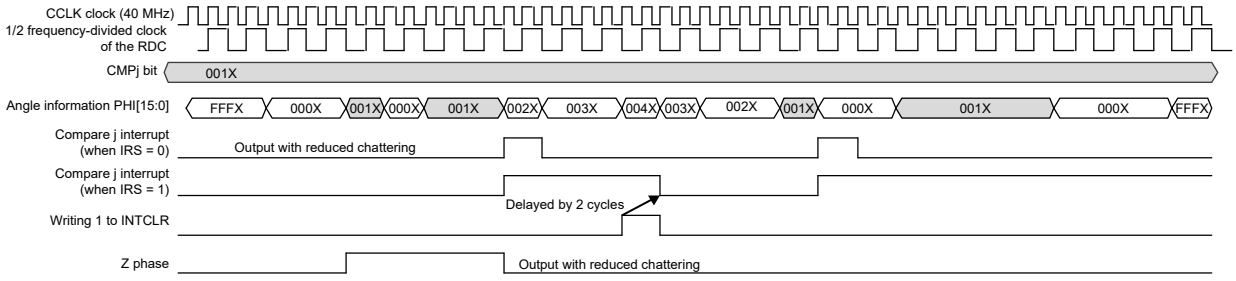
This hysteresis circuit can only be used when 12-bit resolution is selected (MAXV[2:0] = 001_B).

Figure 26.3 and **Figure 26.4** show the timing charts for the compare match interrupt request signal when hysteresis is off and on, respectively.



Note: n = 0, j = 0 to 2
IRS, INTCLRj: Bits in the RDC3AnPHICP0 register
This example is the case when MAXV[2:0] = 001_B (12-bit resolution selected).

Figure 26.3 Timing Chart for the Compare Match Interrupt Request Signal when Hysteresis is Off



Note: n = 0, j = 0 to 2
IRS, INTCLRj: Bits in the RDC3AnPHICP0 register
This example is the case when MAXV[2:0] = 001_B (12-bit resolution selected).

Figure 26.4 Timing Chart for the Compare Match Interrupt Request Signal when Hysteresis is On

26.4.1.8 Encoder Pulse Output Function

This function enables to output encoder pulse signals (A, B, Z, U, V, W phases).

The Z-phase interrupt signal becomes high while the R/D converted angle is 0º. The bit width compared for a match is set using the MAXV [2:0] bits in the RDC3AnPI1 register as it is for the compare match interrupt request signal. The encoder pulse signal is output when the corresponding bit in the RDC3AnENC0 register is set to 1 (enables output).

Whether the encoder pulse signals are to be output through the hysteresis circuit or without going through it can be selected using the HYSS bit in the RDC3AnENC0 register. This hysteresis circuit can only be used when 12-bit resolution is selected (MAXV[2:0] = 001_B).

Figure 26.5 and **Figure 26.6** show the waveforms of encoder phase pulse operation when hysteresis is on and off, respectively.

See **Figure 26.3** and **Figure 26.4** for the z output waveform when the angle output near the target bit for comparison is not stable.

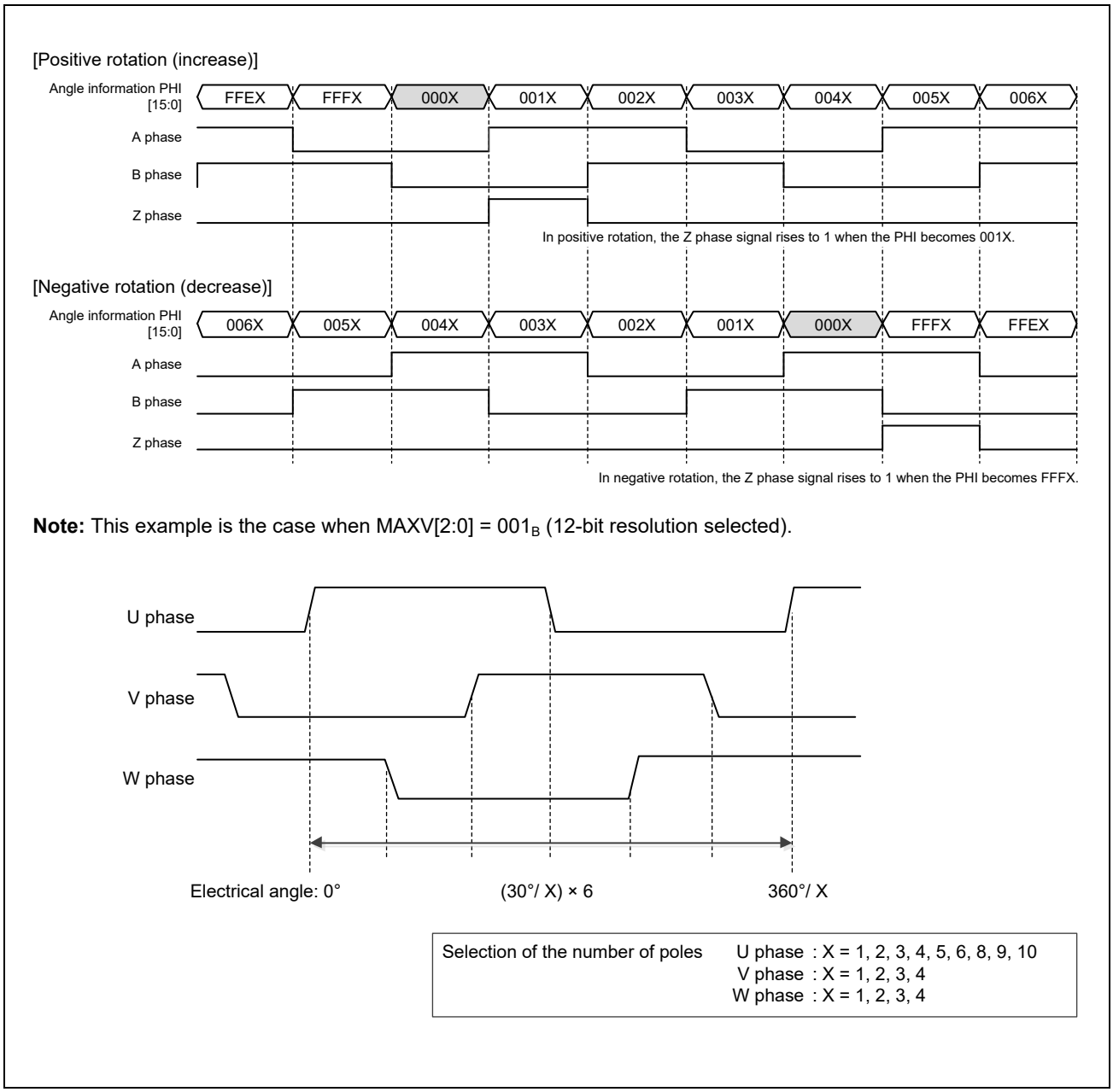


Figure 26.5 Waveform of Encoder Phase Pulse Operation when Hysteresis is On

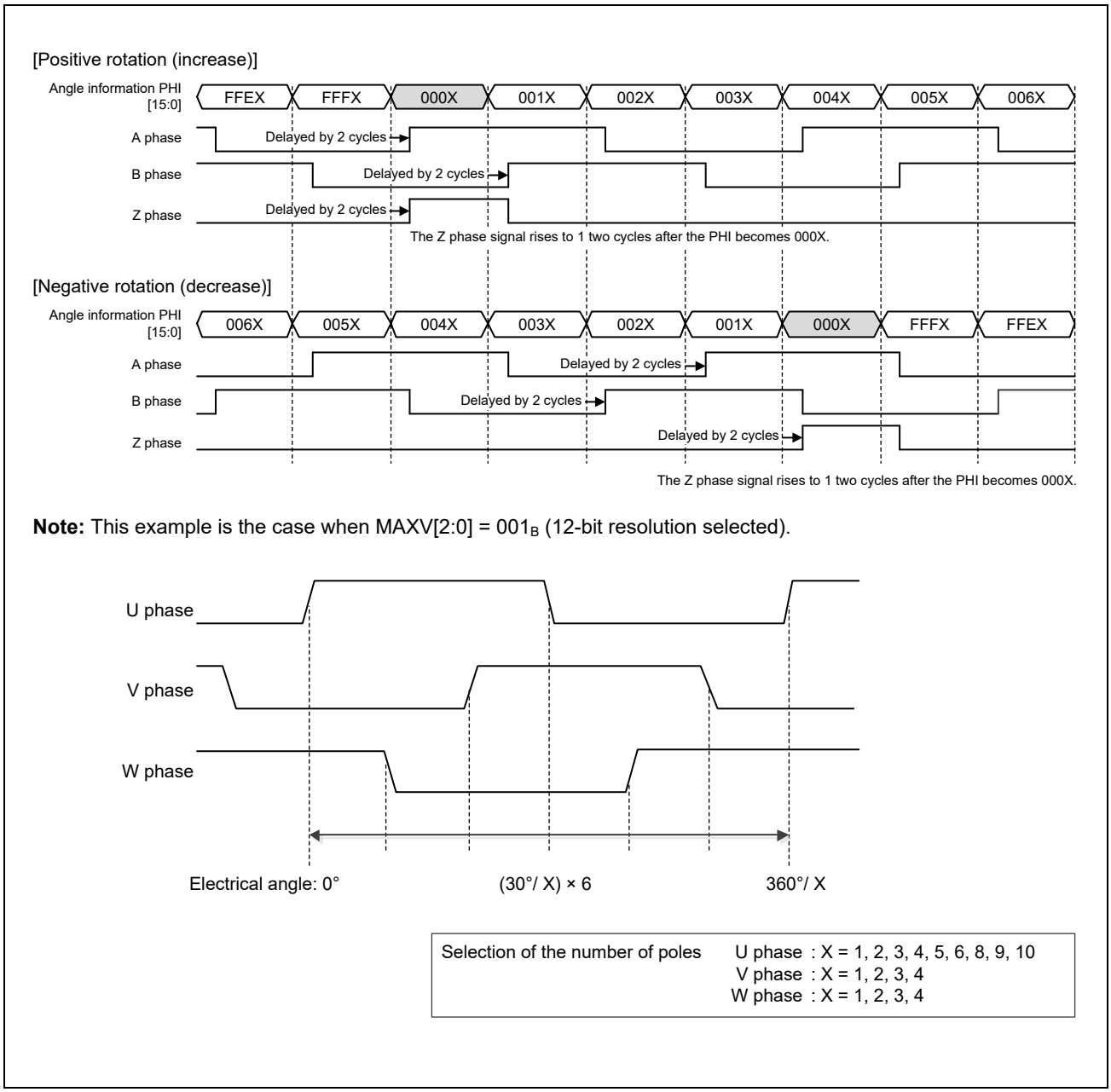


Figure 26.6 Waveform of Encoder Phase Pulse Operation when Hysteresis is Off

26.4.1.9 PHI Angular Velocity Information Reading Function

The angular velocity of the phi angle is read from the relevant register. The amount of change in the phi angle is measured over the period selected by the OMGPTC[1:0] bits, converted into the units of angular velocity (min^{-1}), and stored in the OMG[31:0] bits of the RDC3AnOMG register. The angular velocity of phi (in min^{-1}) does not depend on the selected measuring period, but the maximum measurable angular velocity does. The value of the OMG[31:0] bits is updated every time the measuring period is changed, and the period can be changed even during conversion.

Table 26.60 Contents of OMGPTC[1:0] Bits

OMGPTC[1:0]	Measuring Period	Maximum Measurable Angular Velocity (min^{-1})
00	12.8 µs	2,343,750
01	51.2 µs	585,938
10	102.4 µs	292,969
11	204.8 µs	146,484

Note: This table shows the relation between the measuring period and the range of measurable angular velocity.

Table 26.61 OMG Bits and the PHI Angular Velocity (min^{-1})

Bit	Angular Velocity (min^{-1})
b31 to 25	Sign (0: +, 1: –)
b24	1,171,875
b23	585,937.5
b22	292,968.8
b21	146,484.4
b20	73,242.19
b19	36,621.09
b18	18,310.55
b17	9,155.27
b16	4,577.64
b15	2,288.82
b14	1,144.41
b13	572.20
b12	286.10
b11	143.05
b10	71.53
b9	35.76
b8	17.88
b7	8.94
b6	4.47
b5	2.24
b4	1.12
b3	0.56
b2	0.28
b1	0.14
b0	0.07

For example, the angular velocity is – 164,795 min^{-1} if the result for angular velocity read from the OMG bits is FFDB $FFFF_H$.

26.4.1.10 Monitor Function

This function is for reading a control variation value directly from a register.

When reading the control variation in angular conversion mode 0 (ADRD = 0), read the DATA[7:0] bits while the DATSEL[5:0] bits are set to 05_H.

When reading the control variation in angular conversion mode 1 (ADRD = 1), read the DATA[7:0] bits while the DATSEL[5:0] bits are set to $2F_H$.

Values read are expressed as two's complements. The control variation is ± 0% under the ideal condition where the resolver input angle exactly matches the phi digital angle output ($\theta = \phi$).

When the ADRD bit is 0, the value read is the control variation within angular conversion mode 0.

When the ADRD bit is 1, the value read is the control variation within angular conversion mode 1.

Table 26.62 lists the relationship between the DATA[7:0] bits and control variation values (%).

Table 26.62 Relationship between the Bits of the RDC3AnTBUS Register and Control Variation Values (%)

Bit	Control Variation (%)
b7	Sign (0: positive, 1: negative)
b6	50%
b5	25%
b4	12.5%
b3	6.25%
b2	3.13%
b1	1.56%
b0	0.78%

26.4.2 Sine and Cosine Correction Function

26.4.2.1 ADC Noise Elimination Function

This function eliminates noise from the values of the SINMNT and COSMNT signals obtained by AD conversion.

The converted values of the SINMNT and COSMNT signals change every 2 µs (when angle conversion using the angular conversion mode 1 is selected by setting ADRD to 1).

The value of the MNT signal obtained from a conversion is compared with that from the previous session, and, if the gap between two values is greater than the threshold set in the GNCNS[2:0] bits, the value from the current session is judged to be noise, and the previous value is retained. It is also possible to compare the value of the MNT signal with that from the session before the previous one (by setting the GNJSP bit to 1).

Figure 26.7 shows how noise is detected and removed from the SINMNT signal.

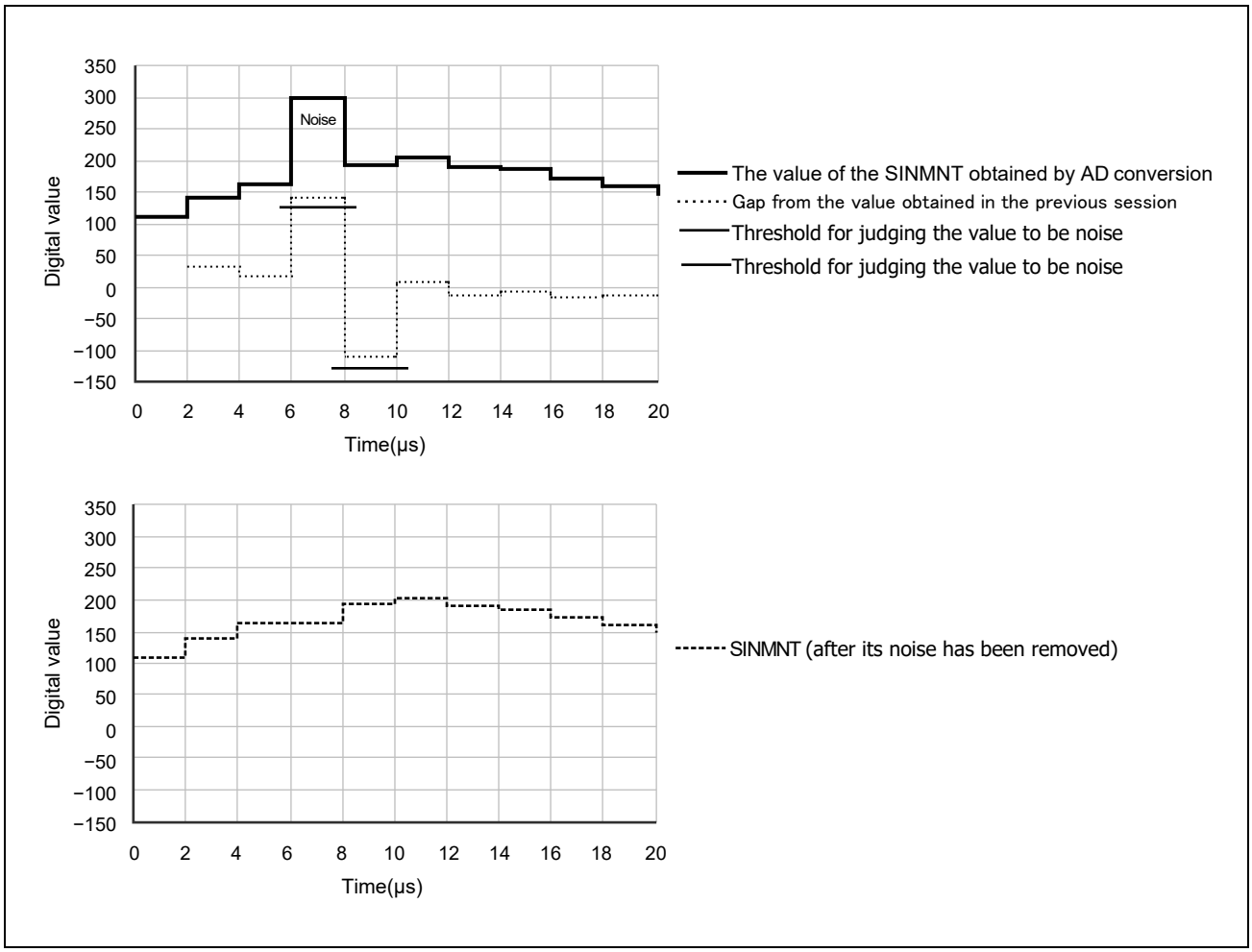


Figure 26.7 Judging and Eliminating Noise

If the result of AD conversion becomes stuck at a value equivalent to the power supply voltage or the ground due to extremely strong noise, the gaps between the compared values may not exceed the thresholds with the result that noise is directly output. To avoid this, this device is also equipped with a function to detect the value obtained by AD conversion being stuck. If the result of AD conversion is greater than 0.992 × RVCC or smaller than 0.008, the input is determined to be stuck, and the result from the previous conversion is retained.

Writing 1 to the GNCND bit disables the noise elimination function for the ADC. If it is disabled, results of AD conversion are used in operations as they are, without being checked for noise.

The circuits which use the ADC output signal whose noise has been removed by this function are selected by the NSRSL bit.

- NSRSL = 0: (default) Use the ADC output signal of which noise has been removed only in the sine and cosine gain correction circuits and the sine and cosine common offset correction circuits.
- NSRSL = 1: Use the ADC output signal of which noise has been remobed in the sine and cosine gain correction circuits, the sine and cosine common offset correction circuits, and the angle conversion calculation (adder and subtractor) circuits.

26.4.2.2 Sine and Cosine Gain Correction

Resolver errors or errors in the accuracy of components of the printed circuit board may lead to gaps between the amplitudes of the sine- and cosine-wave signals input to the RD converter that lead to errors in the result of angle conversion by the RDC. In order to reduce errors in the results of conversion, the maximum amplitudes of the sinmnt and cosmnt signals are compared, and if there is a gap between the amplitudes, that of the cosmnt signal is automatically adjusted to have the same range as that of the sinmnt signal. The values after correction are used in operations. This function is only applicable to the values used internally. The SINMNT and COSMNT signals from the LSI chip are not adjusted.

Note that this function is effective for reducing errors in the results of angle conversion only in angler conversion mode 1 (ADRD = 1). This function can be run with the angular conversion mode 0 (ADRD = 0), but does not have the effect of reducing differences between the results of angle conversion.

If the resolver angle is fixed, the maximum amplitudes of the sinmnt and cosmnt signals cannot be compared.

In this case, the resolver is required to go through at least half a rotation (electrical angle) to obtain signal values for comparison.
Accordingly, when the RDC is ready to output results of conversion, the user needs to wait for the resolver to go through at least half a rotation (electrical angle) before writing 1 to the GNCST bit. The gain correction function will then be ready for use. Writing 1 to the GNCST bit before waiting for the required period may result in invalid correction value.

The range of values for correction is specified by setting a percentage for the limit on correction values in the GNCLT[1:0] bits. Results of conversion beyond the specified range will not corrected in the same way as values within the range.
If the value for correction obtained from comparison of sinmnt and cosmnt is greater than the specified limit, the specified limit is applied to the value for correction.

GNCLT[1:0] = 00_B: limit ± 20% (default)

GNCLT[1:0] = 01_B: limit ± 10%

GNCLT[1:0] = 10_B: limit ± 40%

GNCLT[1:0] = 11_B: limit ± 0% (no correction)

Figure 26.8 below shows how the amplitude of COSMNT, which is initially smaller than that of SINMNT, is corrected to have the same amplitude as SINMNT.

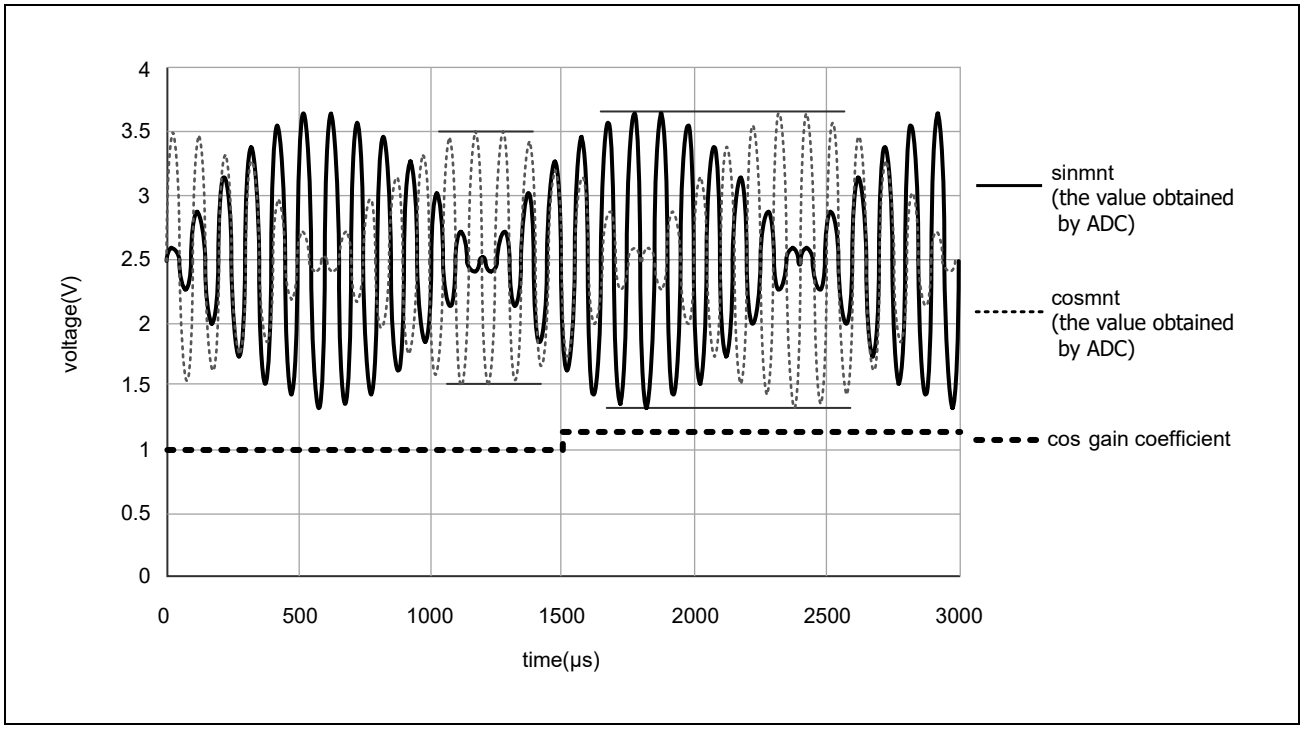


Figure 26.8 Correction to Make the Amplitudes Correspond

The timing of correcting the gain value is selected by the GNCSL[1:0] bits as follows;

- GNCSL[1:0] = 00_B

  No correction (the default value, 1024, is used as the value for correction)

- GNCSL[1:0] = 01_B

  The fixed value set in the GNCNM[11:0] bits is used as the value for correction.

- GNCSL[1:0] = 10_B

  The value calculated when the GNCST bit is set to 1 is used as the value for correction. The value from the calculation is retained.

- GNCSL[1:0] = 11_B

  The value is updated at the time of z-phase output following GNCST being set to 1.

  The value is updated on every single rotation.

A limit can be set to the value for common offset correction. No correction value beyond the limit specified by the CMCLT[1:0] bits will be applied. If the value for correction obtained from calculation is greater than the specified limit, the specified limit is applied. If the value for correction obtained from calculation is within the specified limit, the calculated value is applied.

26.4.2.3 Sine and Cosine Common Offset Correction

A resolver error or error in the accuracy of the components in the printed circuit board may lead to an offset from the original common level (0.5 × RVCC) for the common levels of the sine- and cosine-wave signals, which are input to the RD converter. This may lead to errors in the results of angle conversion by the RDC.

In order to reduce errors in the results of conversion, an offset of the common level of the sinmnt and cosmnt signals from the original one is detected and automatically adjusted. The values after correction are used in operations. This function is only applicable to the values used internally. The SINMNT and COSMNT signals from the LSI chip are not adjusted.

Note that this function is effective for reducing errors in the results of angle conversion in the angular conversion mode 1 (ADRD = 1). This function can be run in the angular conversion mode 0 (ADRD = 0), but does not have the effect of reducing differences between the results of angle conversion.

If the resolver angle is fixed, the maximum amplitudes of the sinmnt and cosmnt signals cannot be compared. In this case, the resolver is required to go through half a rotation (electrical angle) to obtain signal values for comparison.

Accordingly, when the RDC is ready to output the results of conversion, the user needs to wait for the resolver to go through at least half a rotation (electrical angle) before writing 1 to the GNCST bit. The gain correction function will then be ready for use. Writing 1 to the GNCST bit before waiting for the required period may result in invalid correction values.

A limit can be set to the value for common offset correction by CMCLT[1:0] bits. No correction value beyond the specified limit will be applied. If the value for correction obtained from sinmnt and cosmnt is greater than the specified limit, the specified limit is applied.

CMCLT[1:0]=00_B: limit ± 0.0312 × RVCC (approx. ± 150mV) (default)

CMCLT[1:0]=01_B: limit ± 0.0156 × RVCC (approx. ± 78mV)

CMCLT[1:0]=10_B: limit ± 0.125 × RVCC (approx. ± 625mV)

CMCLT[1:0]=11_B: limit ± 0 × RVCC (± 0mV) (no correction)

Figure 26.9 below shows how the common offset of COSMNT and SINMNT are corrected to 2.5V (= 0.5 × RVCC).

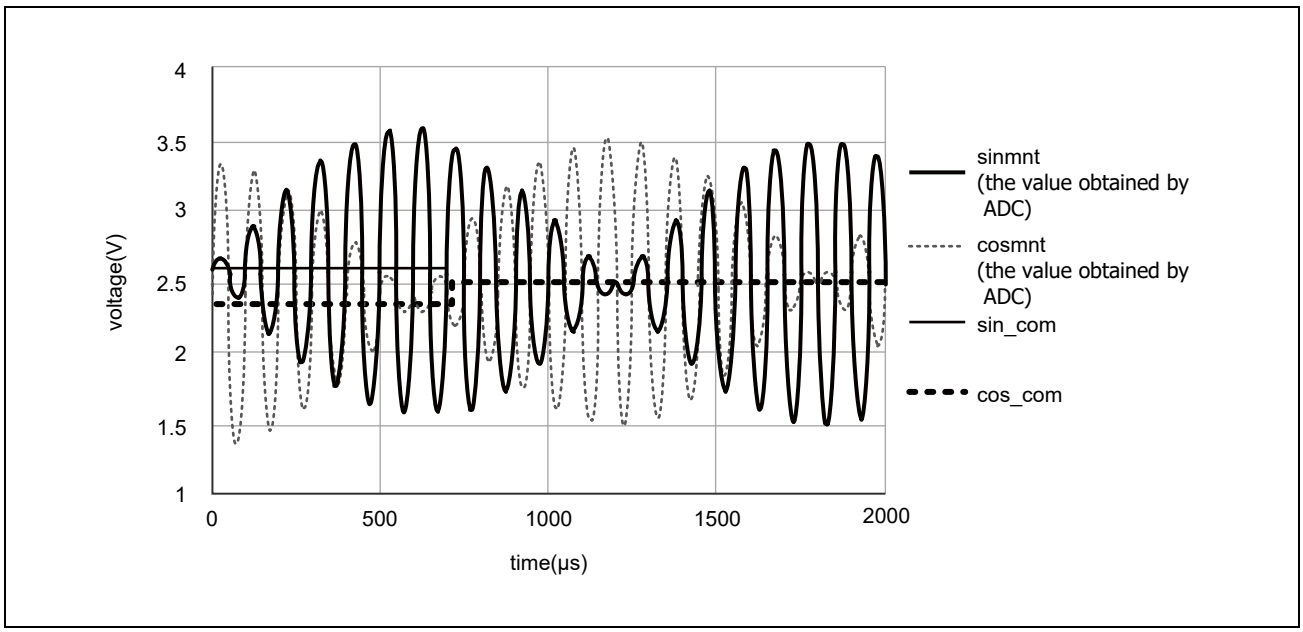


Figure 26.9 Correction of Common Offsets to 2.5V (= 0.5 × RVCC)

The timing of updating the value for common offset correction in selected by the CMCSL[1:0] bits as follows;

- CMCSL[1:0] = 00_B
  No correction (0 is used as the value for correction)
- CMCSL[1:0] = 01_B
  The value for correction set in CCOSN[9:0] is selected as the value for the cosine side.
  The value for correction set in CSOSN[9:0] is selected as the value for the sine side.
- CMCSL[1:0] = 10_B
  The calculated value is used as the value for correction.
  - When GNCSL[0] = 0, the value calculated when the GNCST bit is set to 1 is used as the value for correction. The value from the calculation is retained.
  - When GNCSL[0] = 1, the value is updated at the time of z-phase output following GNCST being set to 1. The common value for correction is updated on every single rotation.
- CMCSL[1:0] = 11_B
  Setting prohibited

A limit can be set to the value for common offset correction by CMCLT[1:0] bits. No correction value beyond the specified limit will be applied. If the value for correction obtained from sinmnt and cosmnt is greater than the specified limit, the specified limit is applied.

26.4.2.4 SIN and COS Phase Correction Function

Resolver errors or errors in the accuracy of components on the printed circuits board may lead to deviations between the time (phase) of the sine- and cosine-wave signals input to the RD converter. In order to reduce errors in the results of conversion, deviations between the phases of the sinmnt and cosmnt signals are detected by comparing the timing of their excitation zero-crossing. Detected deviations are corrected by delaying the signal which has a phase in advance of the other.
This function is only applicable to the values used internally.
The SINMNT and COSMNT signals from the LSI chip are not adjusted.

Note that this function is effective for reducing errors in the results of angle conversion in the angular conversion mode 1 (ADRD = 1). This function can be run in the angular conversion mode 0 (ADRD = 0), but does not have the effect of reducing difference between the results of angle conversion.

When the power supply voltage is applied and the RDC initialization sequence is performed so that it is ready for use in angular tracking, write 1 to the register bit PHCST which gives an instruction to start phase correction of the Sine and Cosine signals. After 1 is written to this bit, application of the corrected values starts.

Note that the obvious variations in amplitude between the sinmnt and cosmnt signals are required in order to compare the timing of their excitation zero-crossings. For this reason, this function automatically detects the phase deviation when the output phi angle is within ± 16.8° of either 45°, 135°, 225°, and 315°.

A limit can be set to the delay value for phase correction by PHCLT[1:0] bits. No correction value beyond the specified limit will be applied. If the value for correction obtained from sinmnt and cosmnt is greater than the specified limit, the specified limit is applied.

PHCLT[1:0]=00_B: limit is 3 µs (default)

PHCLT[1:0]=01_B: limit is 1 µs

PHCLT[1:0]=10_B: limit is 6 µs

PHCLT[1:0]=11_B: limit is 0 µs (no correction)

Figure 26.10 below shows the waveforms where the sinmnt signal delays from the cosmnt signal.

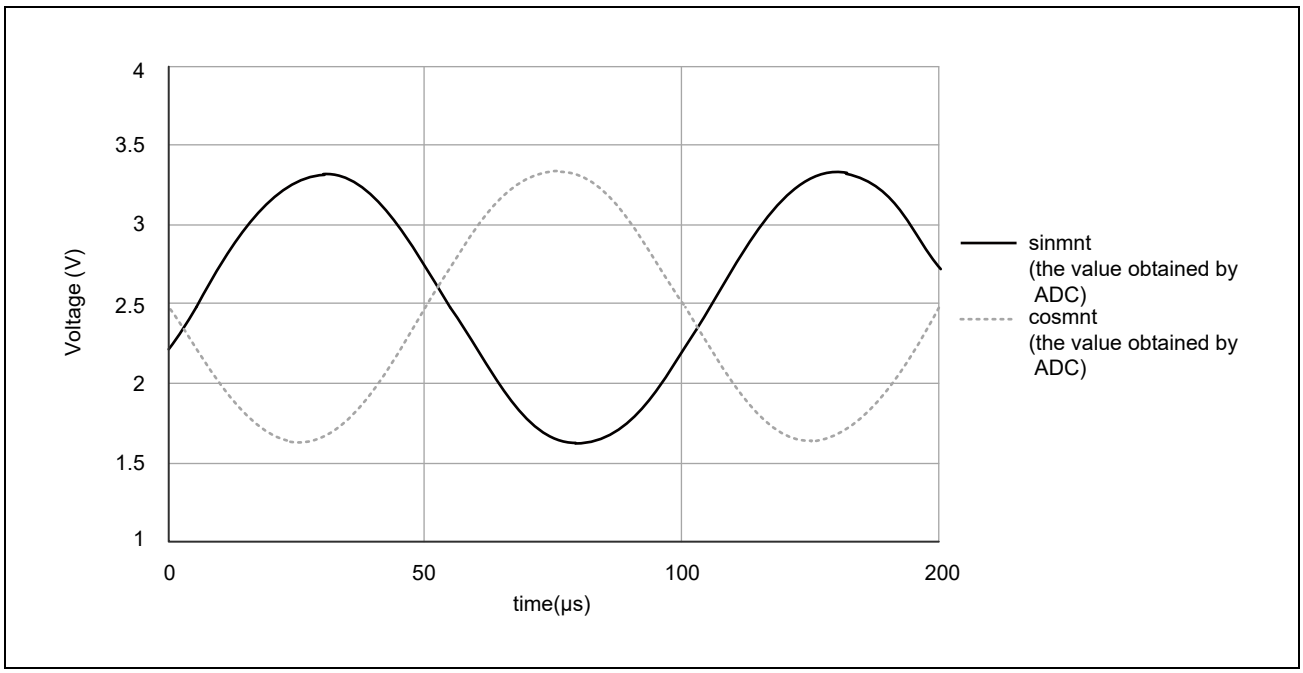


Figure 26.10 Signal with Delay

Figure 26.11 below shows the two synchronized waveforms obtained by detecting the advanced phase of the cosmnt signal and delaying.

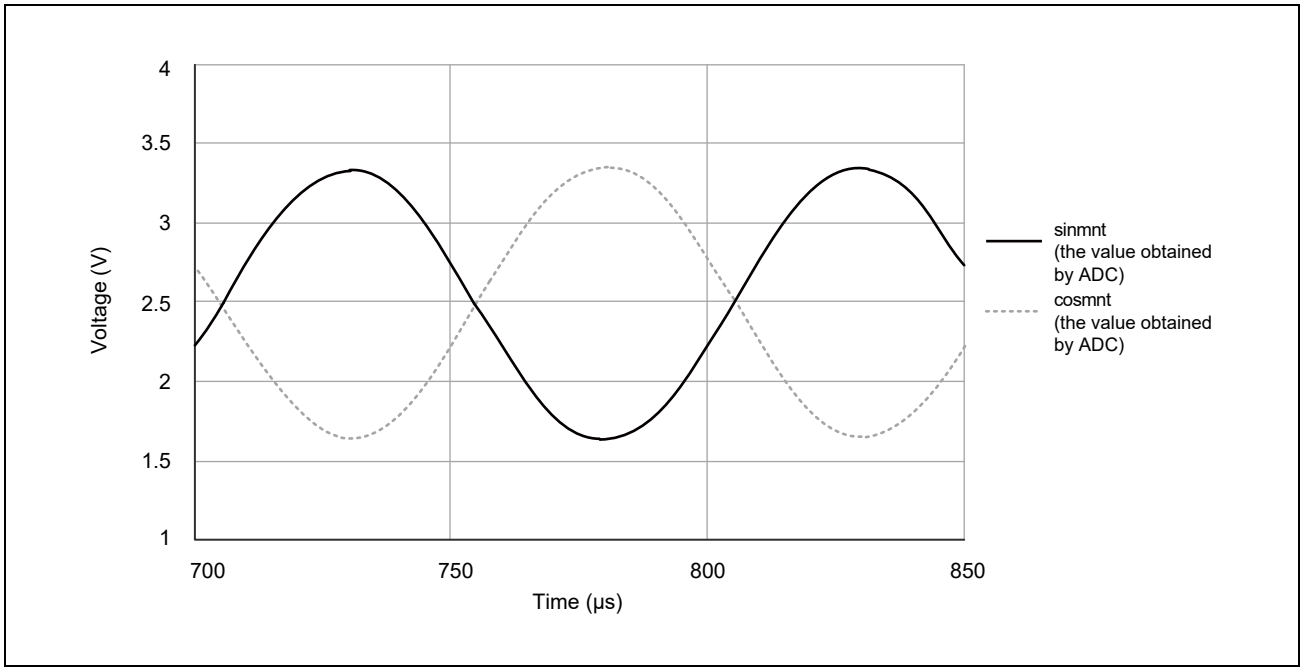


Figure 26.11 Phase Synchronization

The timing of correcting the phase deviation after writing 1 to the PHCST bit is select by the GNCSL[1:0] bits as follows;

- PHCSL[1:0] = 00_B
  No correction (default)
- PHCSL[1:0] = 01_B
  The value for correction is obtained by taking the average of phase deviations on the sixteen zero-crossings of the excitation signals. The value for correction is updated every time 1 is written to the PHCST bit.
- PHCSL[1:0] = 10_B
  The value for correction is obtained by taking the average of phase deviations on the sixteen zero-crossings of the excitation signals. The value for correction obtained in the first round is retained.
- PHCSL[1:0] = 11_B
  The fixed correction values set by the PHSNM[5:0] bits (value for delay in the sine side) and the PHCNM[5:0] bits (value for delay in the cosine side) are used.

26.4.2.5 Sine and Cosine Angle Correction Function

The RDC is capable of handling the mounting of the resolver at an angle to the motor shaft by correcting the resulting orthogonality errors of the sine and cosine signals from the resolver. Correction in this case is by adding fixed values for correction to the phi angles to be input to the individual SINROM and COSROM tables. These values are set in the SINPO[11:0] and COSPO[11:0] bits in the sine and cosine angle correction register. Set these bits to 0° for angle conversion BIST.

Sine phase correction bits SINPO[11:0] = 000_H

Cosine phase correction bits COSPO[11:0] = 000_H

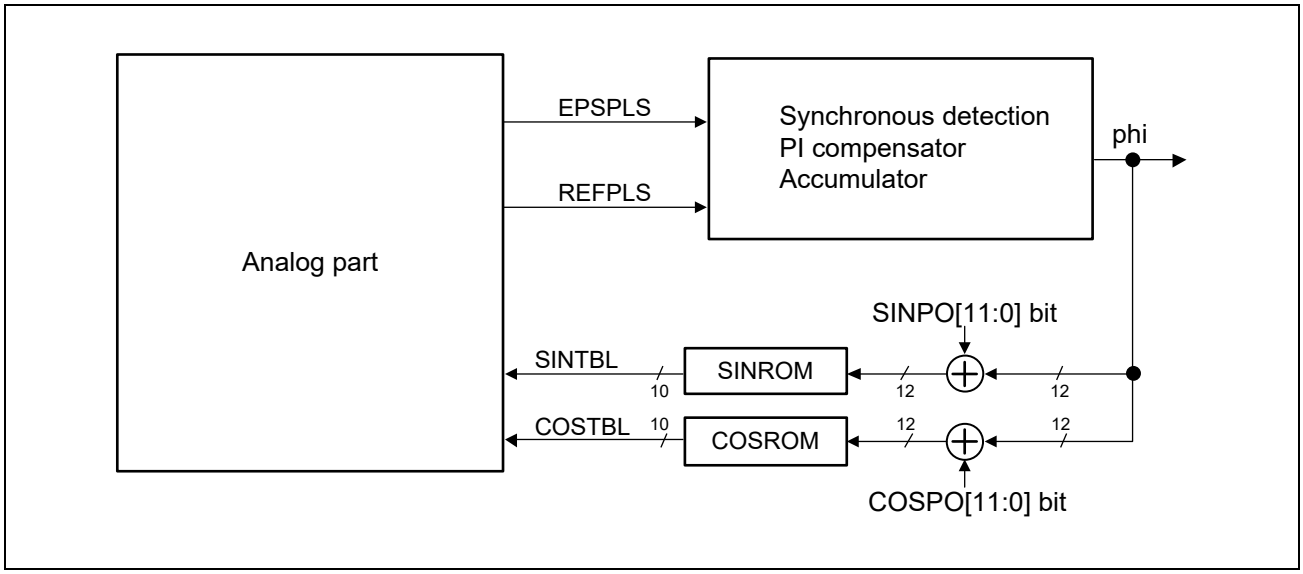


Figure 26.12 Structure of Sine and Cosine Angle Correction Circuit

26.4.3 Excitation Signal Output

26.4.3.1 Excitation Signal Output (RDC3AnRSO, RDC3AnCOM)

Sine wave voltage signal generated by 7-bit D/A can be output from the RDC3AnRSO pin. RVCC/2 (2.5V) common voltage can be output from the RDC3AnCOM pin. Setting the EXIO bit in the RDC3AnREF register to 0 (excitation signal input) disables the output and the RDC3AnRSO and RDC3AnCOM pins become input. The amplitude of the sine wave signal that is output from the RDC3AnRSO pin is set in the EXOC[1:0] bits in the RDC3AnATMNT0 register. The amplitude of the standard value is 0.4 × RVCC [Vp-p].

26.4.3.2 Automatic Amplitude Adjustment

In order to obtain an appropriate R/D conversion accuracy, the resolver signal input amplitude (monitor signal amplitude) needs to be controlled within a range of 0.36 × RVCC to 0.64 × RVCC [Vp-p]. The automatic amplitude adjustment function automatically adjusts the excitation signal output amplitude and the input gain resistance so that the monitor signal amplitude fits within the approximate range of 0.36 × RVCC to 0.64 × RVCC [Vp-p].

The automatic adjustment is performed on the excitation signal output amplitude and the input gain resistance. Priority can be specified by setting the EAAOD bit in the RDC3AnATMNT0 register. When the EAATSP bit in the RDC3AnATMNT0 register is set to 0, the automatic adjustment can be performed continuously. By setting the EAATSP bit to 1, the automatic adjustment can be stopped and the adjusted value at that time is retained. Setting the EAATSP bit to 0 restarts the automatic adjustment process.

The function monitors the amplitude with a 12-bit SAR-ADC, determines the size of amplitude based on the integrated sum-of-squares of the excitation amplitude for a period of 1 ms, and adjusts the amplitude. The determination threshold is approximately 3 Vp-p on the high side, and approximately 2 Vp-p on the low side.

It is also possible to use the excitation signal output settings and the input gain resistance settings that are specified in the register, without using the results of automatic adjustments.

26.4.4 Error Detection

26.4.4.1 Error Detection

This function monitors and detects errors in resolver signal and in R/D conversion operations. If any of the errors listed in **Table 26.63** is detected, an RDC error interrupt request is generated, and the corresponding bit in the RDC3AnDGOUT0 and 1 registers become 1.

The following table lists factors that lead to error detection.

Table 26.63 Detected Errors

Item	Detected Factor
Resolver signal error	• Excitation signal line disconnection (RDC3AnRSO, RDC3AnCOM) including contact failure • Excitation signal down (excitation signal output circuit down, short circuit between lines) • Short circuits between signal lines (RDC3AnS1 and RDC3AnS3, RDC3AnS2 and RDC3AnS4) • Resolver coil layer short
Resolver signal disconnect error	Resolver signal lines disconnection (RDC3AnS1 to RDC3AnS4) including contact failure
R/D conversion error	Excessive control variation (ε) of tracking control loop (negative feedback control system)
Two path comparison conversion error	Comparison of the conversion results from the two angle conversion loops
Resolver signal Power/ground short error	Short circuit of the power supply (power short error) and short circuit to the ground (ground short error) for the resolver signal lines
Square-sum amplitude error	Modulation, distortion, or noise on the amplitude of the sine and cosine resolver input signals.

26.4.4.2 Resolver Signal Error Detection

This function detects disturbance in the resolver signal balance caused by an error in excitation signals input to the resolver. When a resolver signal error is detected, an RDC error interrupt request signal becomes high. A resolver signal error is detected if the monitor outputs (RDC3AnSINMNT, RDC3AnCOSMNT) fall below the threshold for approximately 220 µs.

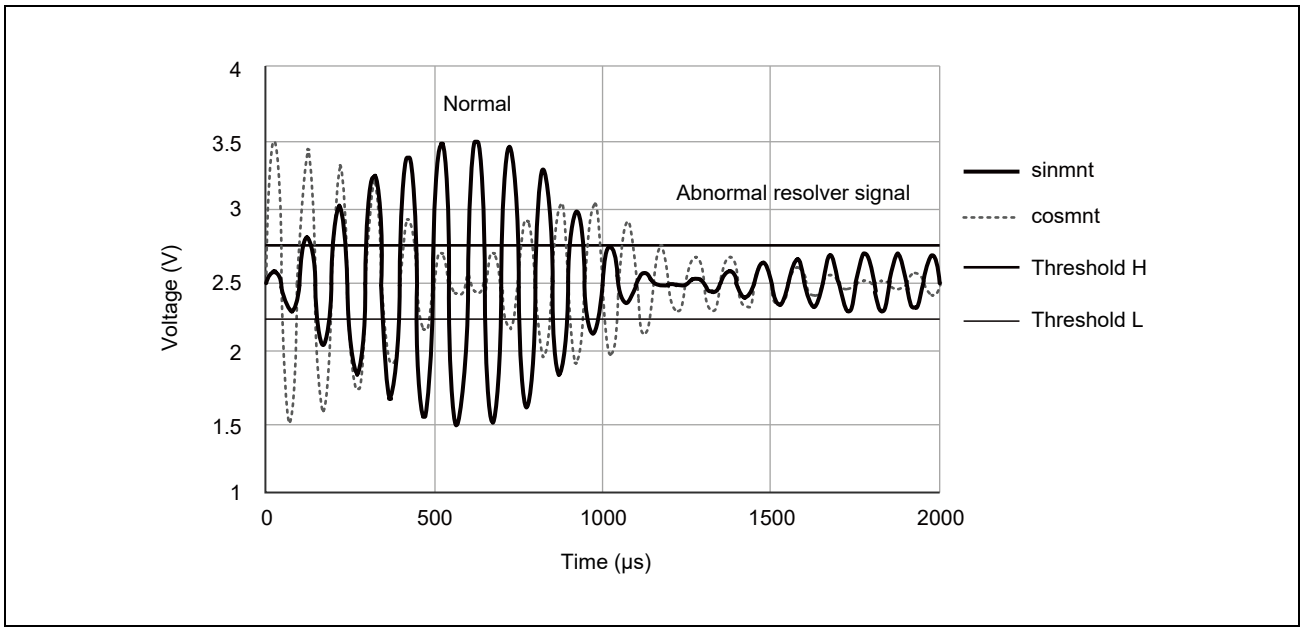


Figure 26.13 Monitor Signal Waveforms when the Resolver Signal is Abnormal

26.4.4.3 Resolver Signal Disconnection Error Detection

This function detects disconnection (including contact failure) of the resolver signals (RDC3AnS1 to RDC3AnS4). When a resolver signal disconnect error is detected, the RDC error interrupt request signal becomes high. A resolver signal disconnect error is detected if the DC level fluctuations in the monitor output (RDC3AnSINMNT, RDC3AnCOSMNT) rise above the threshold. For a description of the relationship between register values and threshold values, see **Section 39.5.3, Error Detect Characteristics**.

When operation with a VR resolver is selected (by setting values other than the combination of EXIO = 1 and SENS = 0), this function monitors the common levels of the monitored signals and determines any case of them exceeding the configured threshold to be a disconnection error. The threshold is set at 2.9V as the default and can be changed by the SGBTH[7:0] bits.

When operation with a DC resolver is selected (by setting the combination of values as EXIO = 1 and SENS = 0), this function monitors the DC level of the monitored signals and determines any case of them exceeding the configured threshold to be a disconnection error. The threshold is set at 4.25 V as the default and can be changed by the SGBDTH[7:0] bits.

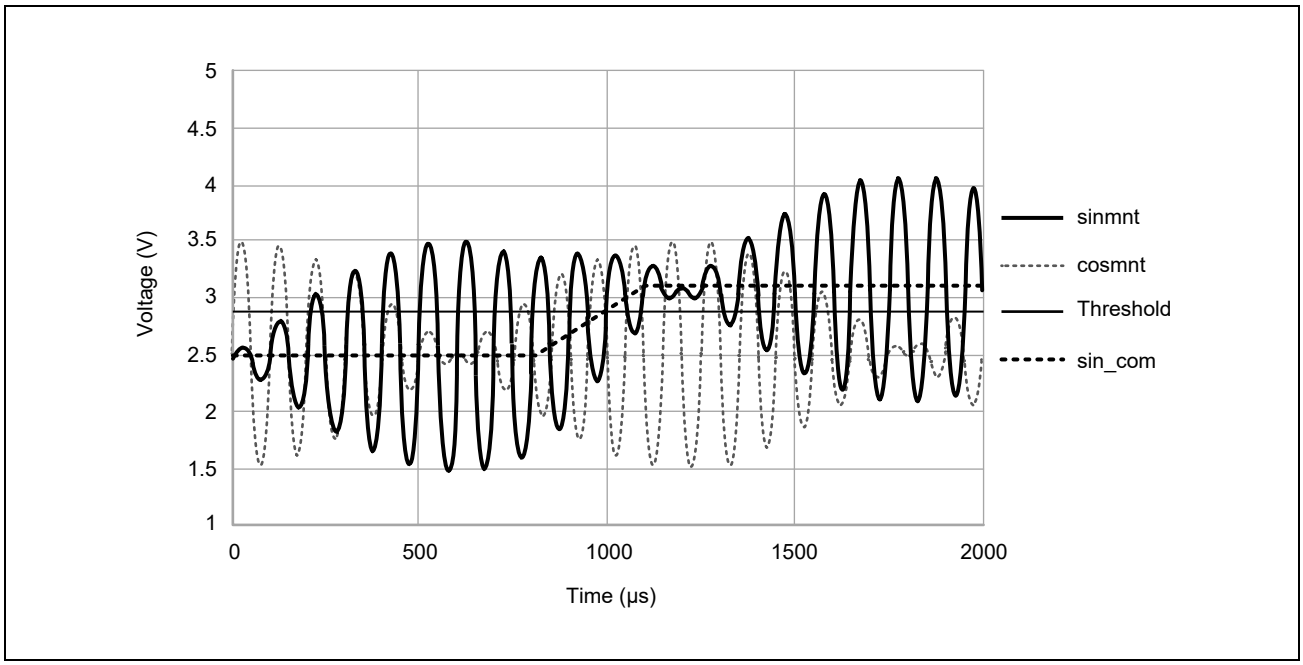


Figure 26.14 Monitor Signal Waveforms on Occurrence of Sinmnt Signal Disconnection in VR Resolver

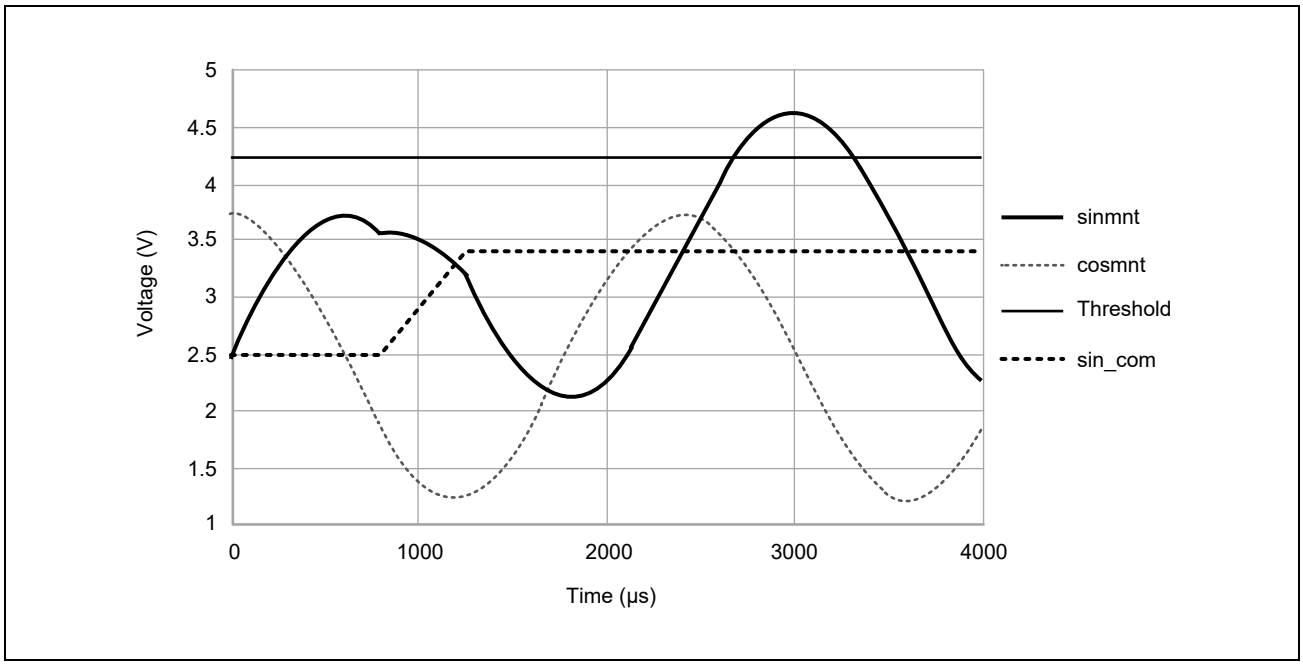


Figure 26.15 Monitor Signal Waveforms on Occurrence of Sinmnt Signal Disconnection in DC Resolver

26.4.4.4 R/D Conversion Error Detection

This function monitors the control variation in R/D conversion loop, and detects operation errors in the R/D conversion function. When an R/D conversion error is detected, the RDC error interrupt request signal becomes high.

A control variation (ε) is considered excessive if the control variation rises above or falls below a configured threshold level. For a description of the relationship between register values and threshold values, see **Section 39.5.3, Error Detect Characteristics**.

An R/D conversion error is detected if the control variation stay excessive for more than 50% of the error determination time set in the EDPS[1:0] bits in the RDC3AnDIAG1 register.

The R/D conversion error detection circuit includes a circuit supporting high-speed rotation of a resolver and a circuit not supporting it. These circuits can be selected by using the CVEDS bit in the RDC3AnDIAG1 register.

CVEDS = “0”: A circuit supporting high-speed rotation is selected. However, this setting is not allowed when the excitation frequency is set to 22 kHz or above.
When this setting is selected, set EDPS[1:0] to 10 or 11.

CVEDS = “1”: A circuit not supporting high-speed rotation is selected.

26.4.4.5 Two Paths Comparison Conversion Error Detection

This function compares the results of angle conversion from two loops, and detects conversion errors and failures in the circuit. In the angular conversion mode 0 (ADRD = 0), this function compares the phi angles output from the angular conversion mode 0 path and that from the angular conversion mode 1 path, and if the difference between two angles is larger than the threshold, this is judged to be a two paths comparison conversion error. The threshold value is set by the P2ANT[1:0] bits.

26.4.4.6 Resolver Signal Power/Ground Short Error Detection

This function detects power short errors (short circuits to the power supply) and ground short errors (short circuits to the ground) of the resolver signal lines RDC3AnRSO, RDC3AnCOM, RDC3AnS1, RDC3AnS2, RDC3AnS3, and RDC3AnS4. Note that if these errors are on the RDC3AnRSO and RDC3AnCOM pins, they are not correctly detected when the internal excitation signal is in use (EXIO = 1, SENS = 1) due to a conflict between the current from the short to the power supply or ground and the excitation current from the buffer. When an external excitation signal is used (EXIO = 0), errors on those pins are detected.

Detection is also possible during an angle conversion and the angular tracking is continued even while detection of power/ground short error is running (for 78 µs). The timing to start this function is selected by the VGASL[1:0] bits as follows;

- VGASL[1:0] = 00 (default)
  No detection performed.
- VGASL[1:0] = 01
  When the VGST bit is set to 1.
- VGASL[1:0] = 10
  Performed automatically every 10 ms.
  When this setting is selected, set the CVEDS bit to 1 to select a circuit not supporting high-speed rotation.
- VGASL[1:0] = 11
  No detection performed.

When the detection of power/ground short errors starts, the following operations are automatically performed by the circuit.

[Operation Sequence]

1. The RDC3AnS1, RDC3AnS2, RDC3AnS3, and RDC3AnS4 pins are disconnected from the PGA input circuit (**Figure 26.16**).
2. Each of the six resolver signal lines RDC3AnCOM, RDC3AnRSO, RDC3AnS1, RDC3AnS2, RDC3AnS3, RDC3AnS4 is selected, in sequence and in that order, for 13 µs*[1]. Detection finishes after all six signals have been selected, one after another, in a total of 78 µs.
3. The RDC3AnS1, RDC3AnS2, RDC3AnS3, and RDC3AnS4 pins are connected to the PGA input circuits.
4. The RDC enters the forced gain control state (for approx. 5 ms) and the angle is tracked.
   (If SAGD is set to 1, the RDC does not enter the forced gain control state.)

Note 1. The timing value is when the CCLK is running at 40 MHz.

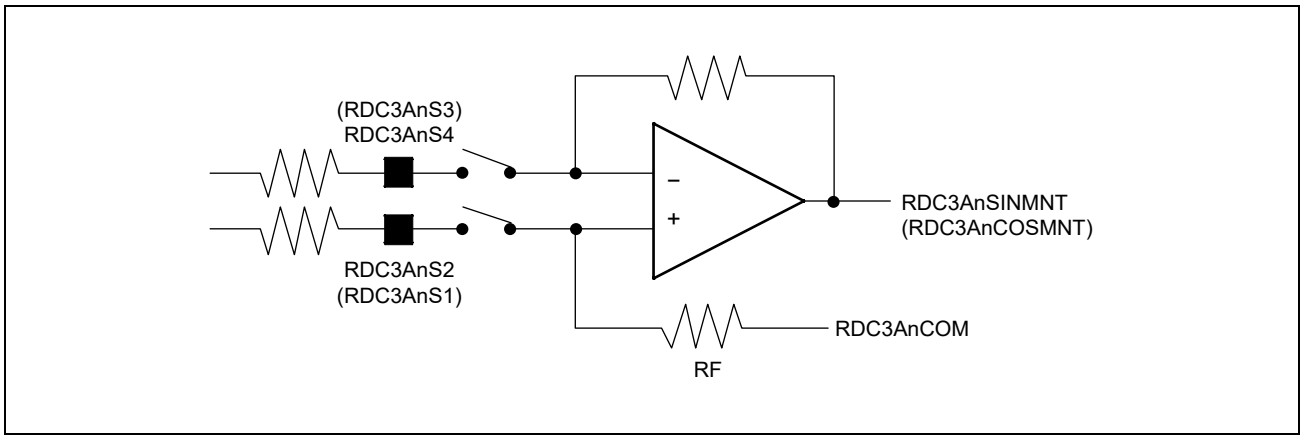


Figure 26.16 RDC3AnS1, RDC3AnS2, RDC3AnS3, RDC3AnS4 Pins on Detection of Power/Ground Short Error

While the detection of power/ground short errors is running (for 78 µs), the RDC3AnS1, RDC3AnS2, RDC3AnS3, RDC3AnS4 pins are disconnected from the PGA input circuit.

26.4.4.7 Sum-of-Squares Amplitude Error Detection

This function detects modulation, distortion and noise in the amplitudes of the sine and cosine signals input from the the resolver. This function is only available for resolvers requiring the input of signals with excitation components, but not for DC resolvers, which do not require this.

The sum-of-squares of the monitored signals (SINMNT, COSMNT) acquired in the ADC within the RDC are taken and integrated within the excitation period. Upper and lower threshold values are set for the value thus calculated and the number of times the calculated value rise above or fall below the thresholds is counted. If the number exceeds the threshold for the counted value, it is output as a sum-of-squares amplitude error.

The procedure runs automatically, but the user is required to set the upper and lower threshold for the integrated sum-of-squares values in the SQHTH and SQLTH bits and the threshold for the number of times the calculated value falls outside the range in the SQCTH bits. The counter values are cleared at the desired time by the writing to the SQERST bit.

Setting the ERSQS bit to 0 enables the detection of sum-of-squares amplitude errors. After setting this bit to 0, clear the counted value by setting the SQERST bit to 1.

In checking for a power/ground short error, the excitation amplitude disappears for 78 µs, so the resulting error is counted once in sum-of-squares amplitude error detection.

When checking for a power/ground short error by writing to the VGST bit when VGASL[1:0] = 01, set the ERSQS bit to 1 to switch sum-of-squares amplitude error detection off during the check.

When automatic checking for power/ground short errors is to proceed at 10-ms intervals because the setting of VGASL[1:0] = 10, clear the effect of this on the counting of sum-of-squares amplitude errors at least once per 10 ms by using the SQERST bit.

If you want to check for power/ground short errors but do not require checking of sum-of-squares amplitude errors, leave the value of ERSQS as 1.

Figure 26.17 shows an example of waveforms where the amplitudes of the signals input to the resolver are reduced, the calculated value fell outside the threshold range three times, and a sum-of-squares amplitude error is generated.

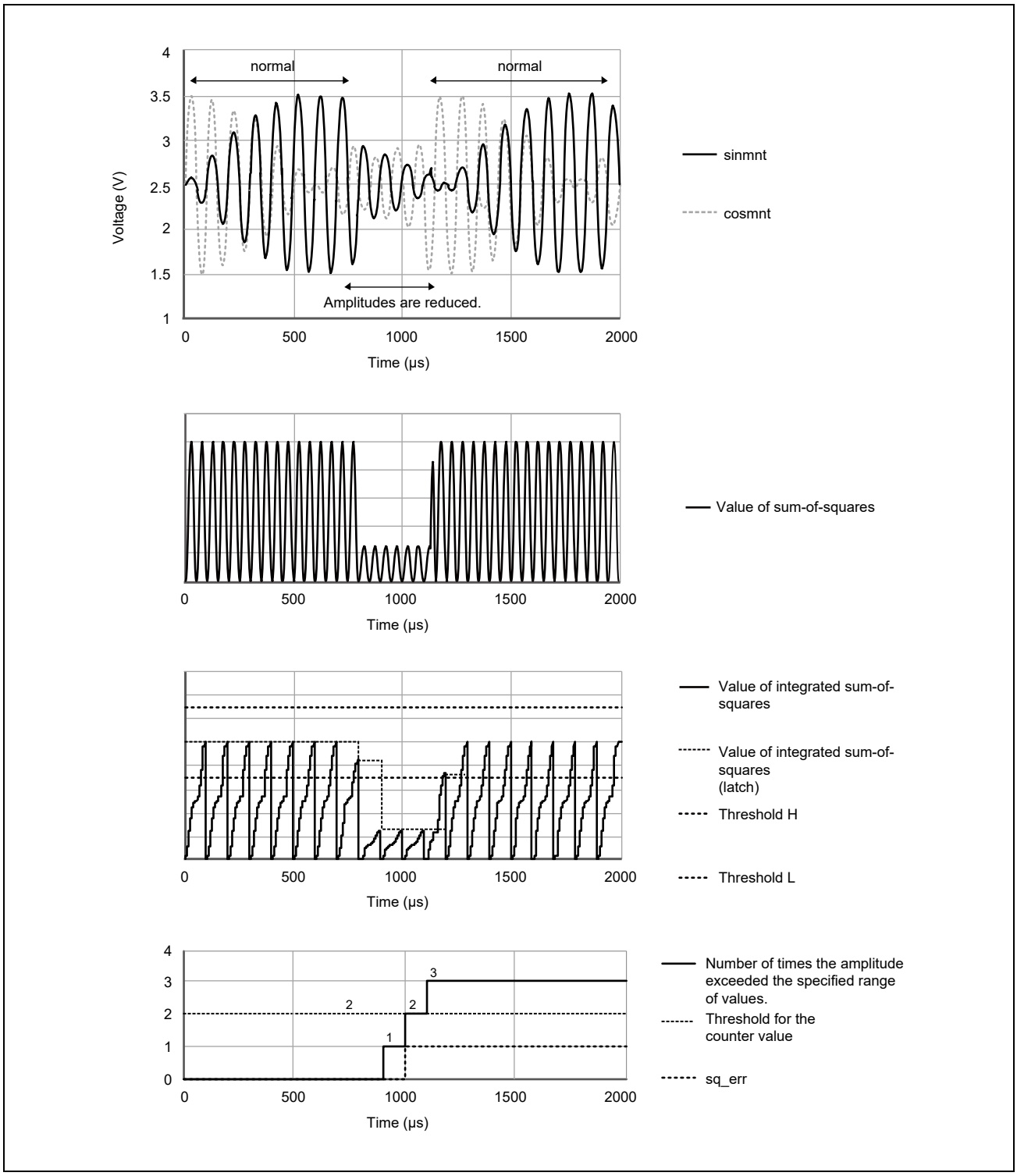


Figure 26.17 Example of Waveforms with Sum-of-Squares Error

26.4.5 Self-Diagnosis

26.4.5.1 Built-in Self-Test Function

In order to check the validity of a given operation, the Built-in Self-Test (BIST) function generates an intended signal input that is simulated internally by setting a BIST instruction in the RDC3AnBIST1 register, and monitors the signal that is output in response to the simulated signal input. **Table 26.64** lists test items.

Each output during the execution of BIST operates in response to the simulated signal.

Perform the following settings to execute BIST.

(1) Enable the forced gain control function during BIST operation. (Set the AGCD bit in the RDC3AnPI1 register to 0.)

(2) When the BIST is completed, set the ERRST bit in the RDC3AnDIAG1 register to 1 to reset the error signal.

Table 26.64 BIST Instructions

Item	Diagnosis
Angle conversion BIST	Self-test of the R/D conversion function The following electrical angles can be set as a resolve signal input: • Target angle 0° • Target angle 45° • Target angle 270°
Error detection BIST	Self-test for the error detection function • Resolver signal error detection BIST • Resolver signal disconnect error detection BIST • R/D conversion error detection BIST • Power/ground short error detection BIST • Square-sum amplitude error detection BIST
ADBIST	Self-test for the validity of the result of 12-bit SAR-ADC conversion • Apply voltages of 1, 2.5, and 4 V in order and judge the result of ADC conversion. • The threshold value for determination is set by the ADB3TH[1:0] bits.

Each output during the execution of BIST operates in response to the simulated signal. Depending on the BIST, an RDC error interrupt is generated due to the erroneous internal state. If the occurrence of this error disturbs the operation, disable RDC error interrupt by the EINTEN bit.

For the types of BIST which can be executed during angle conversion, tracking of the angle for the resolver input signal continues during execution of the BIST.

The angle output during the types of BIST which can only be executed during power up does not match the resolver input signal during execution of the BIST.

The BISTs are categorized into two groups depending on their execution timing as follows;

- Execution is possible during angle conversion and when starting up the power (short-period BIST): ADBIST, resolver signal error detection BIST, resolver signal disconnect detection BIST, power short error BIST, ground short error BIST, sum-of-squares amplitude error detection BIST (high side), sum-of-squares amplitude error detection BIST (low side)
- Execution is possible when starting up the power: angle conversion BIST, conversion error BIST

Perform the following settings to execute BIST.

(1) Enable the forced gain control function when executing a BIST. (Set the AGCD bit in the RDC3AnPI1 register to 0.)

(2) To execute a short-period BIST, set the EINTEN bit to 0 to disable error interrupts.

(3) Set the VGASL[1:0] bits to 00_B when executing a BIST.

(4) Read the VGFLG bit (Power/ground short error detection running bit) as 0. If the value is 1, read the bit again after waiting for about 80 µs.

(5) Set the phase correction bits in the sine and cosine angle correction register to 0° when executing the angle conversion BIST.
Sine phase correction bits SINPO[11:0] = 000_H
Cosine phase correction bits COSPO[11:0] = 000_H

(6) To execute conversion error BIST, set the EDPS bit to 10_B.

(7) Set the input gain resistance value at default (21 kΩ) when executing a BIST.
IRSS1 = 0, IRSS0 = 1, IRSC[3:0] = 0100_B

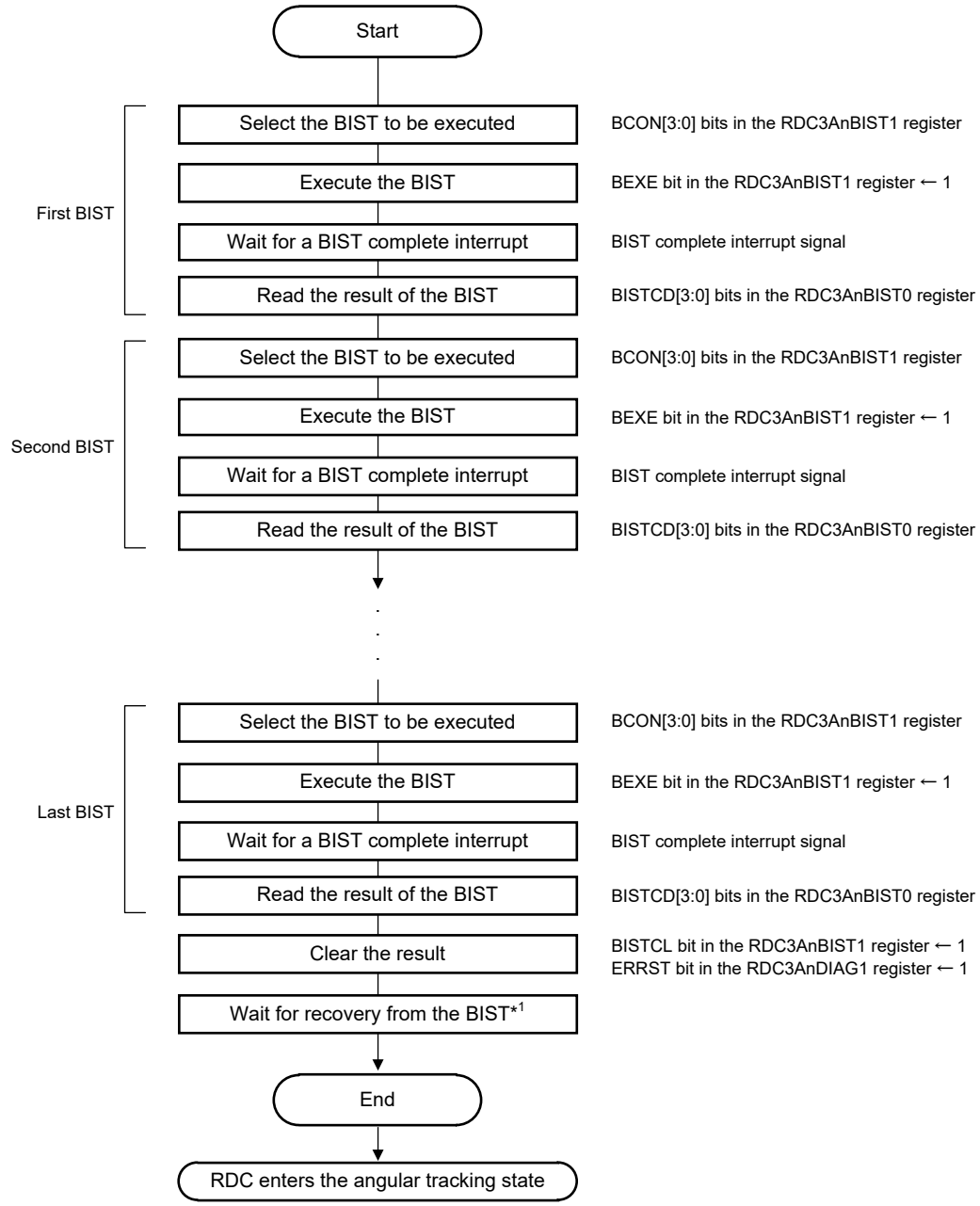
(8) Clear the count value in the SQERST bit by setting 1 before runningsum-of-squares amplitude detection BIST (high and low sides).

(9) When the BIST is completed, set the ERRST bit in the RDC3AnDIAG1 register to 1 to reset the error signal. Return the settings of the registers which were changed in steps (1), (2), (3), (5), (6), and (7) to their original values.

Table 26.65 lists the BISTs to be executed depending on the setting of values.

Table 26.65 BISTs for Each Setting of Values

BCON[3:0]	BIST to be Executed
0000	BEXE bit is disabled.
0001	This setting is not allowed.
0010	Square-sum amplitude error detection BIST (low side)
0011	Square-sum amplitude error detection BIST (high side)
0100	ADBIST
0101	Angle conversion BIST (0°)
0110	Angle conversion BIST (45°)
0111	Angle conversion BIST (270°)
1000	This setting is not allowed.
1001	Error detection BIST: resolver signal error detection BIST
1010	Error detection BIST: resolver signal disconnect detection BIST (cosine side)
1011	Error detection BIST: resolver signal disconnect detection BIST (sine side)
1100	Error detection BIST: conversion error BIST
1101	Error detection BIST: power short error BIST
1110	Error detection BIST: ground short error BIST
1111	This setting is not allowed.



Note: In the conversion error BIST, write 1 to the KIRST bit after reading the result.
If the type of BIST to be executed is conversion error BIST in conversion mode 1 (ADRD bit = 1), execution of the XXth BIST must follow the sequence in Figure 26.19.

Note 1. See **Section 39.5.1, RDC Conversion Performance** for the recovery time from a BIST.

Figure 26.18 Sequence of Executing BISTs after Starting Up the Power

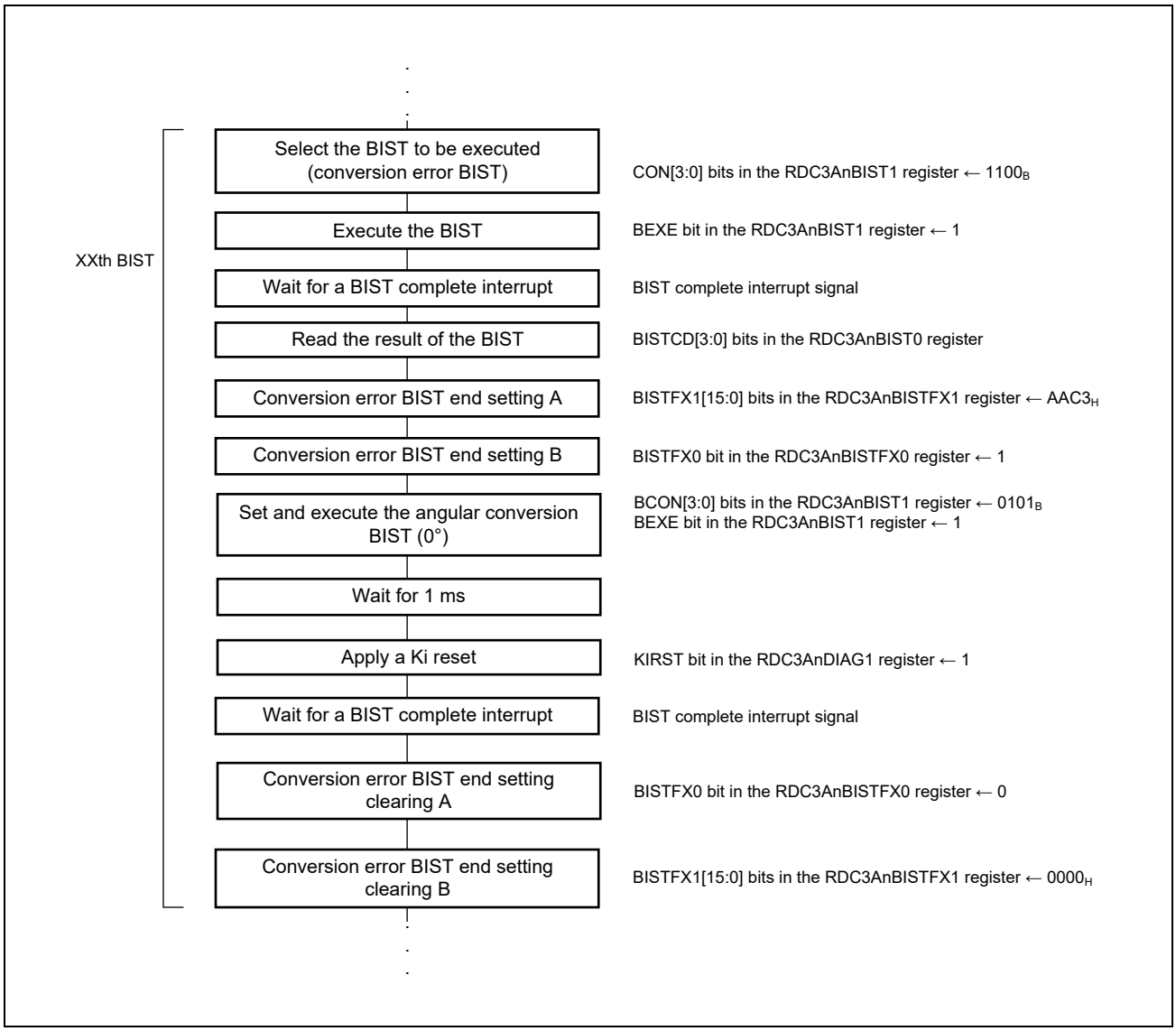
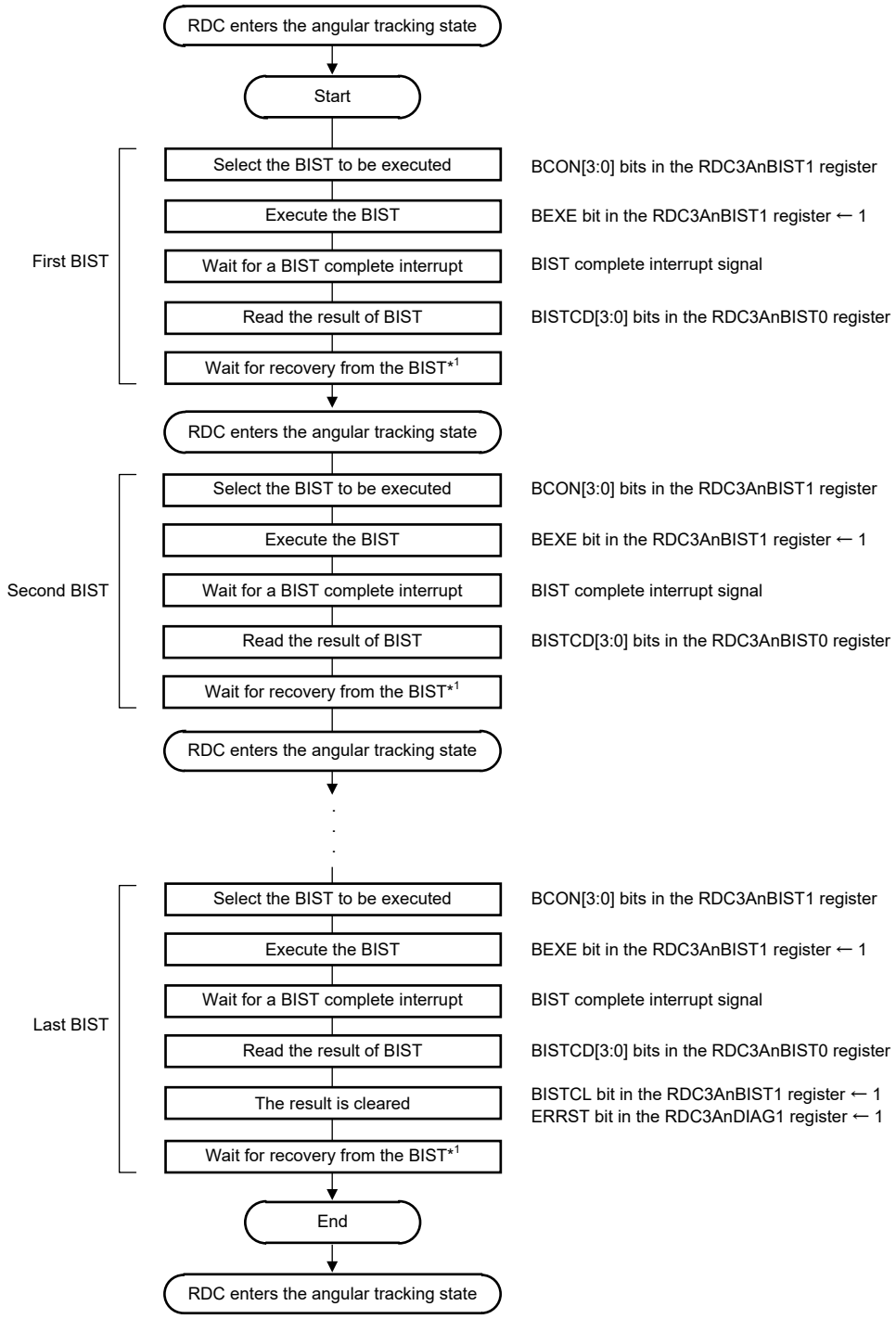


Figure 26.19 Sequence of Executing Conversion Error BIST in Angular Conversion Mode 1



Note: In the sequence of executing a BIST that is possible during angle conversion, tracking of the resolver continues even during execution of the BIST.

Note 1. See **Section 39.5.1, RDC Conversion Performance** for the recovery time from a BIST.

Figure 26.20 Sequence of Executing BISTs during Angle Conversion

26.4.5.2 ADC Software BIST

Failures in the ADC are diagnosed by the CPU through the following procedure: the DAC code in the 12-bit SAR-ADC written to the relevant register is converted in the ADC and the result is read from the relevant register.

The user can set the DAC codes freely, that is, to any of the 4096 codes for the 12-bit ADC. In the ideal state, the software will return the exact value of the DAC code which was set for the software BIST as the result of AD conversion. In actual operations, however, values with errors are output due to non-ideal factors. Therefore, users are required to set a suitable value for the CPU to judge these errors (around ± 32 LSB).

This BIST is capable of diagnosing the normality of combined operations for all DAC switches in the ADC. On the other hand, the ADBIST described in **Section 26.4.5.1, Built-in Self-Test Function**, diagnoses the results of conversion by applying three voltages to the input nodes of the ADC. In summary, failures in all nodes of the 12-bit ADC are diagnosed by using the ADBIST and software BIST in combination.

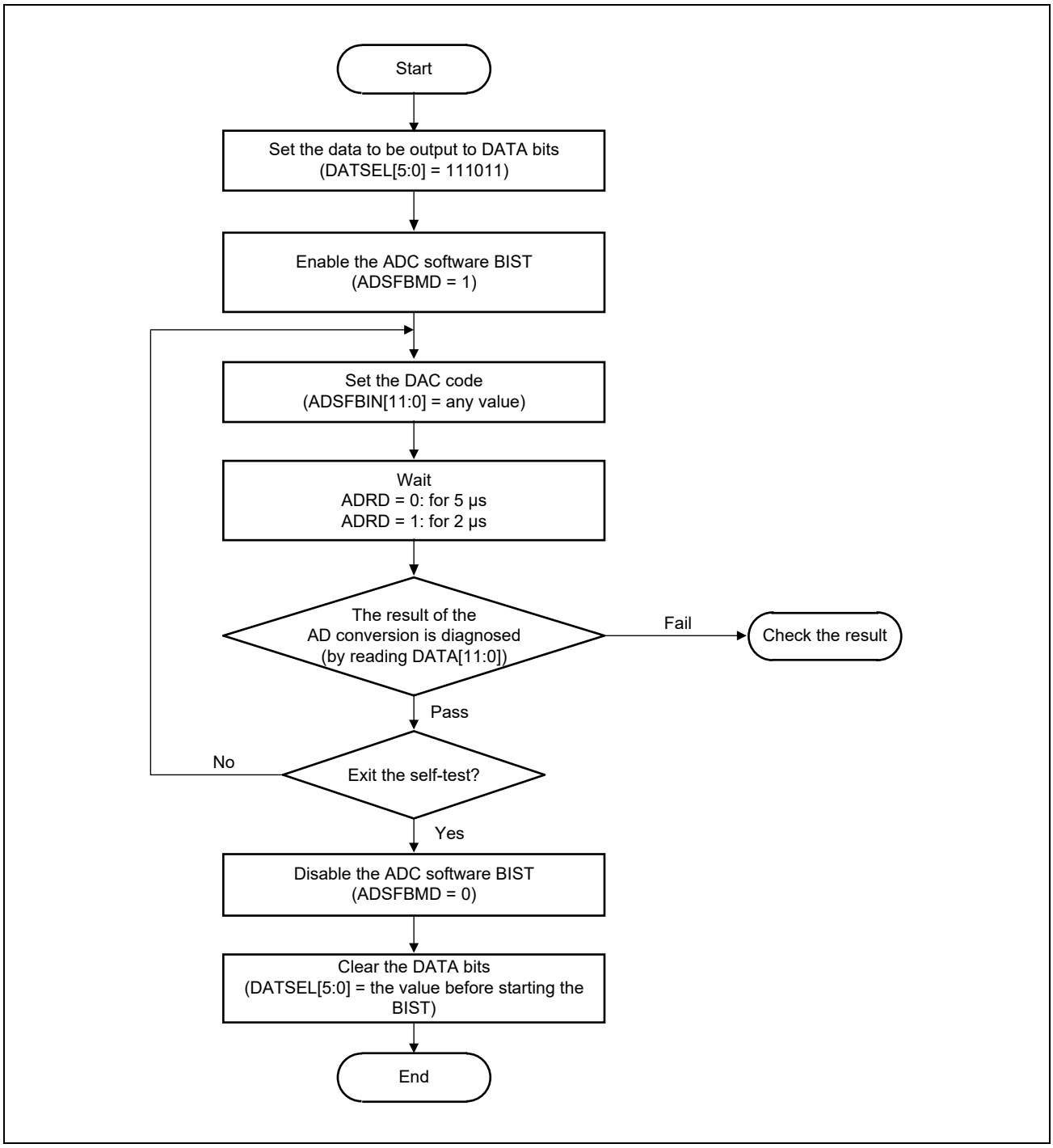


Figure 26.21 Operating Sequence of the ADC Software BIST

26.4.6 Excitation Timer (ET) Function

The excitation timer comprises two 16-bit timers: a period measurement timer and an event generation timer. The operating clock of the timers is CCLK (40 MHz). **Figure 26.22** shows a block diagram of the excitation timer.

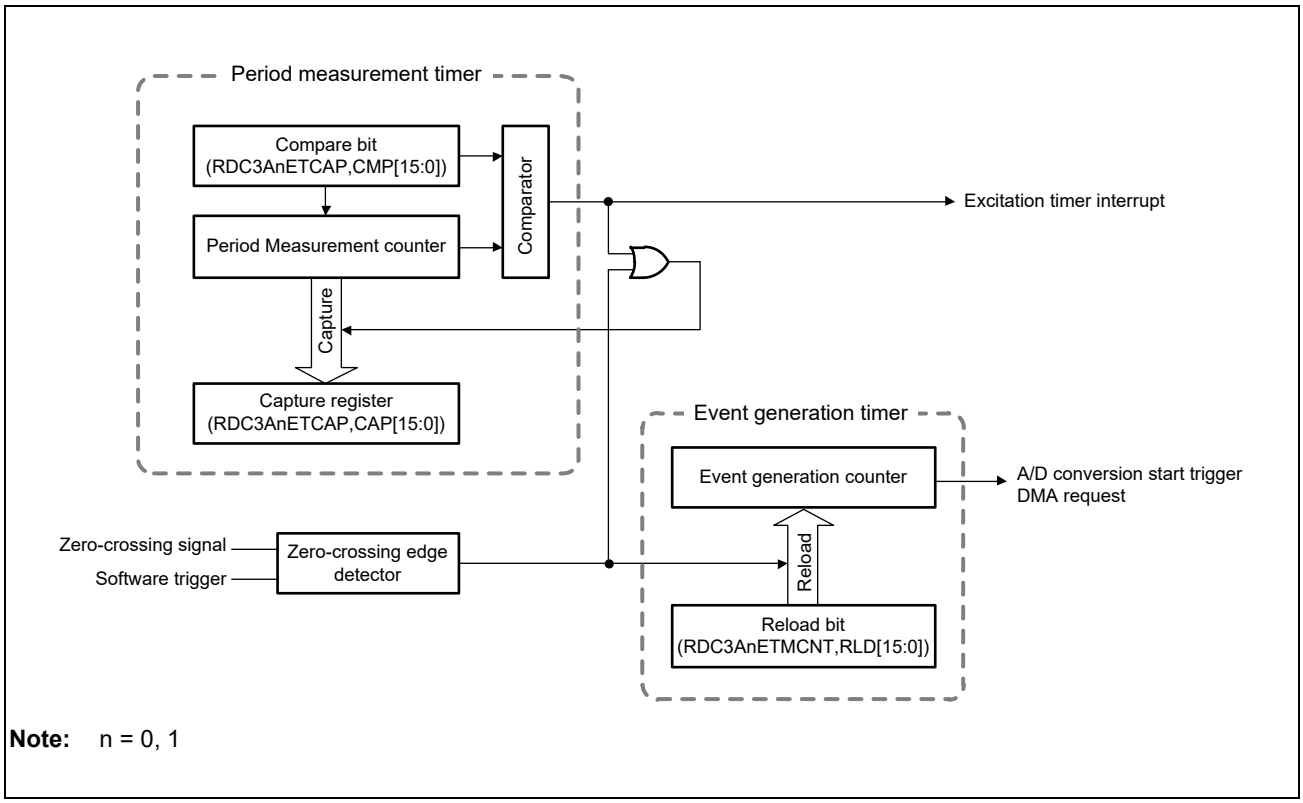


Figure 26.22 Block Diagram of Excitation Timer

26.4.6.1 Period Measurement Timer

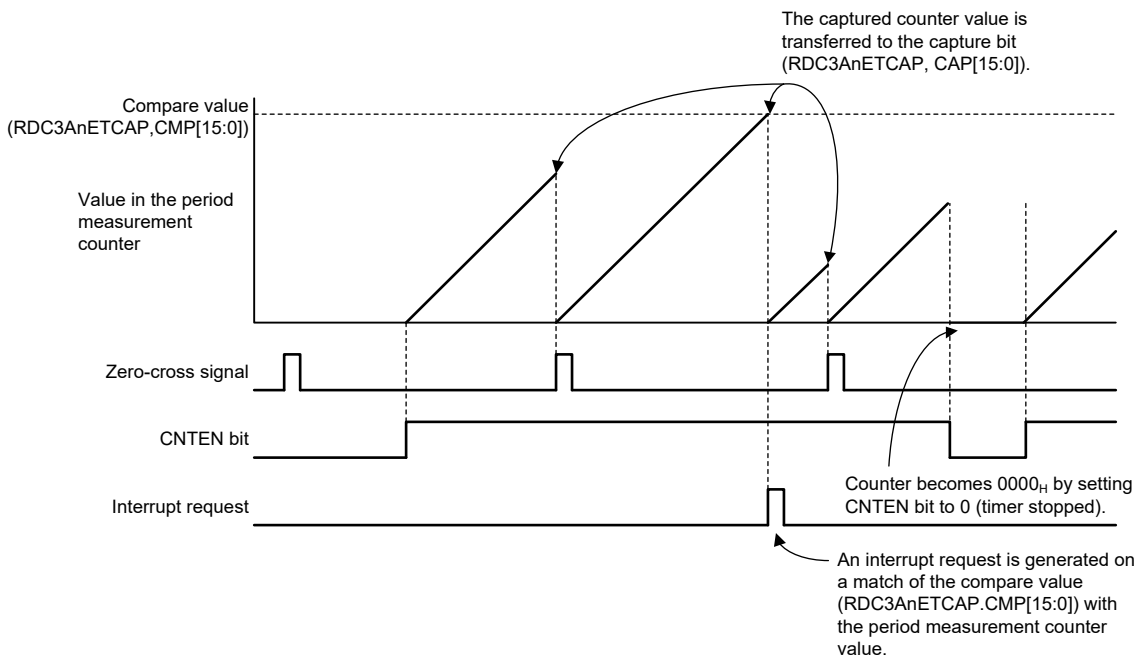
The period measurement timer measures the cycle of excitation signal (zero-crossing signal). When an edge (selectable from rise edge and fall edge) of the zero-crossing signal is detected, the value of the period measurement counter is captured and stored in the RDC3AnETCAP register. By reading the RDC3AnETCAP register, the cycle of excitation signal can be obtained.

The cycle of excitation signal can be calculated from the following formula: (RDC3AnETCAP register value + 1) × CCLK cycle (25 ns).

When the IREN bit in the RDC3AnETEN register is set to 1 (enables the interrupt), an excitation timer interrupt request is generated if the value set in the RDC3AnETCAP register matches the period measurement counter value. The excitation signal cycle error can be detected by setting a value of longer duration than the excitation signal cycle to the RDC3AnETCAP register.

When a zero-crossing signal trigger is generated by writing 1 to the ZCSTRG bit in the RDC3AnETEN register, the period measurement timer operates in the same way as the zero-crossing signal edge detection.

Figure 26.23 shows an example of period measurement timer operation.



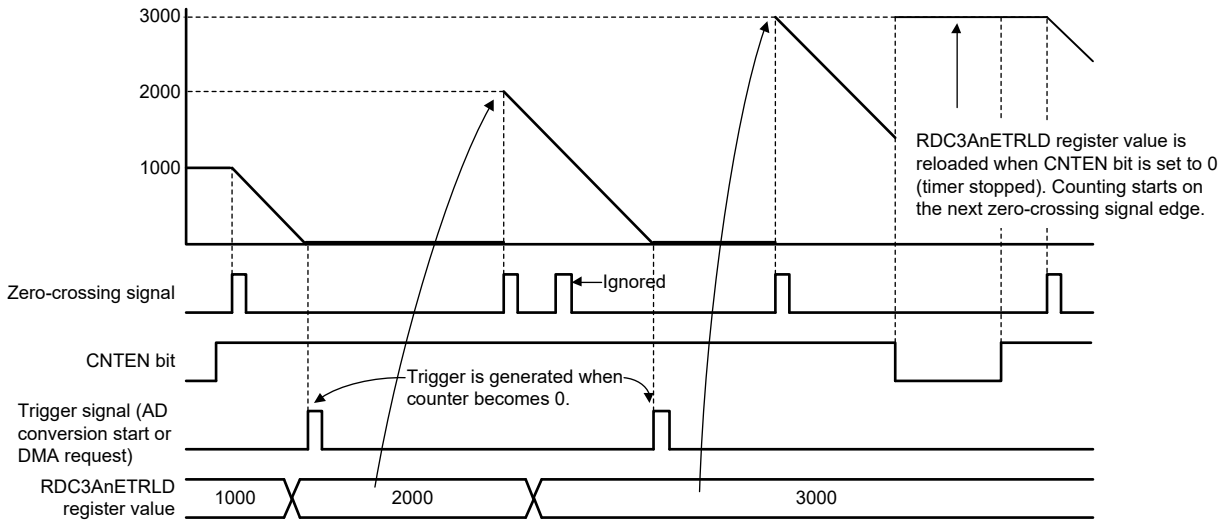
Note: n = 0, 1
CNTEN: A bit in the RDC3AnETEN register

Figure 26.23 Example of Period Measurement Timer Operation

26.4.6.2 Event Generation Timer

The event generation timer can generate a trigger signal (A/D conversion trigger, DMA request) after the time set in the RDC3AnETMCNT register has elapsed since the occurrence of an edge of the zero-crossing signal. When a zero-crossing signal trigger is generated by writing 1 to the ZCSTRG bit in the RDC3AnETEN register, the event generation timer operates in the same way as the zero-crossing signal edge detection.

Figure 26.24 shows an example of event generation timer operation.



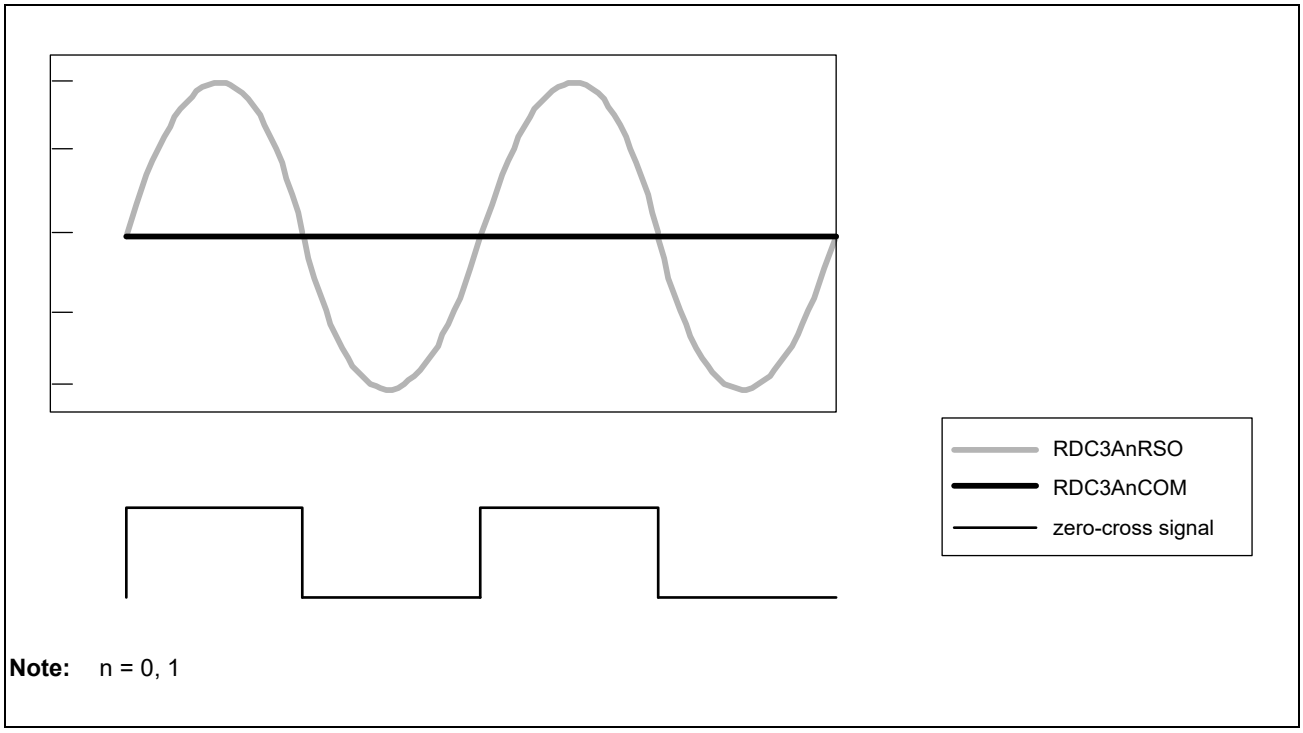
Note: n = 0, 1
CNTEN: A bit in the RDC3AnETEN register

Figure 26.24 Example of Event Generation Timer Operation

26.4.6.3 Excitation Zero-Crossing Signal

The zero-crossing signal indicates zero-crossing timing for differential excitation signal input. The zero-crossing signal is input to the excitation timer circuit to be used for detecting the excitation vertex.

Figure 26.25 shows the relationship between the differential excitation signal input waveforms and zero-crossing signal.



Note: n = 0, 1

Figure 26.25 Relationship between Differential Excitation Signal Input (RDC3AnRSO, RDC3AnCOM) and Zero-Crossing Signal

26.4.7 PGA Inversion

When the resolver angle is near 135° or 315°, the sine and cosine signals output through the PGA shapes are vertically symmetrical to each other relative to a common potential. The input potentials of the two signals acquired by the 12-bit SAR-ADC are significantly different and are relatively strongly affected by the nonlinearity error of the ADC. The RDC is equipped with functionality to reduce this effect (by PGA inversion). The output voltage from the PGA is inverted in the range where the output phi angles whose input potentials of the sine and cosine signals become opposite to one another.

Whether to invert the sine and cosine waves or not is decided depending on the output phi angles as follows.

- Range of phi output angles = 0° to 90°: sin is not inverted, cos is not inverted
- Range of phi output angles = 90° to 180°: sin is not inverted, cos is inverted
- Range of phi output angles = 180° to 270°: sin is inverted, cos is inverted
- Range of phi output angles = 270° to 360°: sin is inverted, cos is not inverted

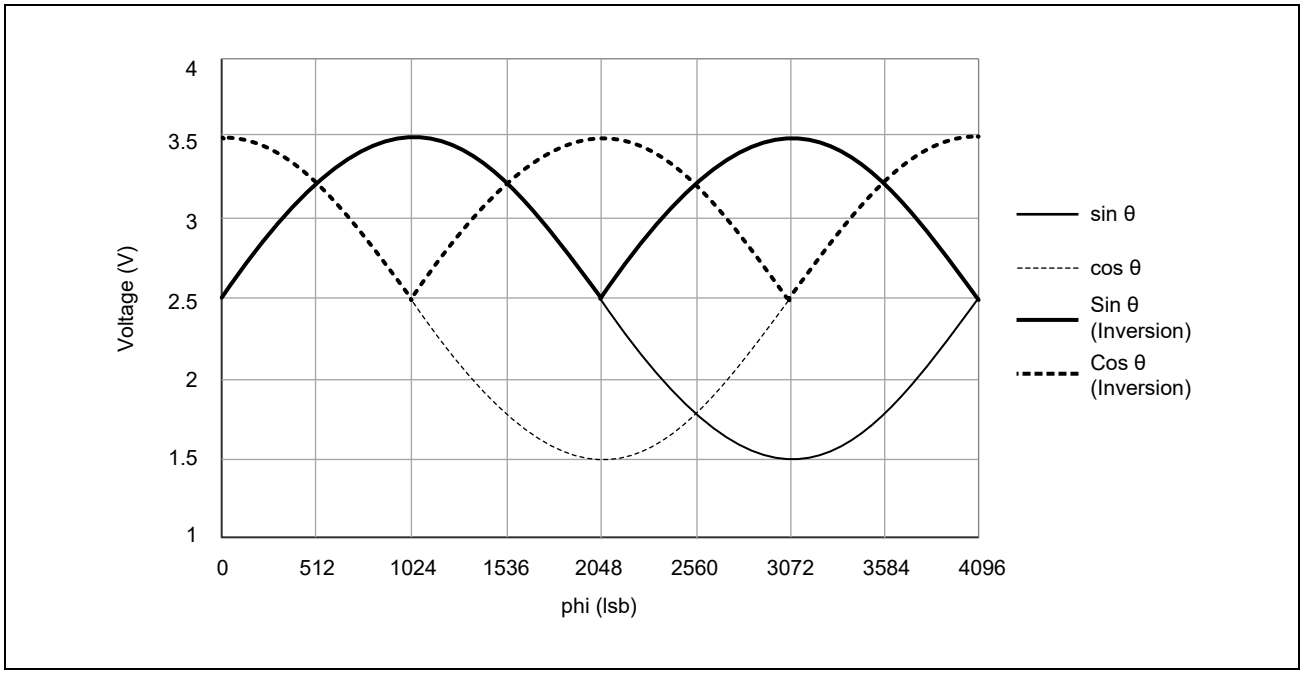


Figure 26.26 Output Phi Angle and Sine and Cosine Signals when Inversion is Enabled

For example, in the case where the resolver stays very close to 90° (1024 LSB), the output phi angle rises and falls at 90°. Here, the cosine signal output through the PGA becomes unstable due to repeated inversion and non-inversion. To avoid this, hysteresis can be added to the thresholds for the range of output phi angles where PGA inversion is applied.

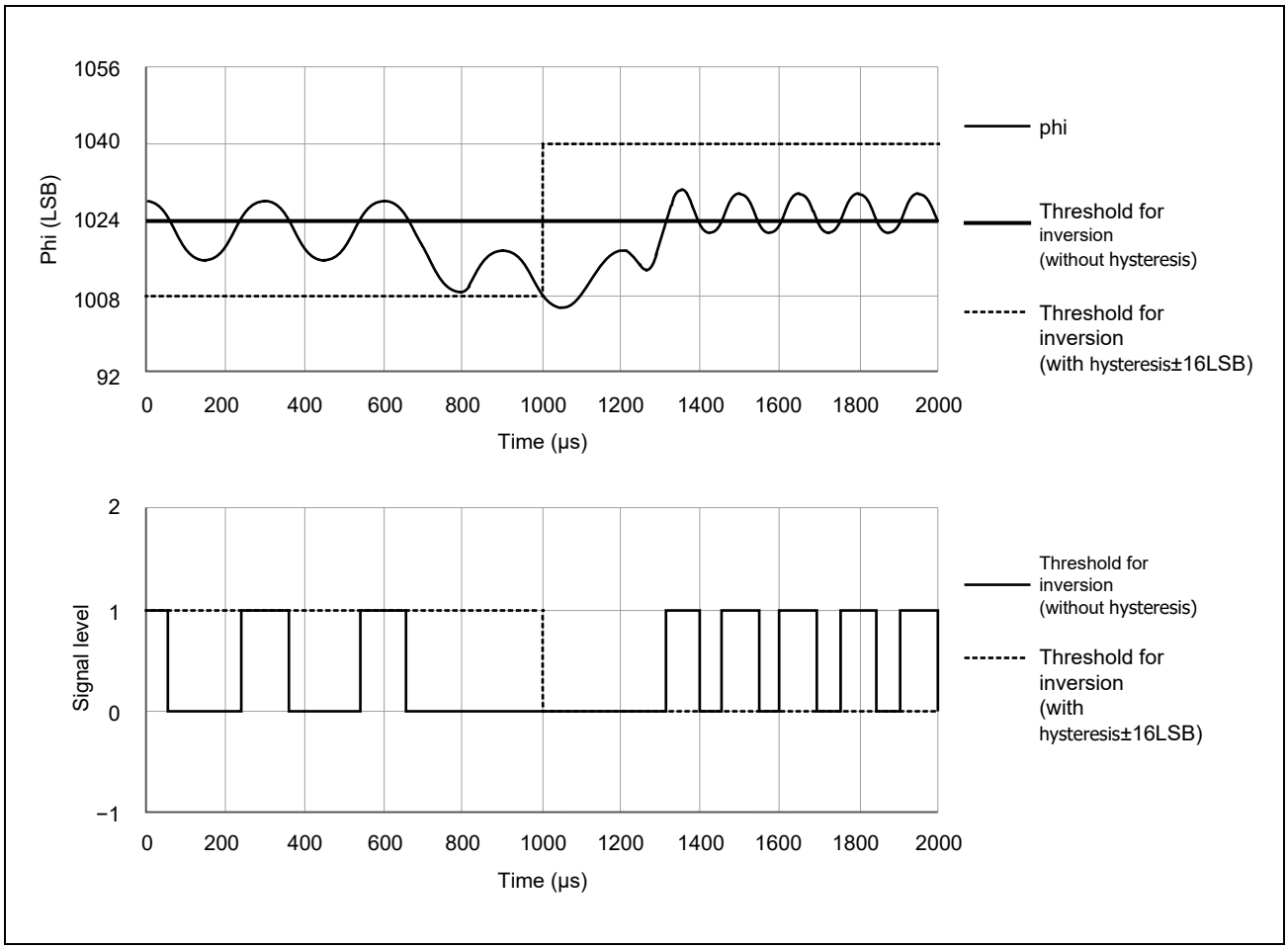


Figure 26.27 Comparison of Thresholds with and without Hysteresis for the Angle with PGA Inversion

The timing of PGA inversion is selected by the PGAIVSL[1:0] bits.

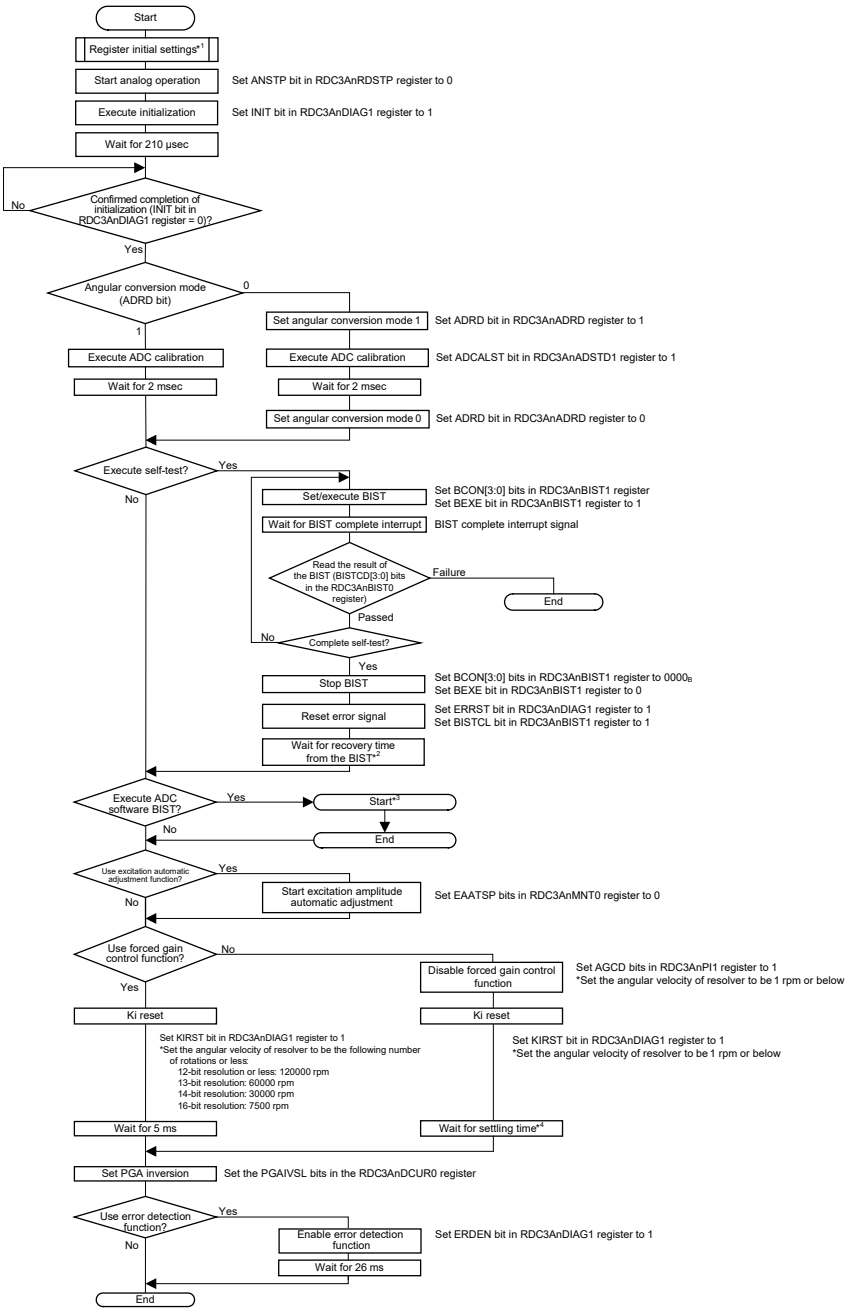
- PGAIVSL[1:0] = 00: no inversion (default)
- PGAIVSL[1:0] = 01: the output phi angle falling outside the thresholds for switching between inversion and non-inversion, followed by an excitation zero-crossing.
- PGAIVSL[1:0] = 10: the output phi angle falling outside the thresholds for switching between inversion and non-inversion, followed by input to the ADC being switched between the sine and cosine signals.
- PGAIVSL[1:0]=11: the output phi angle falling outside the thresholds for switching between inversion and non-inversion, followed by an excitation zero-crossing and input to the ADC being switched between the sine and cosine signals.

Set PGAIVSL[1:0] = 00 in angular conversion mode 0 (ADRD = 0).

To further improve the accuracy of angular conversion in angular conversion mode 1 (ADRD = 1), set PGAIVSL[1:0] to a value other than 00.

26.5 Initial Operation Procedure

Figure 26.28 shows the flow of initial operation of the RDC3A and **Figure 26.29** shows the flow of the initial settings for the registers.



Note 1. n = 0, 1

Note 2. If the angle cannot be tracked when the R/D unit is started, apply a ki reset after the amplitude of RDC3AnSINMNT or RDC3AnCOSMNT rises to at least 1 V p-p and the amplitude of RDC3AnRSO rises to at least 200 mV p-p.

Note 1. See **Figure 26.29, Flow of Initial Settings of the Registers**

Note 2. For the recovery time from a BIST, see **Section 39.5.1, RDC Conversion Performance**

Note 3. See **Figure 26.21, Operating Sequence of the ADC Software BIST**

Note 4. For the settling time, see **Section 39.5.1, RDC Conversion Performance**

Figure 26.28 RDC3A Initial Operation Flow

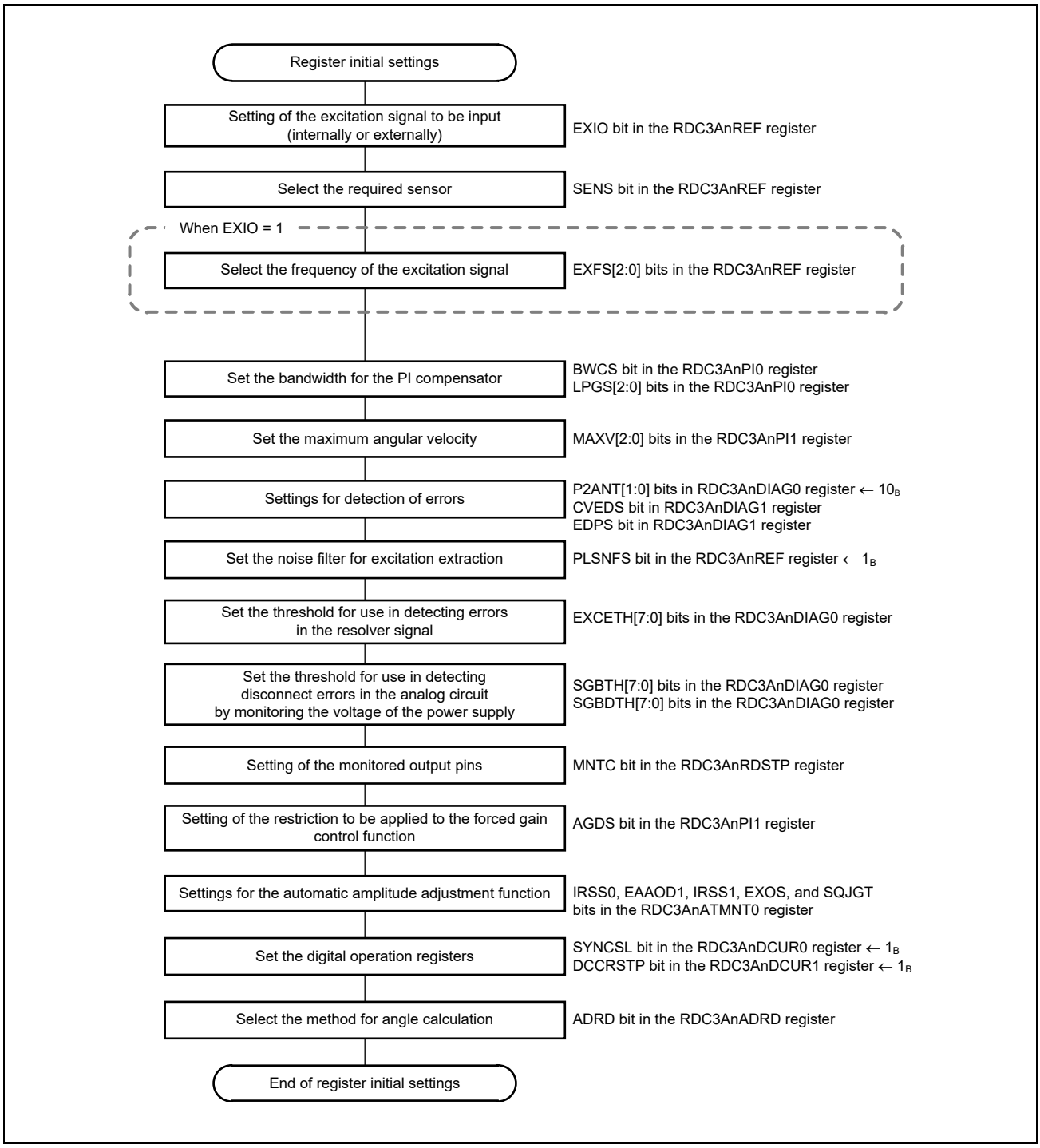


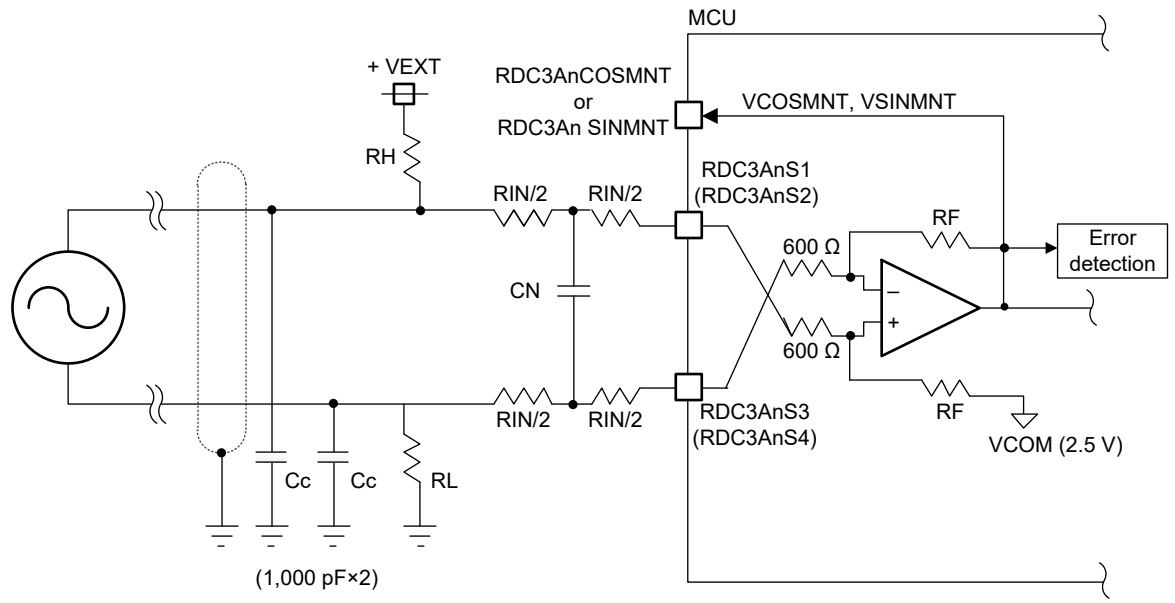
Figure 26.29 Flow of Initial Settings of the Registers

26.6 Resolver Interface Circuit

The following shows specific interface circuits as reference examples. When determining constants such as a resistance value and adding functions such as an input/output protection circuit, a careful decision must be made for each system and conduct adequate evaluation.

26.6.1 Resolver Signal Input (Differential Input) Circuit

Figure 26.30 shows a VR resolver signal input circuit and an equivalent circuit of monitor output.



Note: The input resistance (RIN) tolerance (%) can affect the conversion accuracy (LSB). Approximately 0.3% corresponds with 1 LSB.

VCOSMNT, VSINMNT: Resolver signal monitor voltage
VCOM: IC internal reference voltage (= RVCC/2)
RF: 21 kΩ (typ). Subject to register setting and automatic adjustment function.

Figure 26.30 VR Resolver Signal Input Circuit and Equivalent Circuit of Monitor Output

- RIN: Resolver signal level

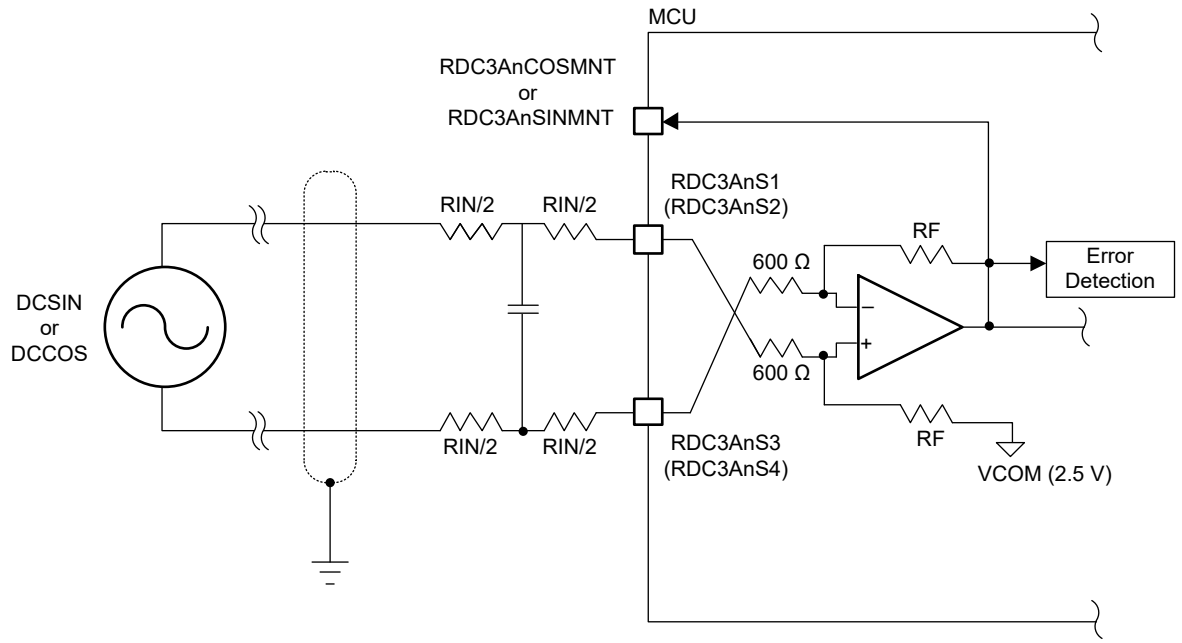
  Adjust the signal level so that VCOSMNT or VSINMNT = (VIN) × (RF / (RIN + 600 Ω)) falls within the range from 0.36 × RVCC to 0.64 × RVCC [Vpp].

  (where VIN denotes the signal output voltage between resolver pins [Vpp], RIN ≥ 2[kΩ])

- RH and RL: Determine a resistance value in an 89% to 100% range from the following calculated values:

  1. $RH \approx \{(RVCC - VCOM) / (22.0 \times 10^{-6})\} - RIN$, where VCOM = RVCC/2[V]
  2. $RL \approx \{VCOM/(22.0 \times 10^{-6})\} - RIN$, where VCOM = RVCC/2[V]

Figure 26.31 shows an equivalent circuit when the DC resolver signal input is used.



Note: The input resistance (RIN) tolerance (%) can affect the conversion accuracy (LSB). Approximately 0.3% corresponds with 1 LSB. Differences in input signal conditions and peripheral circuits can affect the RD conversion accuracy and the error detection function.

Figure 26.31 Equivalent Circuit when DC Resolver Signal Input is Used

26.6.2 Excitation Voltage Booster Amplifier Circuit

26.6.2.1 Excitation Voltage Booster Amplifier Circuit (Single Power Supply)

Figure 26.32 shows an equivalent circuit of the excitation voltage booster amplifier circuit with single power supply.

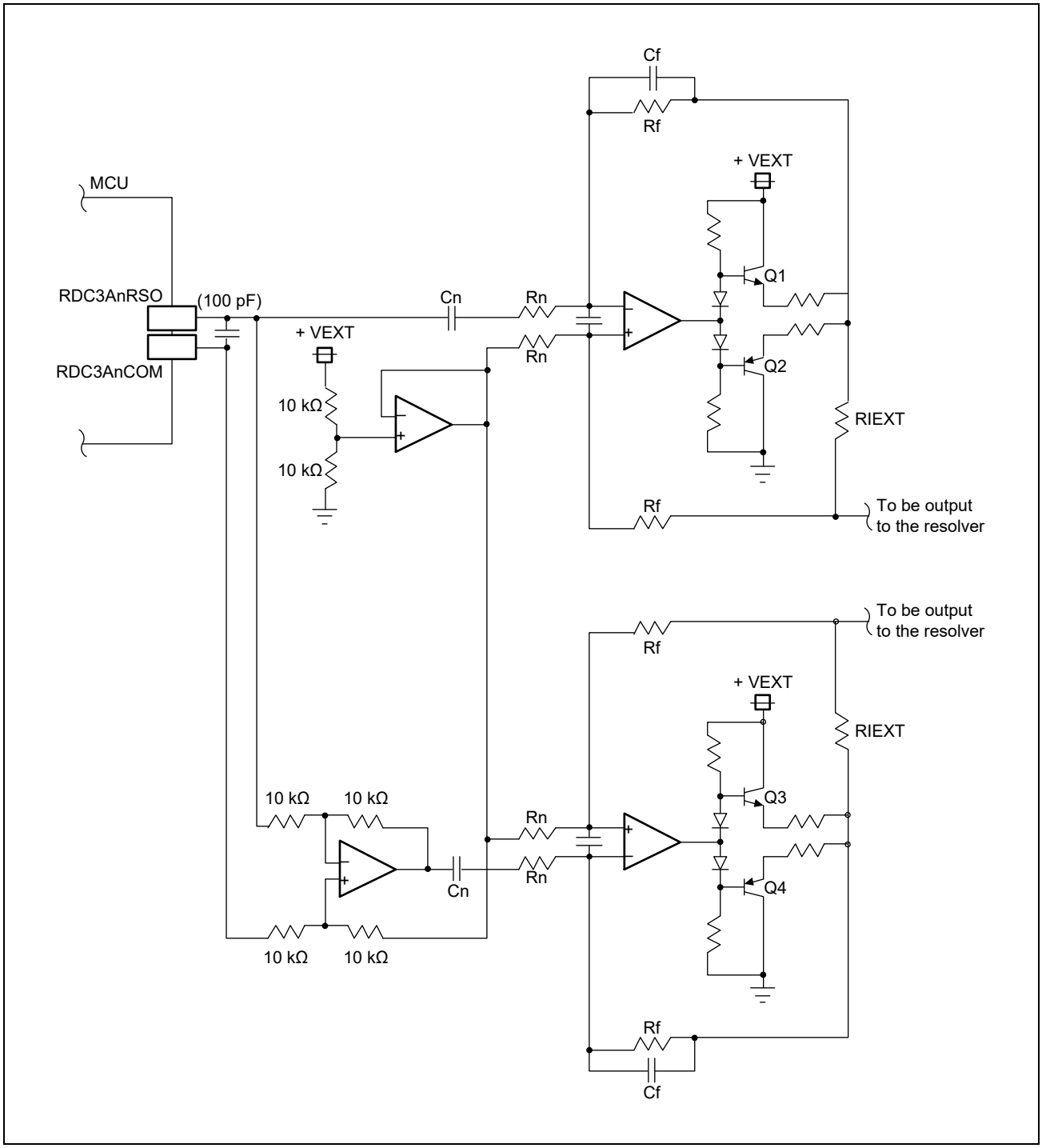


Figure 26.32 Equivalent Circuit Example of Excitation Voltage Booster Amplifier Circuit (Single Power Supply)

26.6.2.2 Excitation Voltage Booster Amplifier Circuit (Dual Power Supply)

Figure 26.33 shows an equivalent circuit of the excitation voltage booster amplifier circuit with dual power supply.

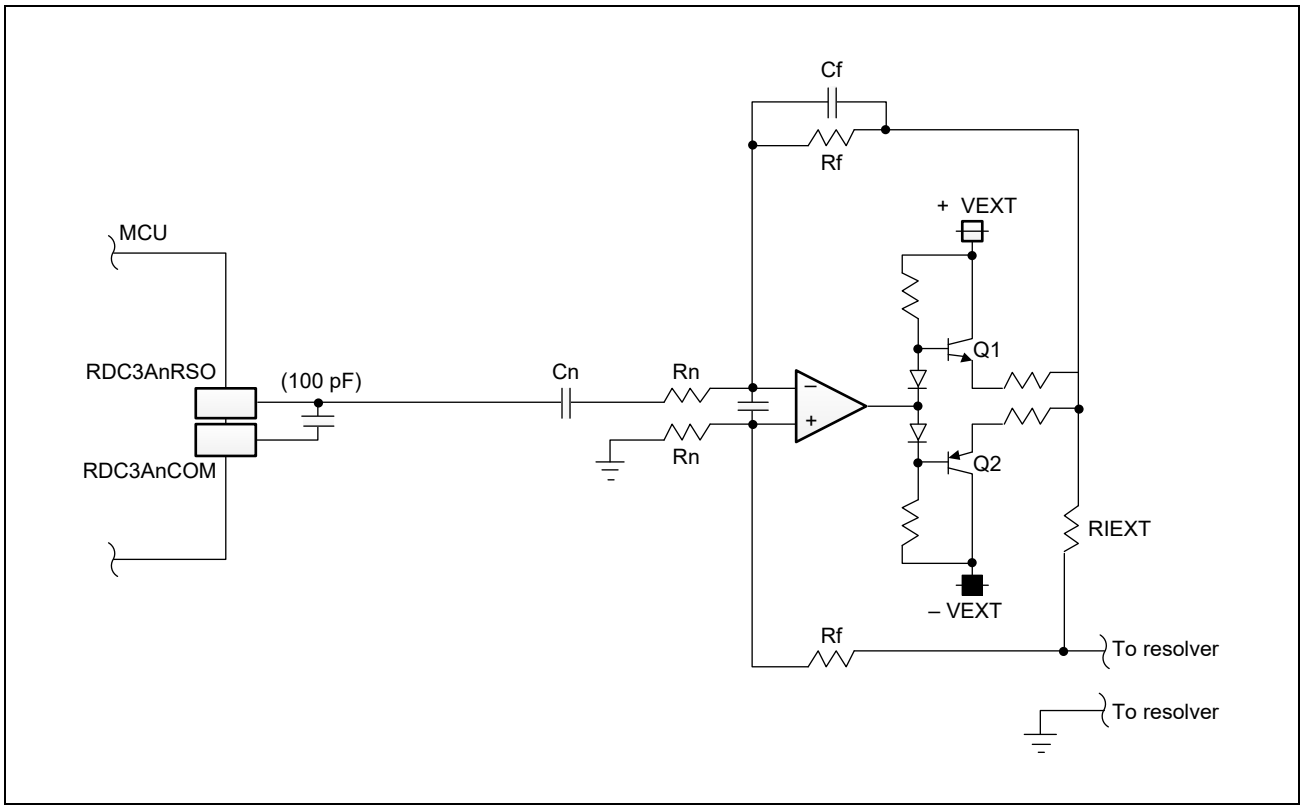


Figure 26.33 Equivalent Circuit Example of Excitation Voltage Booster Amplifier Circuit (Dual Power Supply)

26.6.2.3 Method for Setting Constants of Excitation Voltage Booster Amplifier Circuit

Both of the excitation voltage booster amplifier circuits shown as reference use the current control type. This type is effective in preventing secondary failure caused by short-circuit accident between excitation lines. Also, it is expected to improve the S/N ratio of resolver signal by boosting voltage.

Step (1): Calculate the excitation current by setting the excitation voltage based on the voltage of external power supply.

VREF = IREF × ZRO

Step (2): Calculate the circuit constants based on the excitation current.

IREF = (VRSO × Rf) / RIEXT × Rn)

[Legend]

+VEXT, – VEXT: External power supplies (for the excitation voltage booster amplifier circuit)

IREF : Excitation current of the resolver

RIEXT : Resistor for setting excitation current of the resolver

VREF : Excitation voltage for the resolver

ZRO : Input impedance of the resolver (specification value)

VRSO : RDC3AnRSO pin output voltage (= 2Vp-p)

<Settings conditions>

- RIEXT ≤ (ZRO / 10) [Ω]
- Rf ≥ 50 kΩ, Cn × Rn ≥ 5×10^{-4}[s], Cf × Rf ≤ 5×10^{-6}[s]
- Use the same power supply for an operational amplifier as that for the transistor buffer.

26.6.3 Resolver Excitation Signal External Input Method

26.6.3.1 Resolver Excitation Signal Input Circuit (Single Power Supply)

Figure 26.34 shows an equivalent circuit when the resolver excitation signal is input from an external source (single power supply). **Table 26.66** lists additional resistor values (reference values) of the resolver excitation signal external input circuit with a single power supply.

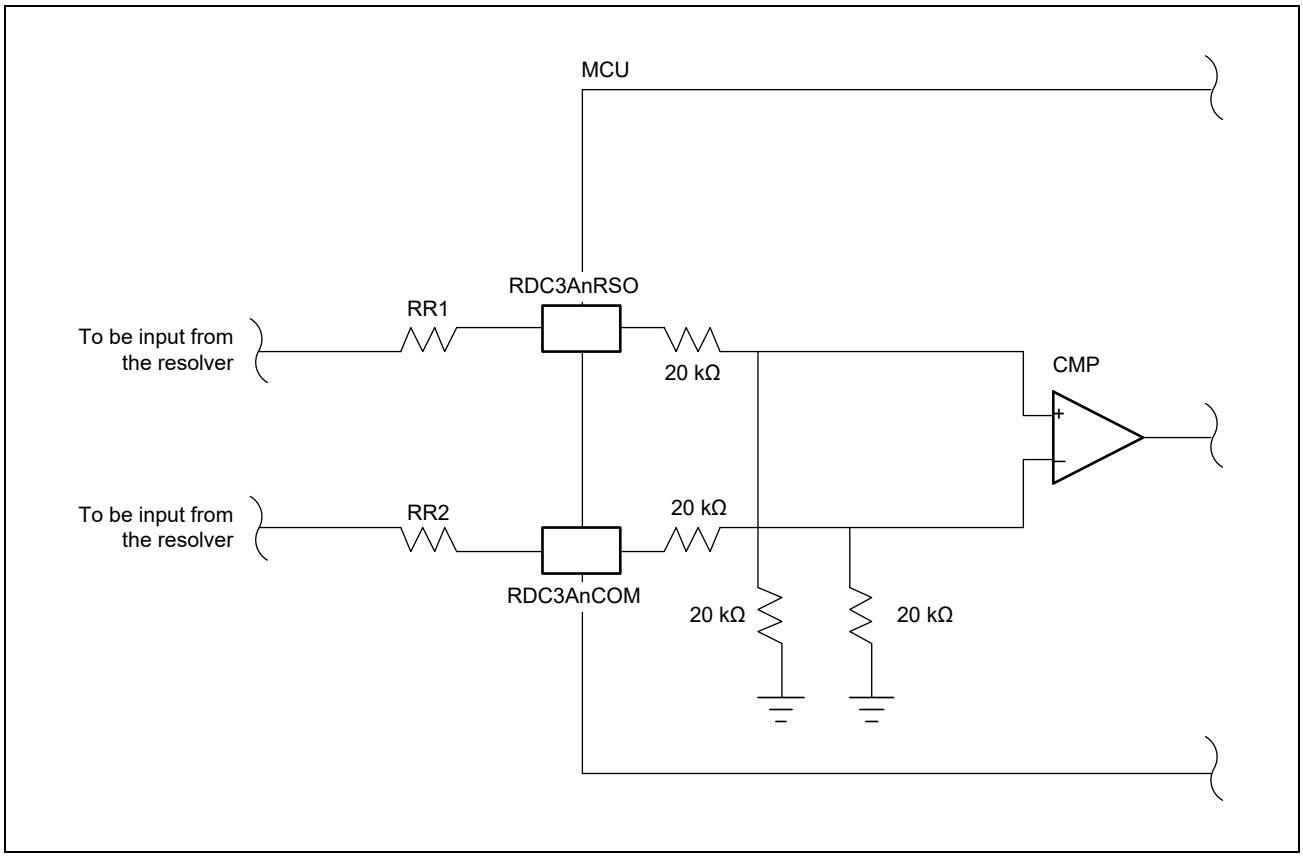


Figure 26.34 Resolver Excitation Signal External Input Circuit (With Single Power Supply)

Table 26.66 Additional Resistor Value for Resolver Excitation Signal External Input Circuit (Reference Values, Single Power Supply)

+VEXT	RR1, RR2
5 V system	0 kΩ
12 V system	47 kΩ
24 V system	120 kΩ

26.6.3.2 Resolver Excitation Signal Input Circuit (Dual Power Supply)

Figure 26.35 shows an equivalent circuit when the resolver excitation signal is input from an external source (dual power supply). Adjust Rn1 and Rn2 so that the voltage on the RDC3AnCOM pin does not exceed 5V.

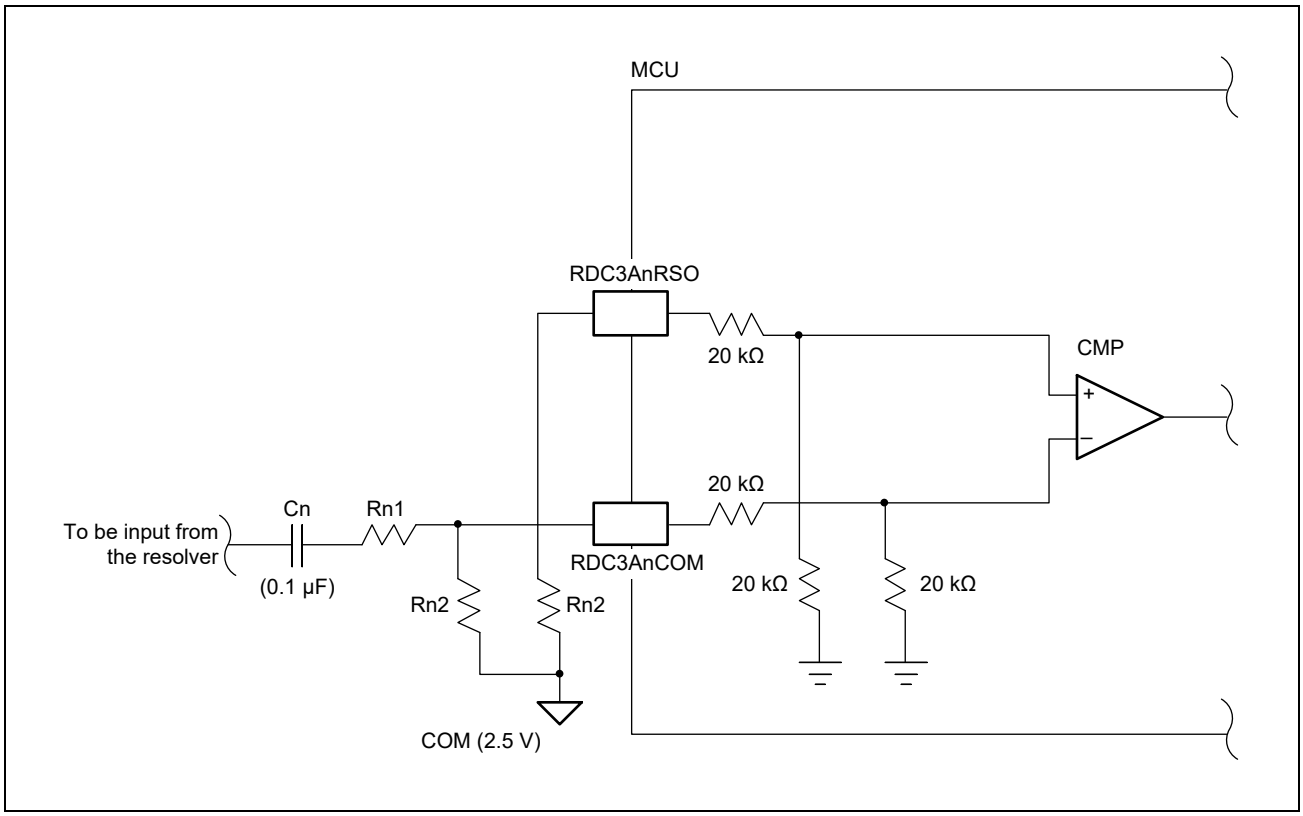


Figure 26.35 Resolver Excitation Signal External Input Circuit (with Dual Power Supply)

26.7 Usage Notes

When the resolver is used as a motor sensor, the resolver signal is affected by various types of noises depending on the drive control configuration of the motor. To perform R/D conversion successfully, sufficient S/N ratio of the resolver signal is required.

This RDC is a highly responsive R/D conversion module. Although considerations have been made for noise immunity, the RDC by itself is not designed to accommodate all noise environmental factors. Appropriate peripheral circuits need to be considered depending on the environment in which the module is deployed. The following describes the specific countermeasures for noise in [Countermeasure I] to [Countermeasure VIII] as reference.

26.7.1 Countermeasures for Magnetic Disturbance Noise

If the leakage flux of the motor passes through the resolver, the resolver signal behaves as if the angle changes, resulting in malfunction.

[Countermeasure I]

When installing the motor and resolver, use the configuration and material that block the magnetic loop passing through the resolver (magnetic shield effect) to minimize the leakage flux of the motor that passes through the resolver.

[Countermeasure II]

If the leakage flux of the motor that passes through the resolver cannot be completely avoided, raise the resolver excitation voltage (current) to improve the S/N ratio of the original signals.

26.7.2 Countermeasures for Electric Disturbance Noise

The electric disturbance (spike noise, and so on) caused by the PWM drive of the motor is extremely large, and affects all power systems, such as the excitation signal lines of resolver and power supply line, via various paths.

[Countermeasure III]

Insert a common mode/normal mode filter into the resolver excitation lines to remove the spike noise components. Generally, the low-impedance excitation line hardly has noise and the countermeasure is less needed.

[Countermeasure IV]

Insert a common mode/normal mode filter into the resolver signal lines (RDC3AnS1-RDC3AnS3, RDC3AnS2-RDC3AnS4) to remove the spike noise components. Select the time constant that takes effect only on noise and leaves the original resolver signal waveform undistorted. In addition, ensure that the electric noise waveform of the RDC3AnS1 to RDC3AnS4 pins viewed from RVSS (ground) are in phase.

If an error due to electric external disturbance noise persists after this countermeasure is taken, keeping the resolver signal level low can be effective.

[Countermeasure V]

If necessary, insert a bypass capacitor to the power supply (RVCC) line.

26.7.3 Other General Measures

[Countermeasure VI]

Use a shielded twisted pair cable for resolver wiring. Shielded terminals must be treated collectively on the circuit side (grounded to GND). The cable must be routed separately from the motor cable.

[Countermeasure VII]

Enhance the GND system for low impedance to reduce common impedance noise and to provide shielding effect. Another possible measure is to fix the potential of the motor driver radiator and motor case to the control-system ground potential.

[Countermeasure VIII]

Physically separate the motor driver from the sensor circuit and cover each of them with a shield case.

Section 27 A/D Converter (ADCC)

This section contains a generic description of the A/D Converter (ADCC).

The first part describes all RH850/C1M-A specific properties such as the number of units, register base address, etc. The subsequent parts describe the functions and registers of ADCC.

27.1 Features of RH850/C1M-A ADCC

27.1.1 Number of Units

This LSI has the following number of units of ADCC.

Table 27.1 Number of Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	3	3
Name	ADCCn (n = 0, 1, 2)	ADCCn (n = 0, 1, 2)

Table 27.2 Indices

Index	Meaning
n	Throughout this section, the individual ADCC units are identified by the index “n” (n = 0, 1, 2), for example, ADCCnVCRj for the virtual channel register j.
j	Throughout this section, the number of data registers and virtual channels of ADCC is indicated by the index “j” (j = 0 to 39), for example, ADCCnDRj for the data register j.
k	Throughout this section, the number of each T&H channel of ADCC is indicated by the index “k” (k = 0 to 5).
p	Throughout this section, the physical channel group is identified by the index “p” (p = 0 to 3).
q	Throughout this section, the physical sub-channel group identified by the index “q” (q = 0 to 3).
x	Throughout this section, the scan group is identified by the index “x” (x = 0 to 4).
y	Throughout this section, the number of A/D timers is indicated by the index “y” (y = 3 to 4).

27.1.2 Register Base Address

ADCC base addresses are listed in the following table.

ADCC register addresses are given as offsets from the individual base address.

Table 27.3 Register Base Address

Base Address Name	Base Address
<ADCC0_base>	$FFF2\ 0000_H$
<ADCC1_base>	$FF92\ 1000_H$
<ADCC2_base>	$FFF2\ 2000_H$

27.1.3 Clock Supply

ADCC clock is listed in following table.

Table 27.4 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
ADCCn	PCLK	CLK_LSB (low-speed peripheral clock)
	clkad	CLKC_LSB (non-modulated low-speed peripheral clock)

27.1.4 Interrupt Requests

ADCC interrupt requests are listed in the following table. Scan group x end interrupt of ADCCn is expressed as ADInx.

Table 27.5 Interrupt Requests

Interrupt Name (Outline)		Interrupt Number	DMA Trigger Number		DTS Trigger Number	
			1st	2nd	1st	2nd
ADCC0						
ADI00	ADCC0 scan group 0 end interrupt	243	0	—	0	—
ADI01	ADCC0 scan group 1 end interrupt	244	1	—	1	—
ADI02	ADCC0 scan group 2 end interrupt	245	2	—	2	—
ADI03	ADCC0 scan group 3 end interrupt	246	3	—	3	—
ADI04	ADCC0 scan group 4 end interrupt	247	4	—	4	—
ADE0	ADCC0 AD error interrupt	258	—	—	—	—
ADCC1						
ADI10	ADCC1 scan group 0 end interrupt	248	5	—	5	—
ADI11	ADCC1 scan group 1 end interrupt	249	6	—	6	—
ADI12	ADCC1 scan group 2 end interrupt	250	7	—	7	—
ADI13	ADCC1 scan group 3 end interrupt	251	8	—	8	—
ADI14	ADCC1 scan group 4 end interrupt	252	9	—	9	—
ADE1	ADCC1 AD error interrupt	259	—	—	—	—
ADCC2						
ADI20	ADCC2 scan group 0 end interrupt	253	10	—	10	—
ADI21	ADCC2 scan group 1 end interrupt	254	11	—	11	—
ADI22	ADCC2 scan group 2 end interrupt	255	12	—	12	—
ADI23	ADCC2 scan group 3 end interrupt	256	13	—	13	—
ADI24	ADCC2 scan group 4 end interrupt	257	14	—	14	—
ADE2	ADCC2 AD error interrupt	260	—	—	—	—

—: No number is assigned.

Note: 1st: Primary channel, 2nd: Secondary channel

27.1.5 Reset Sources

ADCC reset sources are listed in the following table. ADCC is initialized by these reset sources.

Table 27.6 Reset Source

Unit	Reset Source
ADCCn	Any reset source

27.1.6 External Input/Output Signals

External input/output signals of ADCC and pin assignment of track and hold (T&H) circuit are listed in the following table.

External output pins are not assigned for ADCC.

Table 27.7 External Input/Output Signals (1/2)

Unit Signal Name	Outline	Alternative Port Signal	Assignment of T & H circuit	Product	
				C1M-A2 252 pin	C1M-A1 176 pin
AVcc0	ADCC0 power supply pin	A0VCC	—	√	√
AVss0	ADCC0 ground pin	A0VSS	—	√	√
AVcc1	ADCC1 power supply pin	A1VCC	—	√	√
AVss1	ADCC1 ground pin	A1VSS	—	√	√
AVcc2	ADCC2 power supply pin	A2VCC	—	√	√
Avss2	ADCC2 ground pin	A2VSS	—	√	√
AVREFH0	ADCC0 reference voltage pin	A0VREFH	—	√	√
AVREFH1	ADCC1 reference voltage pin	A1VREFH	—	√	√
AVREFH2	ADCC2 reference voltage pin	A2VREFH	—	√	√
AN000	Analog input pin 000	ADCC0I00	T&H circuit 0	√	√
AN001	Analog input pin 001	ADCC0I01	T&H circuit 1	√	√
AN002	Analog input pin 002	ADCC0I02	T&H circuit 2	√	√
AN003	Analog input pin 003	ADCC0I03	T&H circuit 3	√	√
AN010	Analog input pin 010	ADCC0I10	T&H circuit 4	√	√
AN011	Analog input pin 011	ADCC0I11	T&H circuit 5	√	√
AN012	Analog input pin 012	ADCC0I12	—	√	√
AN013	Analog input pin 013	ADCC0I13	—	√	√
AN020	Analog input pin 020	ADCC0I20	—	√	√
AN021	Analog input pin 021	ADCC0I21	—	√	√
AN022	Analog input pin 022	ADCC0I22	—	√	—
AN023	Analog input pin 023	ADCC0I23	—	√	—
AN030	Analog input pin 030	ADCC0I30	—	√	√
AN031	Analog input pin 031	ADCC0I31	—	√	—
AN032	Analog input pin 032	ADCC0I32	—	√	—
AN033	Analog input pin 033	ADCC0I33	—	√	—

Table 27.7 External Input/Output Signals (2/2)

Unit Signal Name	Outline	Alternative Port Signal	Assignment of T & H circuit	Product	
				C1M-A2 252 pin	C1M-A1 176 pin
AN100	Analog input pin 100	ADCC1I00	—	√	√
AN101	Analog input pin 101	ADCC1I01	—	√	√
AN102	Analog input pin 102	ADCC1I02	T&H circuit 0	√	√
AN103	Analog input pin 103	ADCC1I03	T&H circuit 1	√	√
AN110	Analog input pin 110	ADCC1I10	T&H circuit 2	√	√
AN111	Analog input pin 111	ADCC1I11	T&H circuit 3	√	√
AN112	Analog input pin 112	ADCC1I12	T&H circuit 4	√	√
AN113	Analog input pin 113	ADCC1I13	T&H circuit 5	√	√
AN120	Analog input pin 120	ADCC1I20	—	√	√
AN121	Analog input pin 121	ADCC1I21	—	√	√
AN122	Analog input pin 122	ADCC1I22	—	√	√
AN123	Analog input pin 123	ADCC1I23	—	√	√
AN130	Analog input pin 130	ADCC1I30	—	√	—
AN131	Analog input pin 131	ADCC1I31	—	√	√
AN132	Analog input pin 132	ADCC1I32	—	√	√
AN133	Analog input pin 133	ADCC1I33	—	√	—
AN200	Analog input pin 200	ADCC2I00	T&H circuit 0	√	√
AN201	Analog input pin 201	ADCC2I01	T&H circuit 1	√	√
AN202	Analog input pin 202	ADCC2I02	T&H circuit 2	√	√
AN203	Analog input pin 203	ADCC2I03	T&H circuit 3	√	√
AN210	Analog input pin 210	ADCC2I10	—	√	√
AN211	Analog input pin 211	ADCC2I11	—	√	—
AN212	Analog input pin 212	ADCC2I12	—	√	—
AN213	Analog input pin 213	ADCC2I13	—	√	—
AN220	Analog input pin 220	ADCC2I20	—	√	—
AN221	Analog input pin 221	ADCC2I21	—	√	—
AN222	Analog input pin 222	ADCC2I22	—	√	—
AN223	Analog input pin 223	ADCC2I23	—	√	—
AN230	Analog input pin 230	ADCC2I30	—	√	—
AN231	Analog input pin 231	ADCC2I31	—	√	—
AN232	Analog input pin 232	ADCC2I32	—	√	—
AN233	Analog input pin 233	ADCC2I33	—	√	—

27.1.7 Rule for Naming Analog Input Pins

An analog input pin name is represented by a unit signal name or alternative port signal name. An analog input pin is called a physical channel. The name of an analog input pin is represented by a physical channel group number and physical subchannel number.

A rule for naming the unit signal name and alternative port signal name of an analog input pin is as follows:

Unit signal name: AN + Unit number + Physical channel group + Physical subchannel

Alternative port signal name: ADCC + Unit number + I + Physical channel group + Physical subchannel

For example, AN120 (ADCC1I20) for ADCC1, physical channel group 2, and physical subchannel 0.

27.2 Overview

27.2.1 Functional Overview

ADCC has the following features.

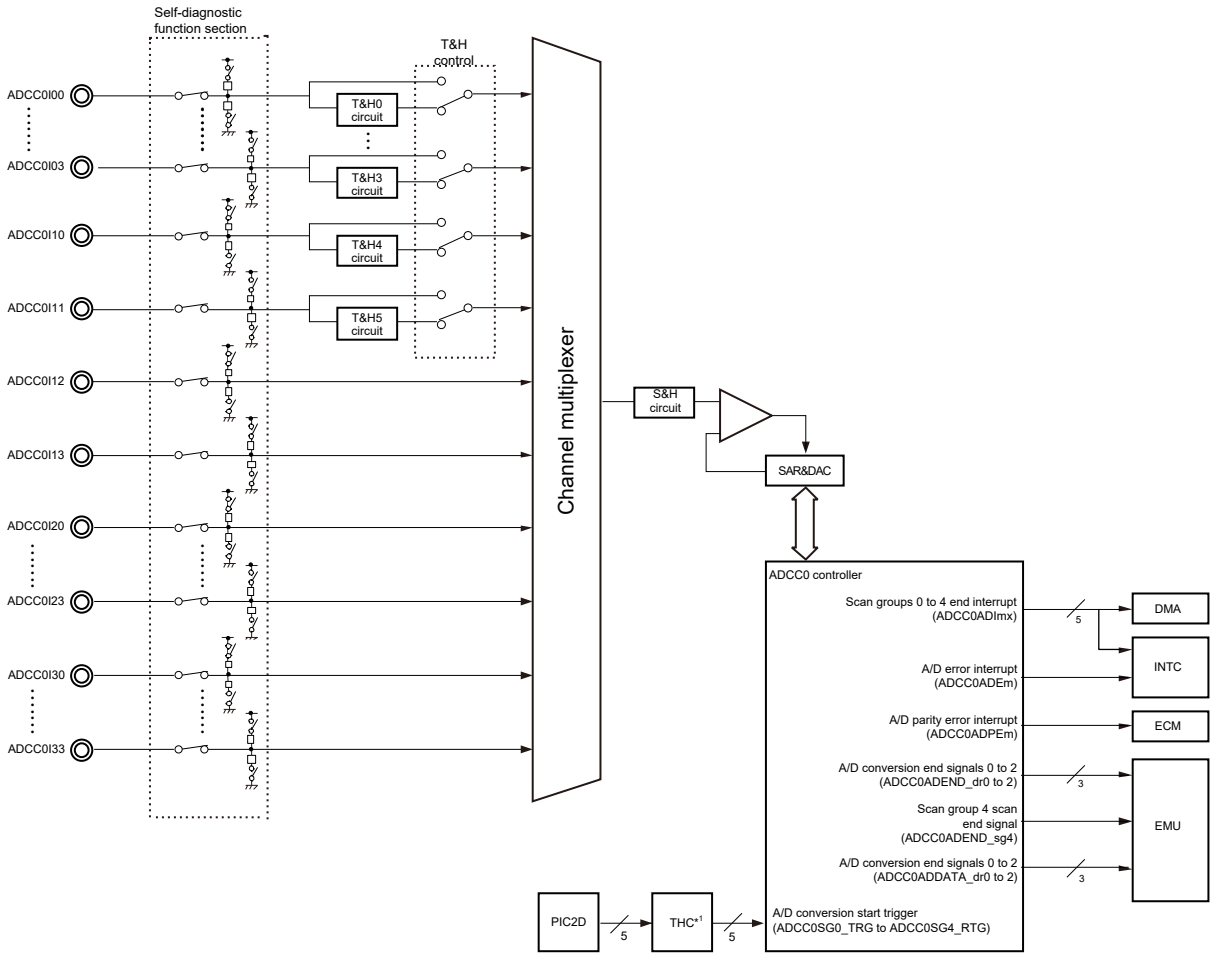
Table 27.8 Functional Overview of ADCC

Item	Outline
Resolution	12 bits
A/D converter	Successive approximation method
Conversion speed	1.0 μs per channel
Number of virtual channels (virtual channel)	ADCC0: 40 channels (virtual ch0 to 39) ADCC1: 40 channels (virtual ch0 to 39) ADCC2: 40 channels (virtual ch0 to 39)
Number of scan groups (SG)	ADCC0: 5 groups (SG0 to 4) ADCC1: 5 groups (SG0 to 4) ADCC2: 5 groups (SG0 to 4)
A/D conversion mode	Each ADCC has four A/D conversion modes: • Normal A/D conversion • A/D conversion with a hold value • Self-diagnosis for A/D conversion circuits • Addition A/D conversion
Scan mode	Each ADCC has two scan modes: • Multicycle scan mode: Specified number of scans are executed. • Continuous scan mode: Scans are repeatedly executed without limit.
A/D conversion start trigger	Each ADCC has three A/D conversion start triggers: • Hardware trigger (HW trigger) • Software trigger (SW trigger) • A/D timer trigger (only supported by SG3 and SG4) **Note:** When you use the simultaneous track and hold function, A/D timer trigger cannot be used.
Priority of scan group processing	A processing for a scan group can interrupt an ongoing processing for another scan group. The priority is as follows: High Low SG4 > SG3 > SG2 > SG1 > SG0
Suspend function	Each ADCC has three suspend methods: • Synchronous suspend • Asynchronous suspend • Synchronous and asynchronous combination suspend Make sure that you select asynchronous suspend when using the simultaneous track and hold function.
Interrupts and DMA and DTS transfers function	Each ADCC can make the following interrupts: • Scan group × end interrupt • A/D error interrupt DMA and DTS transfers can be started by a scan group × end interrupt signal.
Function for transferring results of A/D conversion to EMU	Each ADCC can output the following signals to the EMU. • A/D conversion end signal (virtual ch0 to 2) • A/D converted data (virtual ch0 to 2) • SG4 scan end signal
Self-diagnosis function	Each ADCC has the following self-diagnosis functions: • Self-diagnosis for A/D conversion circuits • Pin-level self-diagnosis • Self-diagnosis for wiring-break detection

27.2.2 Block Diagram

Figure 27.1, **Figure 27.2**, and **Figure 27.3** show the block diagram of ADCC0, ADCC1, and ADCC2, respectively.

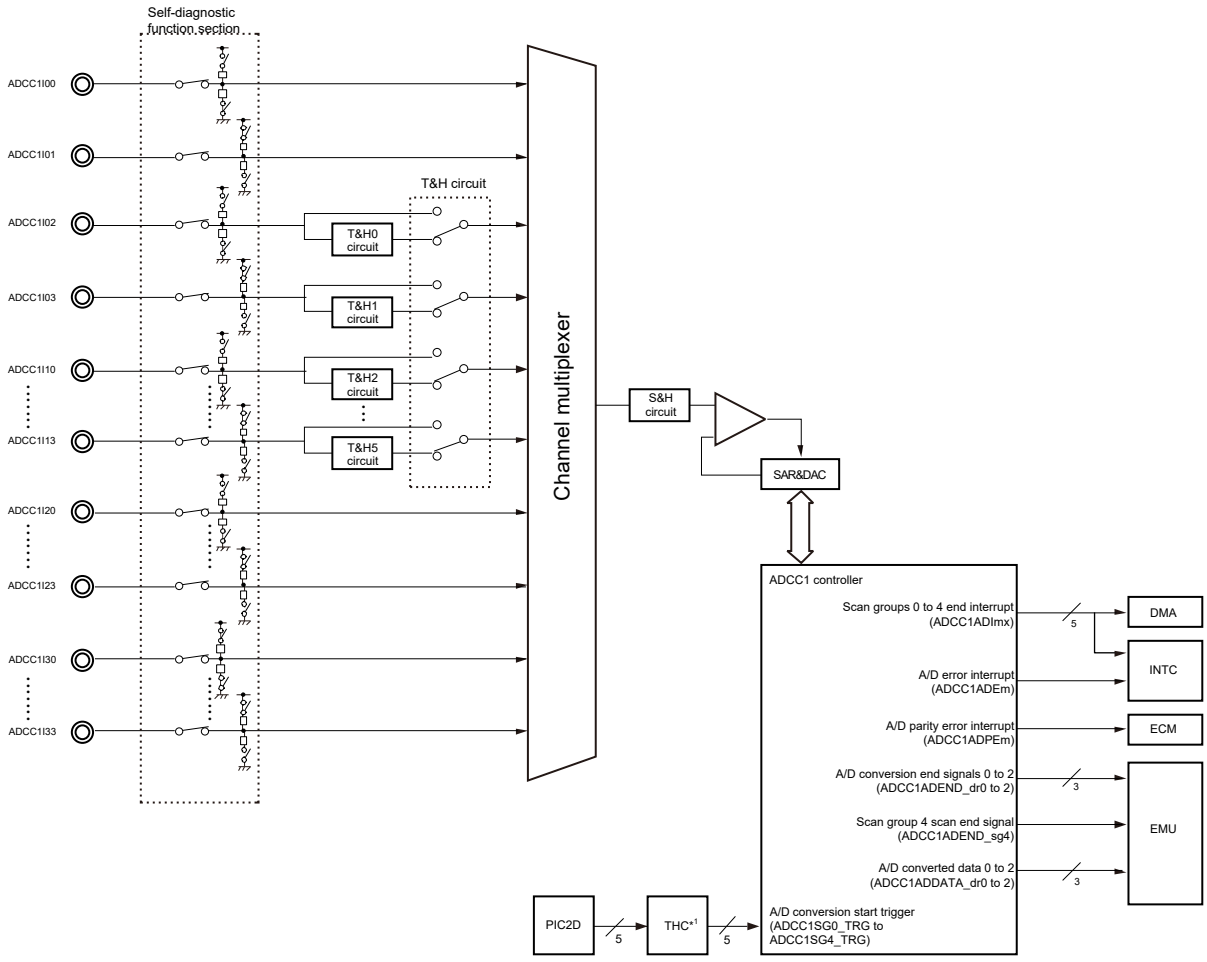
(1) Structure of ADCC0



Note 1. T&H controller

Figure 27.1 ADCC0 Block Diagram

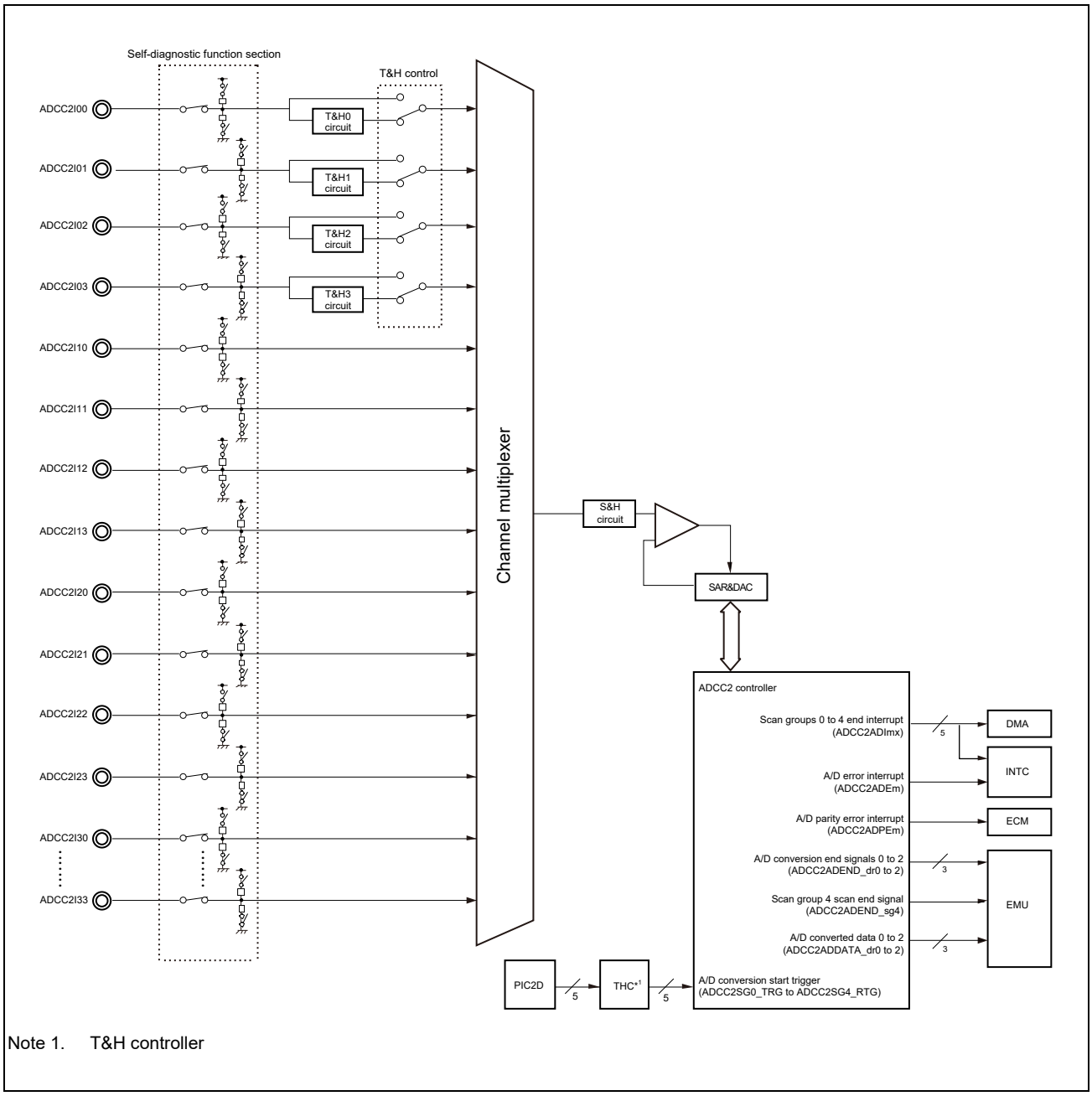
(2) Structure of ADCC1



Note 1. T&H controller

Figure 27.2 ADCC1 Block Diagram

(3) Structure of ADCC2

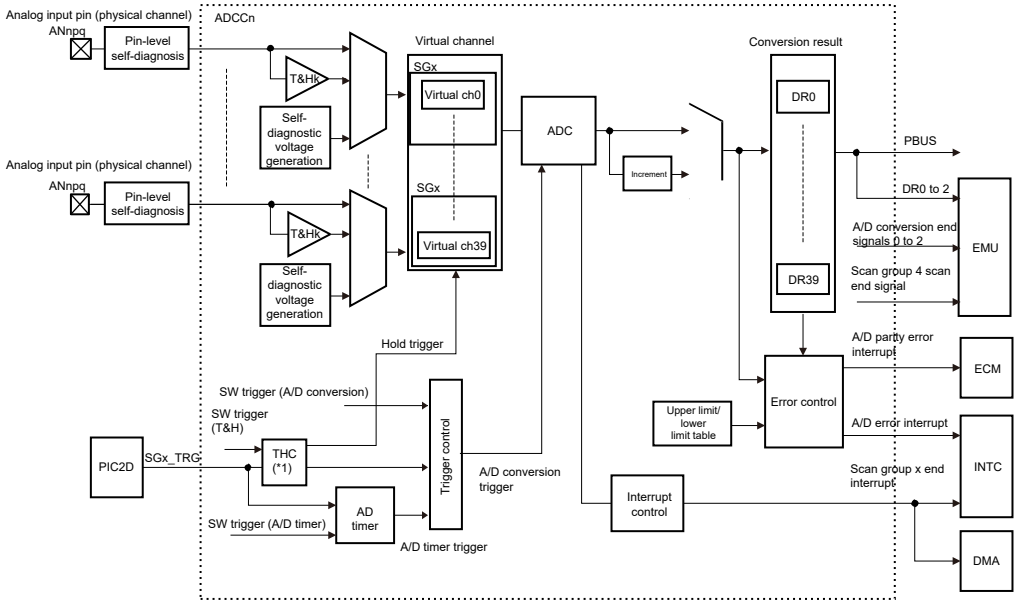


Note 1. T&H controller

Figure 27.3 ADCC2 Block Diagram

(4) ADCC functional block diagram

Figure 27.4 shows a functional block diagram of the ADCC.



Note 1. T&H controller

Figure 27.4 ADCC Functional Block Diagram

27.2.3 Virtual Channel (Virtual ch)

A virtual channel is an analog input pathway which can select an analog input pin (physical channel), T&Hk circuit output, or A/D conversion circuit self-diagnosis output. Each ADCC has 40 virtual channels. The result of A/D conversion is stored in the data register whose number is the same as the virtual channel.

The following diagram is an image of virtual channels.

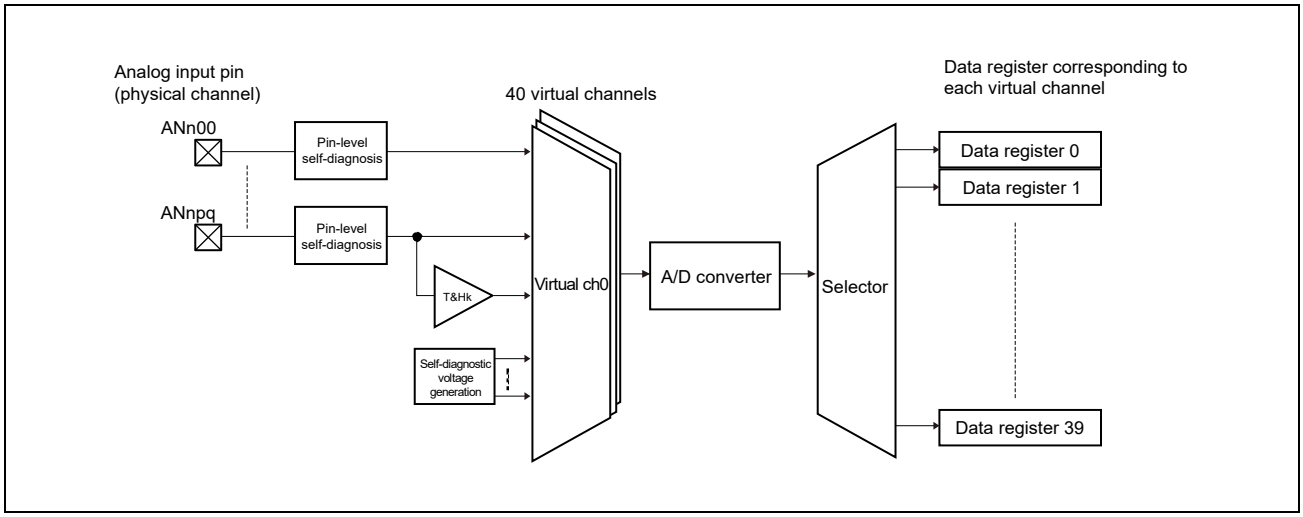


Figure 27.5 Image Diagram of Virtual Channels

27.2.4 Scan Group (SG)

A scan group is a group of multiple virtual channels.

Each ADCC has five scan groups. The priority of A/D conversion is as follows:
SG4 > SG3 > SG2 > SG1 > SG0

SGx can group sequential virtual channels. Specify the grouping settings with a start pointer (ADCCnSGVCSPx register) and an end pointer (ADCCnSGVCEPx register). Disable A/D conversion trigger input for unused scan groups.

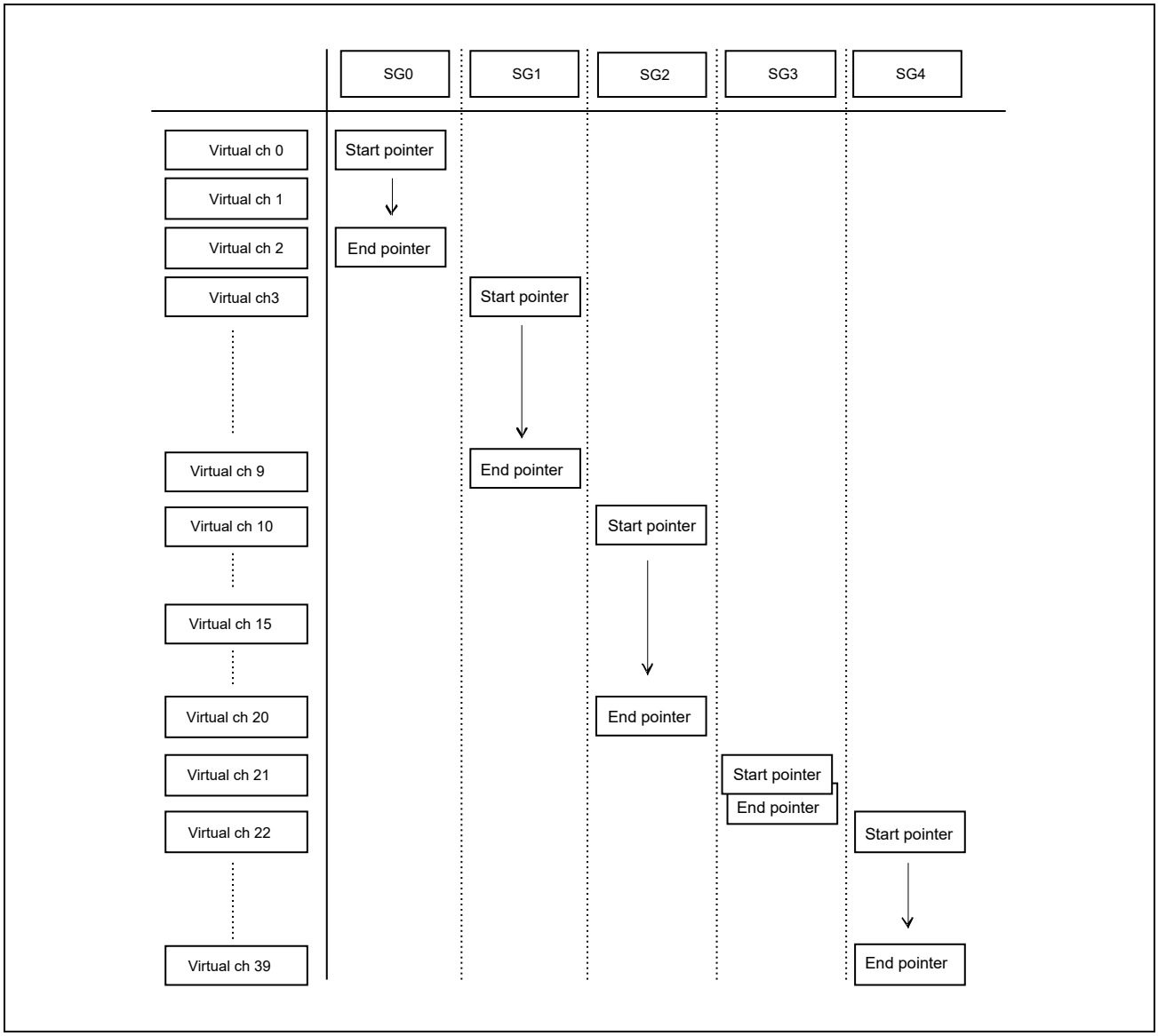


Figure 27.6 Example of SG Assignment

27.3 Register

27.3.1 List of Registers

ADCC registers are listed in the following table.

For <ADCC0_base> and <ADCCn_base>, see **Section 27.1.2, Register Base Address**.

Table 27.9 List of Registers (1/2)

Module Name	Register Name	Symbol	Address
ADCCn	A/D synchronization start control register	ADCC0ADSYNSTCR	<ADCC0_base> + 300_H
ADCCn	A/D timer synchronization start control register	ADCC0ADTSYNSTCR	<ADCC0_base> + 304_H
ADCCn	Virtual channel register j	ADCCnVCRj	<ADCCn_base> + j × 4_H
ADCCn	Data register j	ADCCnDRj	<ADCCn_base> + 100_H + j × 2_H
ADCCn	Data supplementary information register j	ADCCnDIRj	<ADCCn_base> + 200_H + j × 4_H
ADCCn	A/D halt register	ADCCnADHALTR	<ADCCn_base> + 380_H
ADCCn	A/D control register 1	ADCCnADCR1	<ADCCn_base> + 384_H
ADCCn	A/D control register 2	ADCCnADCR2	<ADCCn_base> + 398_H
ADCCn	T&H sampling start control register	ADCCnTHSMPSTCR	<ADCCn_base> + 400_H
ADCCn	T&H stop control register	ADCCnTHSTPCR	<ADCCn_base> + 404_H
ADCCn	T&H control register	ADCCnTHCR	<ADCCn_base> + 408_H
ADCCn	T&H group A hold start control register	ADCCnTHAHLDSTCR	<ADCCn_base> + 410_H
ADCCn	T&H group B hold start control register	ADCCnTHBHLDSTCR	<ADCCn_base> + 414_H
ADCCn	T&H group A control register	ADCCnTHACR	<ADCCn_base> + 420_H
ADCCn	T&H group B control register	ADCCnTHBCR	<ADCCn_base> + 424_H
ADCCn	T&H enable register	ADCCnTHER	<ADCCn_base> + 430_H
ADCCn	T&H group select register	ADCCnTHGSR	<ADCCn_base> + 434_H
ADCCn	Safety control register	ADCCnSFTCR	<ADCCn_base> + $3C0_H$
ADCCn	Pin level self-diagnostic control register	ADCCnTDCR	<ADCCn_base> + $3C4_H$
ADCCn	Wiring-break detection control register	ADCCnODCR	<ADCCn_base> + $3C8_H$
ADCCn	Upper-limit/lower-limit table register 0	ADCCnULLMTBR0	<ADCCn_base> + $3CC_H$
ADCCn	Upper-limit/lower-limit table register 1	ADCCnULLMTBR1	<ADCCn_base> + $3D0_H$
ADCCn	Upper-limit/lower-limit table register 2	ADCCnULLMTBR2	<ADCCn_base> + $3D4_H$
ADCCn	Error clear register	ADCCnECR	<ADCCn_base> + $3D8_H$
ADCCn	Upper-limit/lower-limit error register	ADCCnULER	<ADCCn_base> + $3DC_H$
ADCCn	Overwrite error register	ADCCnOWER	<ADCCn_base> + $3E0_H$
ADCCn	Parity error register	ADCCnPER	<ADCCn_base> + $3E4_H$
ADCCn	ID error register	ADCCnIDER	<ADCCn_base> + $3E8_H$
ADCCn	Scan group x start control register	ADCCnSGSTCRx	<ADCCn_base> + x × 80_H + 480_H
ADCCn	A/D timer y start control register	ADCCnADTSTCRy	<ADCCn_base> + y × 80_H + 488_H
ADCCn	A/D timer y end control register	ADCCnADTENDCRy	<ADCCn_base> + y × 80_H + $48C_H$
ADCCn	Scan group x control register	ADCCnSGCRx	<ADCCn_base> + x × 80_H + 490_H
ADCCn	Scan group x start virtual channel pointer	ADCCnSGVCSPx	<ADCCn_base> + x × 80_H + 494_H
ADCCn	Scan group x end virtual channel pointer	ADCCnSGVCEPx	<ADCCn_base> + x × 80_H + 498_H
ADCCn	Scan group x multicycle register	ADCCnSGMCYCRx	<ADCCn_base> + x × 80_H + $49C_H$
ADCCn	Scan group x virtual channel pointer register	ADCCnSGVCPRx	<ADCCn_base> + x × 80_H + $4A0_H$
ADCCn	Scan group x status register	ADCCnSGSRx	<ADCCn_base> + x × 80_H + $4A4_H$
ADCCn	A/D timer initial phase register y	ADCCnADTIPRy	<ADCCn_base> + y × 80_H + $4A8_H$

Table 27.9 List of Registers (2/2)

Module Name	Register Name	Symbol	Address
ADCCn	A/D timer cycle register y	ADCCnADTPRRy	<ADCCn_base> + y × 80_H + $4AC_H$
ADCCn	Scan group x upper-limit/lower-limit table select register	ADCCnULLMSRx	<ADCCn_base> + x × 80_H + $4B0_H$
ADCCn	Virtual channel threshold table register 0	ADCCnVCULLMTBR0	<ADCCn_base> + 700_H
ADCCn	Virtual channel threshold table register 1	ADCCnVCULLMTBR1	<ADCCn_base> + 704_H
ADCCn	Virtual channel threshold table register 2	ADCCnVCULLMTBR2	<ADCCn_base> + 708_H
ADCCn	Virtual channel threshold table register 3	ADCCnVCULLMTBR3	<ADCCn_base> + $70C_H$
ADCCn	Virtual channel threshold table register 4	ADCCnVCULLMTBR4	<ADCCn_base> + 710_H
ADCCn	Virtual channel threshold table register 5	ADCCnVCULLMTBR5	<ADCCn_base> + 714_H
ADCCn	Virtual channel threshold table register 6	ADCCnVCULLMTBR6	<ADCCn_base> + 718_H

27.3.2 ADCC0ADSYNSTCR ─ A/D Synchronization Start Control Register

This is a register that controls simultaneous start of A/D conversion for each scan group of ADCC0, ADCC1, and ADCC2. Only ADCC0 has this register.

Access: This register can be written in 8-bit units.

Address: <ADCC0_base> + 300_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADSTART
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.10 ADCC0ADSYNSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	ADSTART	This bit simultaneously starts A/D conversion for each scan group of ADCC0, ADCC1, and ADCC2. 0: Nothing occurs. (A written value of 0 is ignored.) 1: A/D conversion starts. Enable the SG synchronization start enable (ADCCnSGCRx.ADSTARTE bit) of the SG for which you want to simultaneously start A/D conversion.

27.3.3 ADCC0ADTSYNSTCR ─ A/D Timer Synchronization Start Control Register

This is a register that controls simultaneous start of count operation for each A/D timer of ADCC0, ADCC1, and ADCC2. Only ADCC0 has this register.

Access: This register can be written in 8-bit units.

Address: <ADCC0_base> + 304_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTSTART
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.11 ADCC0ADTSYNSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	ADTSTART	This bit simultaneously starts count operation for each A/D timer of ADCC0, ADCC1, and ADCC2. 0: Nothing occurs. (A written value of 0 is ignored.) 1: A/D timer count starts. Enable the A/D timer synchronization start enable (ADCCnSGCRx.ADTSTARTE bit) of the A/D timer for which you want to simultaneously start count operation.

27.3.4 ADCCnVCRj ─ Virtual Channel Register j

This is a register for specifying the virtual channel settings.

Access: This register can be read or written in 32-bit units.

Address: <ADCCn_base> + j × 4_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VCULME	VCLLME	—	—	—	VCULLMTBS[2:0]			—	—	—	—	—	—	PUE	PDE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNVCLS[2:0]			—	—	—	—	—	ADIE	—	GCTRL[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.12 ADCCnVCRj Register Contents (1/3)

Bit Position	Bit Name	Function
31	VCULME	Virtual channel result exceeded upper limit notification enable 0: Notification of the result of A/D conversion on a virtual channel exceeding the upper limit is not given. 1: Notification of the result of A/D conversion on a virtual channel exceeding the upper limit is given.
30	VCLLME	Virtual channel result below limit notification enable 0: Notification of the result of A/D conversion on a virtual channel falling below the lower limit is not given. 1: Notification of the result of A/D conversion on a virtual channel falling below the lower limit is given.
29 to 27	Reserved	When read, the value after reset is read. When written, write the value after reset.
26 to 24	VCULLMTBS[2:0]	Virtual channel upper-/lower-limit threshold table register select These bits select the virtual channel upper- and lower-limit threshold table registers containing the values to be compared with the results of conversion. 0_H: ADCCnVCULLMTBR0 is selected 1_H: ADCCnVCULLMTBR1 is selected 2_H: ADCCnVCULLMTBR2 is selected 3_H: ADCCnVCULLMTBR3 is selected 4_H: ADCCnVCULLMTBR4 is selected 5_H: ADCCnVCULLMTBR5 is selected 6_H: ADCCnVCULLMTBR6 is selected 7_H: Setting prohibited
23 to 18	Reserved	When read, the value after reset is read. When written, write the value after reset.

Table 27.12 ADCCnVCRj Register Contents (2/3)

Bit Position	Bit Name	Function
17	PUE	Physical channel I/O pull-up resistor control 0_H: The pull-up resistor connected to the I/O pin of the relevant physical channel is switched off. 1_H: The pull-up resistor connected to the I/O pin of the relevant physical channel is switched on. When PUE = 1, the pull-up resistor connected to the I/O pin of the physical channel set in GCTRL[5:0] for the given virtual channel is switched on whenever wiring-break detection and the wiring-break detection self-diagnosis circuit are used. Setting PDE and PUE to 1 at the same time is prohibited. Note 1. Use of track-and-hold is prohibited while PUE = 1. Note 2. Use of the pin-level self-diagnosis function is prohibited while PUE = 1. Note 3. When setting ADCCnVCRj.PUE to 1_B, ADCCnTDCR.TDE should be set to 0_B. Similarly, when setting ADCCnTDCR.TDE to 1_B, PUE of ADCCnVCRj for the virtual channel to handle A/D conversion should be set to 0_B while ADCCnTDCR.TDE = 1_B.
16	PDE	Physical channel I/O pull-down resistor control 0_H: The pull-down resistor connected to the I/O pin of the relevant physical channel is switched off. 1_H: The pull-down resistor connected to the I/O pin of the relevant physical channel is switched on. When PDE = 1, the pull-down resistor connected to the I/O pin of the physical channel set as the given virtual channel is switched on whenever wiring-break detection and the wiring-break detection self-diagnosis circuit are used. Setting PDE and PUE to 1 at the same time is prohibited. Note 1. Use of track-and-hold is prohibited while PDE = 1. Note 2. Use of the pin-level self-diagnosis function is prohibited while PDE = 1. Note 3. When setting ADCCnVCRj.PDE to 1_B, ADCCnTDCR.TDE should be set to 0_B. Similarly, when setting ADCCnTDCR.TDE to 1_B, PDE of ADCCnVCRj for the virtual channel to handle A/D conversion should be set to 0_B while ADCCnTDCR.TDE = 1_B.
15 to 13	CNVCLS[2:0]	These bits set the conversion mode: 0_H: Normal A/D conversion 1_H: Hold value A/D conversion 3_H: Self-diagnosis for A/D conversion circuits 4_H: Addition A/D conversion Other than above: Setting prohibited
12 to 8	Reserved	When read, the value after reset is read. When written, write the value after reset.
7	ADIE	This bit sets whether to output a scan group x end interrupt signal (ADInx). 0: Output prohibited 1: Output allowed
6	Reserved	When read, the value after reset is read. When written, write the value after reset.

Table 27.12 ADCCnVCRj Register Contents (3/3)

Bit Position	Bit Name	Function
5 to 0	GCTRL[5:0]	These bits specify the A/D conversion settings for each conversion mode. • For normal A/D conversion (CNVCLS[2:0] = 0_H) GCTRL[5:2]: Physical channel group GCTRL[1:0]: Physical subchannel • For A/D conversion of the voltage held by the T&H circuit (CNVCLS[2:0] = 1_H) 00_H: The voltage held by T&H0 is A/D converted. 01_H: The voltage held by T&H1 is A/D converted. 02_H: The voltage held by T&H2 is A/D converted. 03_H: The voltage held by T&H3 is A/D converted. 04_H: The voltage held by T&H4 is A/D converted. 05_H: The voltage held by T&H5 is A/D converted. Other than above: Setting prohibited. • For A/D converter self-diagnosis (CNVCLS[2:0] = 3_H) 00_H: AVREFH × 0 04_H: AVREFH × 1/4 08_H: AVREFH × 1/2 $0C_H$: AVREFH × 3/4 10_H: AVREFH × 1 Other than above: Setting prohibited. • For additive A/D conversion (CNVCLS[2:0] = 4_H) GCTRL[5:2]: Physical channel group GCTRL[1:0]: Physical subchannel

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.5 ADCCnDRj ─ Data Register j

This is a register that stores A/D conversion result data. This register can be read in 32-bit units (ADCCnDRj and ADCCnDRj + 1) because the ADCCnDRj register and the ADCCnDRj + 1 register are aligned successively.

When the read and clear enable (ADCCnSFTCR.RDCLRE bit) is 1 (enabled), be sure to read this register in 32-bit units (ADCCnDRj_j + 1). For the structure of the register, see **Figure 27.7, Data Register Alignment for 32-bit Read**.

Access: This register can be read in 16-bit units.

Address: <ADCCn_base> +100_H + j × 2_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRj[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.13 ADCCnDRj Register Contents

Bit Position	Bit Name	Function
15 to 0	DRj[15:0]	Data register These bits store A/D conversion result data.

CAUTION

Note that ADCCnDIRj.WFLG and ADCCnDIR(j+1).WFLG are cleared by reading ADCCnDRj in 16-bit units.

The ADCCnADCR2.DFMT bit can be used for changing the format to the signed fixed-point format or signed integer format. The following tables show data arrangement for each format. Convert twice and Convert four times in the table are data formats for addition A/D conversion.

- For signed fixed-point format (ADCCnADCR2.DFMT = 0)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Convert once	S	A/D conversion result												0	0	0
Convert twice	S	A/D conversion result (twice additional value)													0	0
Convert four times	S	A/D conversion result (four times additional value)														0

Position of decimal point

- For signed integer format (ADCCnADCR2.DFMT = 1)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Convert once	S	S	S	S	A/D conversion result											
Convert twice	S	S	S	A/D conversion result (twice additional value)												
Convert four times	S	S	A/D conversion result (four times additional value)													

Position of decimal point

S	: Sign bit (always 0)
0	: Zero extension

When you want to read the data register j in 32-bit units, read ADCCnDRj_j + 1. The data alignment of ADCCnDRj_j + 1 is as follows.

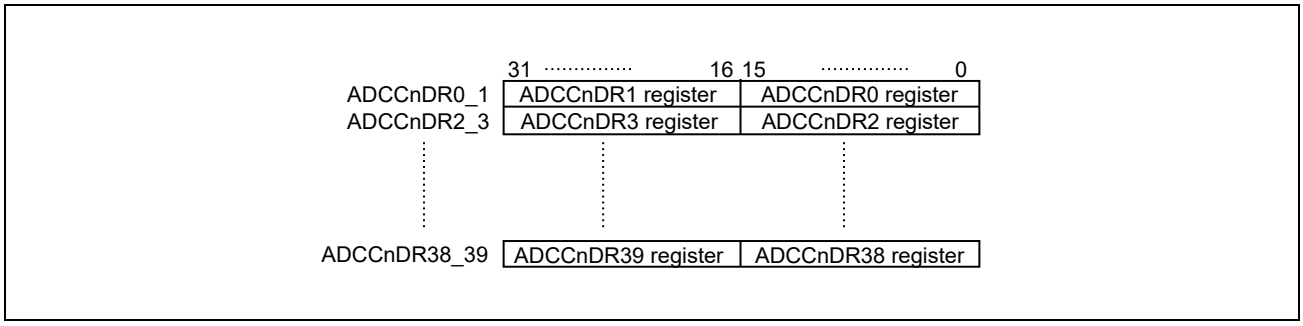


Figure 27.7 Data Register Alignment for 32-bit Read

27.3.6 ADCCnDIRj – Data Supplementary information Register j

This is a register that stores supplementary information of data register (ADCCnDRj register) and A/D converted value. This register must always be read as 32-bit data.

When the read and clear setting (ADCCnSFTCR.RDCLRE bit) is 1, if the ADCCnDRj or ADCCnDIRj register is read, the ADCCnDRj and ADCCnDIRj registers are all cleared to 0. The ADCCnDIRj.WFLG bit is cleared when the ADCCnDRj or ADCCnDIRj register is read regardless of the read and clear setting (ADCCnSFTCR.RDCLRE bit).

Access: This register can be read in 32-bit units.

Address: <ADCCn_base> +200_H + j × 4_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	WFLG	PRTY	—	—	—	ID[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRj[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.14 ADCCnDIRj Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	Reserved	When read, the value after reset is read.
25	WFLG	Write Flag Setting condition: An A/D converted value is stored in ADCCnDRj. Clearing condition: The ADCCnDRj or ADCCnDIRj register is read.
24	PRTY	Parity Parity bit (even parity) for the bits of the ADCCnDRj register and the ADCCnDIR.ID[4:0] bits
23 to 21	Reserved	When read, the value after reset is read.

Table 27.14 ADCCnDIRj Register Contents (2/2)

Bit Position	Bit Name	Function
20 to 16	ID[4:0]	ID information: Hold the physical channel information which has actually been A/D converted. • For normal A/D conversion (CNVCLS[2:0] = 0_H) ID[4:2]: Indicate the physical channel group. ID[1:0]: Indicate the physical sub-channel. • For A/D conversion of the voltage held by the T&H circuit (CNVCLS[2:0] = 1_H) ID[2:0]: Indicate the A/D converted value of the voltage held by the T&H circuit. 0_H: The voltage held by T&H0 is A/D converted. 1_H: The voltage held by T&H1 is A/D converted. 2_H: The voltage held by T&H2 is A/D converted. 3_H: The voltage held by T&H3 is A/D converted. 4_H: The voltage held by T&H4 is A/D converted. 5_H: The voltage held by T&H5 is A/D converted. The other ID bits are fixed to 0. • For self-diagnostic A/D conversion by the AD core (CNVCLS[2:0] = 3_H) ID[4:0]: Indicate the AD-core self-diagnosis level. 10_H: AVREFH × 1 $0C_H$: AVREFH × 3/4 08_H: AVREFH × 1/2 04_H: AVREFH × 1/4 00_H: AVREFH × 0 • For additive A/D conversion mode (CNVCLS[2:0] = 4_H) ID[4:2]: Indicate the physical channel group. ID[1:0]: Indicate the physical sub-channel.
15 to 0	DRj[15:0]	Data register These bits store the same A/D conversion result data as the ADCCnDRj register.

27.3.7 ADCCnADHALTR ─ A/D Halt Register

This is a register for terminating each ADCC.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 380_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HALT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.15 ADCCnADHALTR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	HALT	This bit forcibly terminates all the scan groups, A/D conversions, and A/D timers. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Termination is performed.

27.3.8 ADCCnADCR1 ─ A/D Control Register 1

This is a register for setting ADCC common control (suspend method).

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 384_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SUSMTD[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 27.16 ADCCnADCR1 Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When written, write the value after reset.
1, 0	SUSMTD[1:0]	These bits select the suspend method. 0: Synchronous suspend 1: Synchronous and asynchronous combination suspend 2: Asynchronous suspend 3: Setting prohibited

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.9 ADCCnADCR2 ─ A/D Control Register 2

This is a register for setting ADCC common control (data format and addition count for addition A/D conversion).

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 398_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	DFMT	—	—	—	ADDNT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W

Table 27.17 ADCCnADCR2 Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is read. When written, write the value after reset.
4	DFMT	This bit is the data format setting for the ADCCnDRj and ADCCnDIRj register. 0: Signed fixed-point format 1: Signed integer format For details of data format, see **Section 27.3.5, ADCCnDRj ─ Data Register j**.
3 to 1	Reserved	When read, the value after reset is read. When written, write the value after reset.
0	ADDNT	This bit selects the addition count for addition A/D conversion. 0: Add twice 1: Add four times

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.10 ADCCnTHSMPSTCR ─ T&H Sampling Start Control Register

This is a register that controls start of all T&H sampling.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 400_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SMPST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.18 ADCCnTHSMPSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	SMPST	This bit starts all T&H sampling. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Starts sampling

27.3.11 ADCCnTHSTPCR ─ T&H Stop Control Register

This is a register that controls stop of all T&H.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 404_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	THSTP
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.19 ADCCnTHSTPCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	THSTP	This bit stops all T&H. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Stops all T&H.

Use the ADCCnTHSTPCR.THSTP bit after stopping all SGs by using the ADCCnADHALTR register. After stopping an SG by setting the ADCCnTHSTPCR.THSTP bit, set ADCCnTHER.THkE to all 0s to prevent re-sampling operations due to the auto-sampling function (ADCCnTHCR.ASMPMSK bit).

27.3.12 ADCCnTHCR ─ T&H Control Register

This is a register that controls T&H sampling.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 408_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ASMPMSK
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 27.20 ADCCnTHCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When written, write the value after reset.
0	ASMPMSK	This bit selects whether to perform auto sampling when A/D conversion of the relevant hold value is completed. 0: Automatic sampling is performed. 1: Automatic sampling is not performed.

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.13 ADCCnTHAHLDSTCR ─ T&H Group A Hold Start Control Register

This is a register that controls the start of hold for T&H group A.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 410_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HLDST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.21 ADCCnTHAHLDSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	HLDST	This bit starts hold for T&H group A. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Starts hold.

27.3.14 ADCCnTHBHLDSTCR ─ T&H Group B Hold Start Control Register

This is a register that controls the start of hold for T&H group B.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 414_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HLDST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.22 ADCCnTHBHLDSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	HLDST	This bit starts hold for T&H group B. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Starts hold.

27.3.15 ADCCnTHACR ─ T&H Group A Control Register

This is a register that controls T&H group A.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 420_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	HLDCTE	HLDTE	—	—	SGS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 27.23 ADCCnTHACR Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When written, write the value after reset.
5	HLDCTE	This bit selects whether hold control is enabled or disabled. 0: Disabled. 1: Hold control is enabled.
4	HLDTE	This bit selects whether H/W trigger signals are enabled or disabled. 0: Disabled. 1: The H/W trigger signal of the scan group selected in the ADCCnTHACR.SGS[1:0] bits is enabled.
3, 2	Reserved	When read, the value after reset is read. When written, write the value after reset.
1, 0	SGS[1:0]	These bits select the scan group for T&H group A. 0: SG1 1: SG2 2: SG3 3: SG4

CAUTION

Set both the ADCCnTHACR.HLDCTE and ADCCnTHACR.HLDTE bits to 1 when using the simultaneous track and hold function with a H/W trigger. Set the ADCCnTHACR.HLDCTE bit to 1 and the ADCCnTHACR.HLDTE bit to 0 when using the simultaneous track and hold function with a S/W trigger.

Do not set the ADCCnTHACR.SGS[1:0] bits and the ADCCnTHBCR.SGS[1:0] bits to the same scan group when both T&H group A and B are used. Set this register in an A/D conversion stop state.

When the ADCCnTHACR.HLDCTE bit is set to 1, set the ADCCnSGCRx.TRGMD1 bit and the ADCCnSGCRx.TRGMD0 bit in SGx to be selected by using ADCCnTHACR.SGS[1:0] bits to 0 and 1 respectively.

27.3.16 ADCCnTHBCR ─ T&H Group B Control Register

This is a register that controls T&H group B.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 424_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	HLDCTE	HLDTE	—	—	SGS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 27.24 ADCCnTHBCR Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When written, write the value after reset.
5	HLDCTE	This bit selects whether hold control is enabled or disabled. 0: Disabled. 1: Hold control is enabled.
4	HLDTE	This bit selects whether H/W trigger signals are enabled or disabled. 0: Disabled. 1: The H/W trigger signal of the scan group selected in the ADCCnTHBCR.SGS[1:0] bits is enabled.
3, 2	Reserved	When read, the value after reset is read. When written, write the value after reset.
1, 0	SGS[1:0]	These bits select the scan group for T&H group B. 0: SG1 1: SG2 2: SG3 3: SG4

CAUTION

Set both the ADCCnTHBCR.HLDCTE and ADCCnTHBCR.HLDTE bits to 1 when using the simultaneous track and hold function with a H/W trigger. Set the ADCCnTHBCR.HLDCTE bit to 1 and the ADCCnTHBCR.HLDTE bit to 0 when using the simultaneous track and hold function with a S/W trigger.

Do not set the ADCCnTHACR.SGS[1:0] bits and the ADCCnTHBCR.SGS[1:0] bits to the same scan group when both T&H group A and B are used. Set this register in an A/D conversion stop state.

When the ADCCnTHBCR.HLDCTE bit is set to 1, set the ADCCnSGCRx.TRGMD1 bit and the ADCCnSGCRx.TRGMD0 bit in SGx to be selected by using ADCCnTHBCR.SGS[1:0] bits to 0 and 1 respectively.

27.3.17 ADCCnTHER ─ T&H Enable Register

This is a register that controls whether to enable or disable of each T&H circuit.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 430_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	TH5E	TH4E	TH3E	TH2E	TH1E	TH0E
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.25 ADCCnTHER Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When written, write the value after reset.
5 to 0	THkE	These bits set whether to enable or disable track and hold operation for T&Hk circuits. 0: Disabled 1: Enabled

27.3.18 ADCCnTHGSR ─ T&H Group Select Register

This is a register that selects the T&H group for each T&H.

Access: This register can be read or written in 16-bit units.

Address: <ADCCn_base> + 434_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TH5GS	—	TH4GS	—	TH3GS	—	TH2GS	—	TH1GS	—	TH0GS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W

Table 27.26 ADCCnTHGSR Register Contents

Bit Position	Bit Name	Function
15 to 11, 9, 7, 5, 3, 1	Reserved	When read, the value after reset is read. When written, write the value after reset.
10, 8, 6, 4, 2, 0	THkGS	These bits selects the T&H group for T&Hk. 0: Selects T&H group A. 1: Selects T&H group B.

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.19 ADCCnSFTCR ─ Safety Control Register

This is a register for safety control.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + $3C0_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	RDCLRE	ULEIE	OWEIE	PEIE	IDEIE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 27.27 ADCCnSFTCR Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is read. When written, write the value after reset.
4	RDCLRE	This bit sets whether to perform or prohibit read and clear operations. This bit sets whether to clear the ADCCnDRj and ADCCnDIRj registers to 0 (all 0) when reading the ADCCnDRj or ADCCnDIRj register. 0: Not read and cleared. 1: Read and cleared
3	ULEIE	This bit sets whether to allow or prohibit upper-limit/lower-limit error interrupts. 0: Prohibited 1: Allowed
2	OWEIE	This bit sets whether to allow or prohibit overwrite error interrupts. 0: Prohibited 1: Allowed
1	PEIE	This bit sets whether to allow or prohibit parity errors. 0: Prohibited 1: Allowed
0	IDEIE	This bit sets whether to allow or prohibit ID error interrupts. 0: Prohibited 1: Allowed

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.20 ADCCnTDCR ─ Pin Level Self-diagnostic Control Register

This is a register that controls the pin level self-diagnosis.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + $3C4_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	TDE	—	—	—	—	—	TDLV[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W

Table 27.28 ADCCnTDCR Register Contents

Bit Position	Bit Name	Function
7	TDE	This bit sets whether to perform or prohibit pin level self-diagnosis. 0: Pin level self-diagnosis is not performed. 1: Pin level self-diagnosis is performed. When TDE is set to 1, all analog pins are disconnected from the input buffer.
6 to 2	Reserved	When read, the value after reset is read. When written, write the value after reset.
1, 0	TDLV[1:0]	These bits set the diagnosis voltage to be applied when pin level self-diagnosis is performed. 0: AVSS is applied to even numbers of physical subchannels, and AVCC is applied to odd numbers of physical subchannels. 1: AVCC is applied to even numbers of physical subchannels, and AVSS is applied to odd numbers of physical subchannels. 2: AVSS is applied to even numbers of physical subchannels, and 1/2 × AVCC is applied to odd numbers of physical subchannels. 3: 1/2 × AVCC is applied to even numbers of physical subchannels, and AVSS is applied to odd numbers of physical subchannels.

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.21 ADCCnODCR ─ Wiring-break Detection Control Register

This is a register that controls wiring-break detection.

Access: This register can be read or written in 32-bit units.

Address: <ADCCn_base> + $3C8_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ODDE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.29 ADCCnODCR Register Contents

Bit Position	Bit Name	Function
31	ODDE	This bit sets whether to perform or prohibit wiring-break detection self-diagnosis. 0: Wiring-break detection diagnosis is not performed. 1: Wiring-break detection diagnosis is performed. When ODDE is set to 1, wiring-break detection self-diagnosis is enabled for all analog pins of ADCCn.
30 to 0	Reserved	When read, the value after reset is read. When written, write the value after reset.

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.22 ADCCnULLMTBR0 to ADCCnULLMTBR2 ─ Upper Limit/Lower Limit Table Registers 0 to 2

These registers are for setting the upper-limit and lower-limit values of an A/D converted value.

Access: This register can be read or written in 32-bit units.

Address: ADCCnULLMTBR0: <ADCCn_base> + $3CC_H$
ADCCnULLMTBR1: <ADCCn_base> + $3D0_H$
ADCCnULLMTBR2: <ADCCn_base> + $3D4_H$

Value after reset: $7FFE\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ULMTB[15:0]															
Value after reset	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LLMTB[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 27.30 ADCCnULLMTBR Register Contents

Bit Position	Bit Name	Function
31 to 16	ULMTB[15:0]	These bits set the upper-limit of an A/D converted value. Setting range: 0000_H to $7FFE_H$ These bits should be set with the signed fixed-point format. ULMTB[15] and ULMTB[0] are always fixed to 0. In additive A/D conversion mode (ADCCnVCRj.CNVCLS[2:0] = 4_H), these bits specify the value for comparison with the sum of the converted values.
15 to 0	LLMTB[15:0]	These bits set the lower-limit of an A/D converted value. Setting range: 0000_H to $7FFE_H$ These bits should be set with the signed fixed-point format. LLMTB[15] and LLMTB[0] are always fixed to 0. In additive A/D conversion mode (ADCCnVCRj.CNVCLS[2:0] = 4_H), these bits specify the value for comparison with the sum of the converted values.

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.23 ADCCnECR ─ Error Clear Register

This is a register that controls error clear.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + $3D8_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	ULEC	OWEC	PEC	IDEC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	W

Table 27.31 ADCCnECR Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When written, write the value after reset.
3	ULEC	This bit clears upper-limit and lower-limit errors. 0: Not cleared. 1: Cleared.
2	OWEC	This bit clears overwrite errors. 0: Not cleared. 1: Cleared.
1	PEC	This bit clears parity errors. 0: Not cleared. 1: Cleared.
0	IDEC	This bit clears ID errors. 0: Not cleared. 1: Cleared.

27.3.24 ADCCnULER ─ Upper Limit/Lower Limit Error Register

This is a register that indicates an upper limit or lower limit error. This register can be cleared by writing 1 to the upper-limit and lower-limit error clear (ADCCnECR.ULEC bit).

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + $3DC_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	ULE	—	ULECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.32 ADCCnULER Register Contents

Bit Position	Bit Name	Function
7	ULE	This bit indicates whether an upper-limit or lower-limit error exists. 0: An upper-limit or lower-limit error does not exist. 1: An upper-limit or lower-limit error exists.
6	Reserved	When read, the value after reset is read.
5 to 0	ULECAP[5:0]	These bits indicate the number of the virtual channel at the time when an upper-limit or lower-limit error occurred.

CAUTION

This register is retained until 1 is written to the upper-limit and lower-limit error clear (ADCCnECR.ULEC bit). While the value of this register is retained, if an upper-limit and lower-limit error occurs, the new error information is discarded.

27.3.25 ADCCnOWER – Overwrite Error Register

This is a register that indicates an overwrite error. This register can be cleared by writing 1 to the overwrite error clear (ADCCnECR.OWEC bit).

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + $3E0_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	OWE	—	OWECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.33 ADCCnOWER Register Contents

Bit Position	Bit Name	Function
7	OWE	This bit indicates whether an overwrite error exists. 0: An overwrite error does not exist. 1: An overwrite error exists.
6	Reserved	When read, the value after reset is read.
5 to 0	OWECAP[5:0]	These bits indicate the number of the virtual channel at the time when an overwrite error occurred.

CAUTION

This register is retained until 1 is written to the overwrite error clear (ADCCnECR.OWEC bit). While the value of this register is retained, if an overwrite error occurs, the new error information is discarded.

27.3.26 ADCCnPER ─ Parity Error Register

This is a register that indicates a parity error. This register can be cleared by writing 1 to the parity error clear (ADCCnECR.PEC bit).

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + $3E4_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	PE	—	PECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.34 ADCCnPER Register Contents

Bit Position	Bit Name	Function
7	PE	This bit indicates whether a parity error exists. 0: A parity error does not exist. 1: A parity error exists.
6	Reserved	When read, the value after reset is read.
5 to 0	PECAP[5:0]	These bits indicate the number of the virtual channel at the time when a parity error occurred.

CAUTION

This register is retained until 1 is written to the parity error clear (ADCCnECR.PEC bit). While the value of this register is retained, if a parity error occurs, the new error information is discarded.

27.3.27 ADCCnIDER – ID Error Register

This is a register that indicates an ID error. This register can be cleared by writing 1 to the ID error clear (ADCCnECR.IDEC bit).

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + $3E8_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	IDE	—	IDECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.35 ADCCnIDER Register Contents

Bit Position	Bit Name	Function
7	IDE	This bit indicates whether an ID error exists. 0: An ID error does not exist. 1: An ID error exists
6	Reserved	When read, the value after reset is read.
5 to 0	IDECAP[5:0]	These bits indicate the number of the virtual channel at the time when an ID error occurred.

CAUTION

This register is retained until 1 is written to the ID error clear (ADCCnECR.IDEC bit). While the value of this register is retained, if an ID error occurs, the new error information is discarded.

27.3.28 ADCCnSGSTCRx ─ Scan Group x Start Control Register

This is a register that controls the start of A/D conversion for SGx.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + 480_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SGST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.36 ADCCnSGSTCRx Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	SGST	This bit starts A/D conversion for SGx. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Start A/D conversion when the A/D conversion start scan group status (ADCCnSGSRx.SGACT) is 0.

27.3.29 ADCCnADTSTCRy ─ A/D Timer y Start Control Register

This is a register that controls the start of A/D timer y.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + y × 80_H + 488_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.37 ADCCnADTSTCRy Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	ADTST	This bit starts A/D timer y. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Starts the A/D timer. Start the A/D timer when the A/D timer status (ADCCnSGSRx.ADTACT) is 0.

27.3.30 ADCCnADTENDCRy ─ A/D Timer y End Control Register

This is a register that controls the end of A/D timer y.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + y × 80_H + $48C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTEND
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.38 ADCCnADTENDCRy Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	ADTEND	This bit ends the A/D timer. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Ends the A/D timer.

27.3.31 ADCCnSGCRx ─ Scan Group x Control Register

This is a register that controls SGx.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + 490_H

Value after reset: 00_H

- When x = 0 to 2

Bit	7	6	5	4	3	2	1	0
	—	ADSTARTE	SCANMD	ADIE	—	—	—	TRGMD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R/W

Table 27.39 ADCCnSGCRx Register Contents (x = 0 to 2)

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When written, write the value after reset.
6	ADSTARTE	This bit sets whether the synchronization start signal of SG is enabled or disabled. 0: Disabled 1: Enabled
5	SCANMD	This bit sets scan mode. 0: Multicycle scan mode 1: Continuous scan mode
4	ADIE	This bit sets whether the output of the scan group x end interrupt signal ADInx is allowed or prohibited. 0: Output prohibited. 1: Output allowed.
3 to 1	Reserved	When read, the value after reset is read. When written, write the value after reset.
0	TRGMD0	This bit sets whether A/D conversion start trigger input to scan group x is enabled or disabled. 0: Disabled 1: Enabled

CAUTIONS

1. An A/D conversion start trigger input to the same scan group x that occurs while A/D conversion is being performed for scan group x is ignored.
2. If either of the scan groups is set to continuous scan mode (SCANMD = 1), operation of the lower-priority scan group is not possible.
3. To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

- When x = 3 or 4

Bit	7	6	5	4	3	2	1	0
	ADTSTARTE	ADSTARTE	SCANMD	ADIE	—	—	TRGMD1	TRGMD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Table 27.40 ADCCnSGCRx Register Contents (x = 3 or 4)

Bit Position	Bit Name	Function
7	ADTSTARTE	This bit sets whether the synchronization start signal of A/D timers is enabled or disabled. 0: Disabled 1: Enabled
6	ADSTARTE	This bit sets whether the synchronization start signal of SG is enabled or disabled. 0: Disabled 1: Enabled
5	SCANMD	This bit sets scan mode. 0: Multicycle scan mode 1: Continuous scan mode
4	ADIE	This bit sets whether the output of the scan group x end interrupt signal ADInx is allowed or prohibited. 0: Output prohibited. 1: Output allowed.
3, 2	Reserved	When read, the value after reset is read. When written, write the value after reset.
1	TRGMD1	This bit sets whether trigger input to A/D timers is enabled or disabled. 0: Disabled 1: Enabled
0	TRGMD0	This bit sets whether A/D conversion start trigger input to scan group x is enabled or disabled. 0: Disabled 1: Enabled

27.3.32 ADCCnSGVCSPx ─ Scan Group x Start Virtual Channel Pointer

This is a register that specifies the start pointer of a virtual channel.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + 494_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	VCSP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.41 ADCCnSGVCSPx Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When written, write the value after reset.
5 to 0	VCSP[5:0]	These bits set the start virtual channel number of SGx.

CAUTIONS

1. ADCCnSGVCSPx must be equal or less than ADCCnSGVCEPx.
2. Do not make a setting greater than the number of virtual channels that are included.
3. This register has a mirrored configuration as shown below. Be sure to use the mirror register if the value of SGVCSP must be changed while scan group x is operating.

This Register	Mirror Register
SGVCSPx.VCSP[5:0]	SGVCPRx.VCSP[5:0]

4. Setting a single virtual channel (VCRj) to be in multiple scan groups x is possible by setting the start and end pointers. However, if the value of the virtual channel is overwritten during operation of a scan group, A/D conversion may be performed for the scan group in operation with different settings from those desired.
   To avoid this, note the following point.
   - Confirm that VCRj settings for common use among multiple scan groups to which it is to be assigned will always be the same at the time of system design.
   - Modifications to VCRj should only be made after stopping all multiple scan groups to which it has been assigned.

   If the above is not observed, the results of A/D conversion cannot be guaranteed.
5. See also **Section 27.3.40, ADCCnSGVCPRx ─ Scan Group x Virtual Channel Pointer Register**.
6. To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.33 ADCCnSGVCEPx ─ Scan Group x End Virtual Channel Pointer

This is a register that specifies the end pointer of a virtual channel.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + 498_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	VCEP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.42 ADCCnSGVCEPx Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When written, write the value after reset.
5 to 0	VCEP[5:0]	These bits set the end virtual channel number of SGx.

CAUTIONS

1. ADCCnSGVCSPx must be equal or less than ADCCnSGVCEPx.
2. Do not make a setting greater than the number of virtual channels that are included.
3. This register has a mirrored configuration as shown below. Be sure to use the mirror register if the value of SGVCEP must be changed while scan group x is operating.

This Register	Mirror Register
SGVCEPx.VCEP[5:0]	SGVCPRx.VCEP[5:0]

4. To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

For the other precautions, see **Sections 27.3.32, ADCCnSGVCSPx ─ Scan Group x Start Virtual Channel Pointer**, and **27.3.40 ADCCnSGVCPRx ─ Scan Group x Virtual Channel Pointer Register**.

27.3.34 ADCCnSGMCYCRx – Scan Group x Multicycle Register

This is a register that specifies the number of times for A/D conversion in multicycle scan mode.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + $49C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	MCYC[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.43 ADCCnSGMCYCRx Register Contents

Bit Position	Bit Name	Function
7 to 0	MCYC[7:0]	These bits set the frequency of A/D conversion in multicycle scan mode. Number of times for A/D conversion = MCYC[7:0] + 1

27.3.35 ADCCnSGSRx ─ Scan Group x Status Register

This is a register that indicates the A/D conversion operation status of SGx.

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + x × 80_H + $4A4_H$

Value after reset: 00_H

- When x = 0 to 2

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SGACT	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.44 ADCCnSGSRx Register Contents (x = 0 to 2)

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read.
1	SGACT	This bit indicates the A/D conversion operation status of SGx. 0: A/D conversion for SGx is in idle state. 1: A/D conversion for SGx is running.
0	Reserved	When read, the value after reset is read.

- When x = 3 or 4

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADTACT	SGACT	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.45 ADCCnSGSRx Register Contents (x = 3 or 4)

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read.
2	ADTACT	This bit indicates the operation status of an A/D timer. 0: A/D timer x is in idle state. 1: A/D timer x is running.
1	SGACT	This bit indicates the A/D conversion operation status of SGx. 0: A/D conversion for SGx is in idle state. 1: A/D conversion for SGx is running.
0	Reserved	When read, the value after reset is read.

27.3.36 ADCCnADTIPRy ─ A/D Timer Initial Phase Register y

This is a register that sets the initial phase (initial value of the counter) of A/D timer y.

Access: This register can be read or written in 32-bit units.

Address: <ADCCn_base> + y × 80_H + $4A8_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	ADTIP[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADTIP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.46 ADCCnADTIPRy Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	When read, the value after reset is read. When written, write the value after reset.
20 to 0	ADTIP[20:0]	These bits set the initial phase (initial value of the counter) of A/D timer y. Setting range: 000000_H to $1FFFFF_H$

CAUTIONS

1. For details, see **Section 27.4.3.2(2), Starting scan groups by A/D timer triggers**.
2. To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.37 ADCCnADTPRRy ─ A/D Timer Cycle Register y

This is a register that sets the cycle of A/D timer y.

Access: This register can be read or written in 32-bit units.

Address: <ADCCn_base> + y × 80_H + $4AC_H$

Value after reset: 001F $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	ADTPR[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADTPR[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.47 ADCCnADTPRRy Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	When read, the value after reset is read. When written, write the value after reset.
20 to 0	ADTPR[20:0]	These bits set the cycle of A/D timer y. Setting range: 000000_H to $1FFFFF_H$

CAUTIONS

1. For details, see **Section 27.4.3.2(2), Starting scan groups by A/D timer triggers**.
2. To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.38 ADCCnULLMSRx ─ Scan Group x Upper Limit/Lower Limit Table Select Register

This is a register that selects the upper-limit/lower-limit table of SGx.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + $4B0_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ULS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 27.48 ADCCnULLMSRx Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When written, write the value after reset.
1, 0	ULS[1:0]	These bits select the upper-limit/lower-limit table. 0: Neither upper limit nor lower limit is checked. 1: Upper limit and lower limit are checked in ADCCnULLMTBR0. 2: Upper limit and lower limit are checked in ADCCnULLMTBR1. 3: Upper limit and lower limit are checked in ADCCnULLMTBR2. When the A/D-converted value is stored in DRn, use the upper-limit/lower-limit table selected by ULS[1:0] to check the value against the upper and lower limits.

CAUTION

To prevent malfunctions, this register should be set while the scan groups to be set are stopped.

27.3.39 ADCCnVCULLMTBR0 to ADCCnVCULLMTBR6 ─ Virtual Channel Threshold Table Registers 0 to 6

These are 32-bit readable/writable registers used for setting the upper and lower limits of the threshold values for the A/D converted value. One register among ADCCnVCULLMTBR0 to ADCCnVCULLMTBR6 is selected by the VCULLMTBS[2:0] bits of the ADCCnVCRj register.

Access: This register can be read or written in 32-bit units.

Address: ADCCnVCULLMTBR0 : <ADCCn_base> + 700_H
ADCCnVCULLMTBR1 : <ADCCn_base> + 704_H
ADCCnVCULLMTBR2 : <ADCCn_base> + 708_H
ADCCnVCULLMTBR3 : <ADCCn_base> + $70C_H$
ADCCnVCULLMTBR4 : <ADCCn_base> + 710_H
ADCCnVCULLMTBR5 : <ADCCn_base> + 714_H
ADCCnVCULLMTBR6 : <ADCCn_base> + 718_H

Value after reset: 7FFE 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VCULMTB[15:0]															
Value after reset	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VCLLMTB[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 27.49 ADCCnVCULLMTBR0 to ADCCnVCULLMTBR6 Registers Contents

Bit Position	Bit Name	Function
31 to 16	VCULMTB[15:0]	Upper limit threshold table entry These bits specify the upper limit for A/D converted values. Setting range: 0000_H to $7FFE_H$ One of the threshold table registers 0 to 6 is selected for each virtual channel by ADCCnVCRj.VCULLMTBS[2:0]. Entries in VCULMTB[15:0] are handled as signed fixed-point format regardless of the format selected for ADCCnDRj. In additive A/D conversion mode (CNVCLS[2:0] = 4_H), these bits specify the value for comparison with the sum of the converted values.
15 to 0	VCLLMTB[15:0]	Lower limit threshold table entry These bits specify the lower limit for A/D converted values. Setting range: 0000_H to $7FFE_H$ One of the threshold table registers 0 to 6 is selected for each virtual channel by ADCCnVCRj. VCULLMTBS[2:0]. Entries in VCLLMTB[15:0] are handled as signed fixed-point format regardless of the format selected for ADCCnDRj. In additive A/D conversion mode (CNVCLS[2:0] = 4_H), these bits specify the value for comparison with the sum of the converted values.

27.3.40 ADCCnSGVCPRx ─ Scan Group x Virtual Channel Pointer Register

This is a register for specifying the pointers for start and end of the virtual channel.

Access: This register can read or written in 16-bit units.

Address: <ADCCn_base> + x × 80_H + $4A0_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	VCEP[5:0]						—	—	VCSP[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.50 ADCCnSGVCPRx Register Contents

Bit Position	Bit Name	Function
15 to 14	Reserved	When read, the value after reset is read. When written, write the value after reset.
13 to 8	VCEP[5:0]	Scan group x virtual channel end pointer (mirror register to ADCCnSGVCEPx).
7 to 6	Reserved	When read, the value after reset is read. When written, write the value after reset.
5 to 0	VCSP[5:0]	Scan group x virtual channel start pointer (mirror register to ADCCnSGVCSPx).

For details of the functions, see **Sections 27.3.32, ADCCnSGVCSPx ─ Scan Group x Start Virtual Channel Pointer**, and **27.3.33, ADCCnSGVCEPx ─ Scan Group x End Virtual Channel Pointer**.

27.4 Function

27.4.1 Method of A/D Conversion

A/D conversion is performed for each scan group. Trigger signals for A/D conversion are prepared for the number of scan groups. When a trigger signal (SGx_TRG) is input, A/D conversion is performed for the signals of virtual channels assigned to the scan group in ascending order. A/D conversion for each virtual channel is completed or A/D conversion for all virtual channels assigned to the scan group is completed, an A/D completion interrupt (ADInx) occurs.

The multiscan mode repeats A/D conversion for the specified number of times when a trigger signal is input. The continuous scan mode repeats A/D conversion unlimitedly when a trigger signal is input.

The signed fixed-point format or signed integer format can be selected as the data format for A/D conversion. For details about the bit alignment in the data format, see **Section 27.3.5, ADCCnDRj — Data Register j**.

The following figure shows an operation example of conversion.

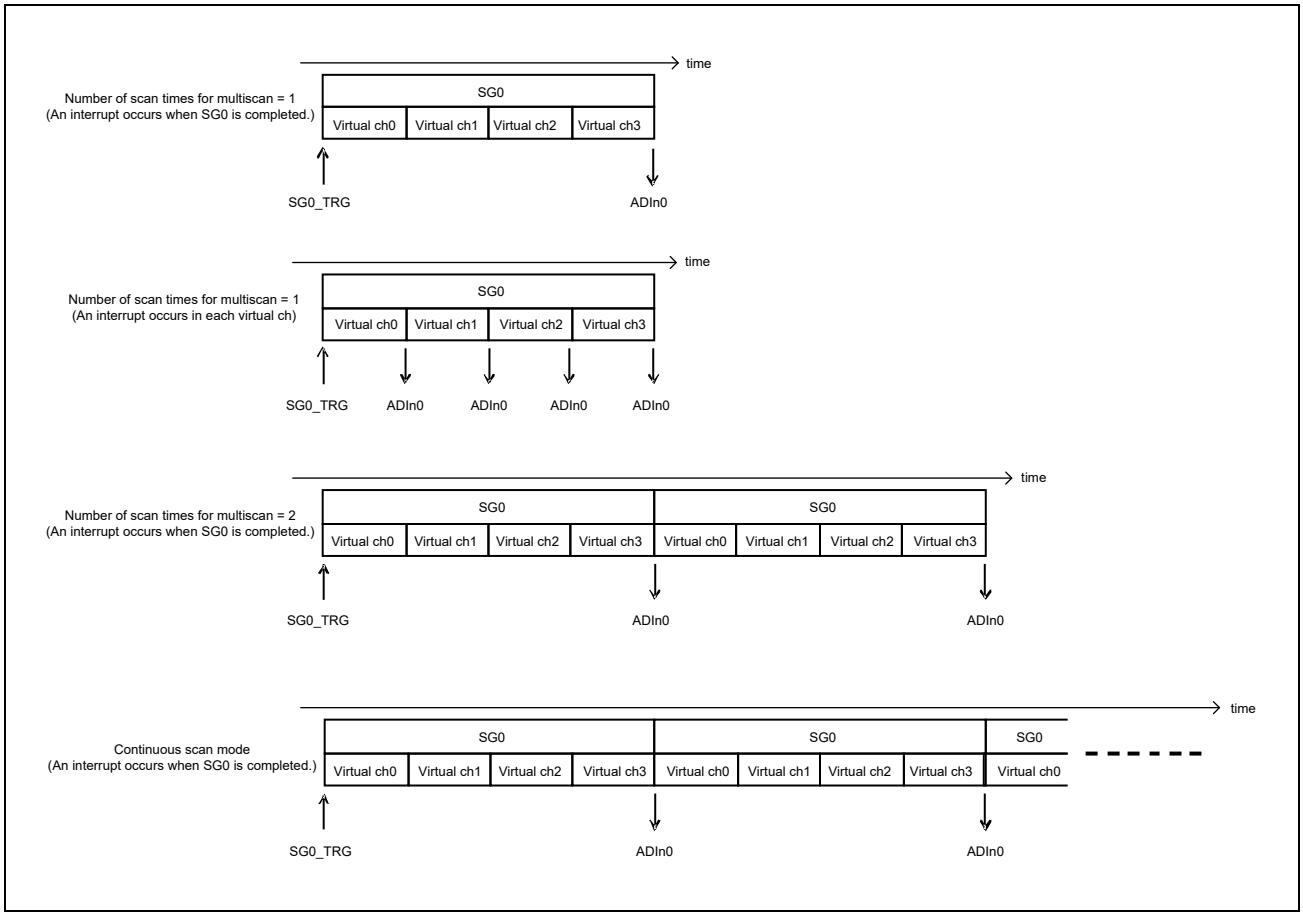


Figure 27.8 Operation Example of A/D Conversion

27.4.2 A/D Conversion Function

27.4.2.1 Normal A/D Conversion Function

The normal A/D conversion function performs A/D conversion for the analog signals of a physical channel.

27.4.2.2 Simultaneous track and hold function

This function holds multiple analog signals and performs A/D conversion for them. Analog signals to be simultaneously held can be assigned to two groups, T&H groups A and B, and can be held in a different timing for each group.

A physical channel that has a T&H circuit is assigned to a virtual channel and scan group, and the scan group is assigned to a T&H group. Note that SG0 cannot be assigned to a T&H group.

The method of T&H hold has the automatic sampling mode. This mode starts sampling automatically when A/D conversion for held analog signals is completed. If you do not specify the automatic sampling mode, you need to perform sampling by software.

The following figure is an operation example of the simultaneous track and hold function.

As shown in **Figure 27.9**, if a trigger signal (hold trigger A) for SG2 is input during A/D conversion, the virtual channel for which A/D conversion is currently being performed is forcibly stopped. After the virtual channel is stopped and analog signals become stable, the analog signals are held by the T&H circuit. After the hold operation is completed, A/D conversion resumes according to the priority of the scan groups.

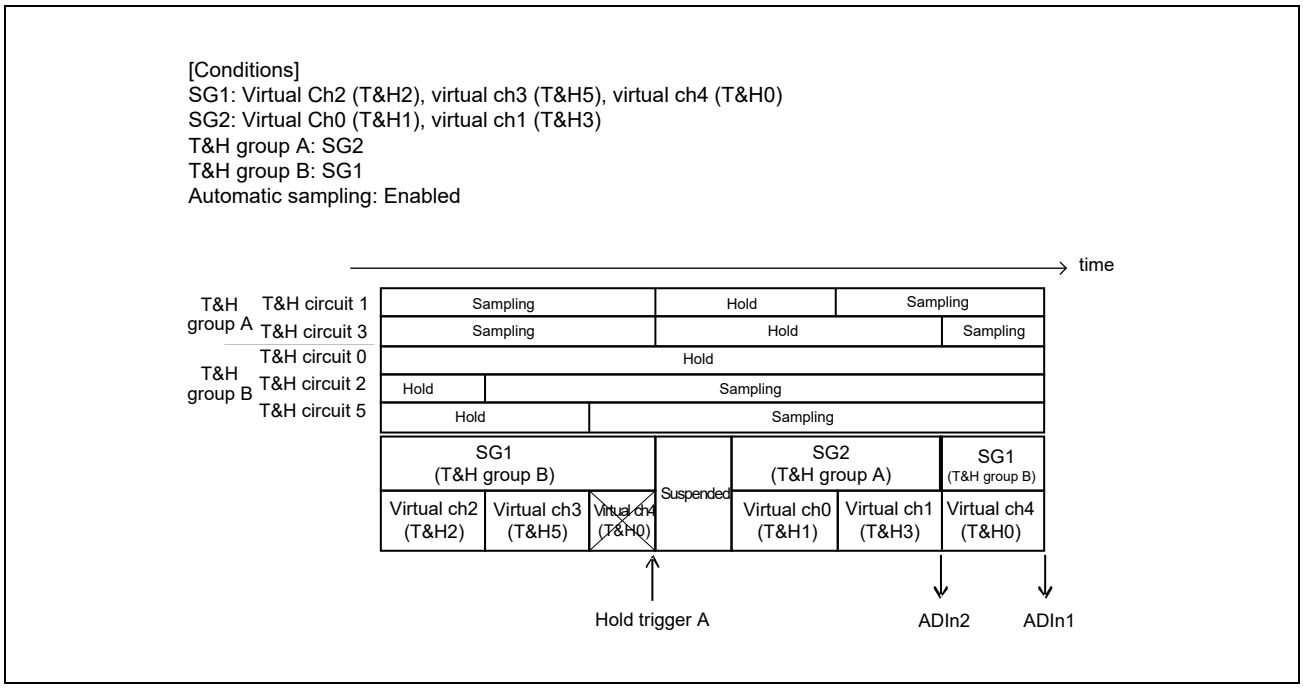


Figure 27.9 Operation Example of Simultaneous Track and Hold (High-Priority SG Hold Trigger Input)

For example, if the A/D conversion that is forcibly stopped is for SG3 as shown in **Figure 27.10**, A/D conversion for SG3 is first performed and then A/D conversion for SG2 is performed because SG3 has a higher priority.

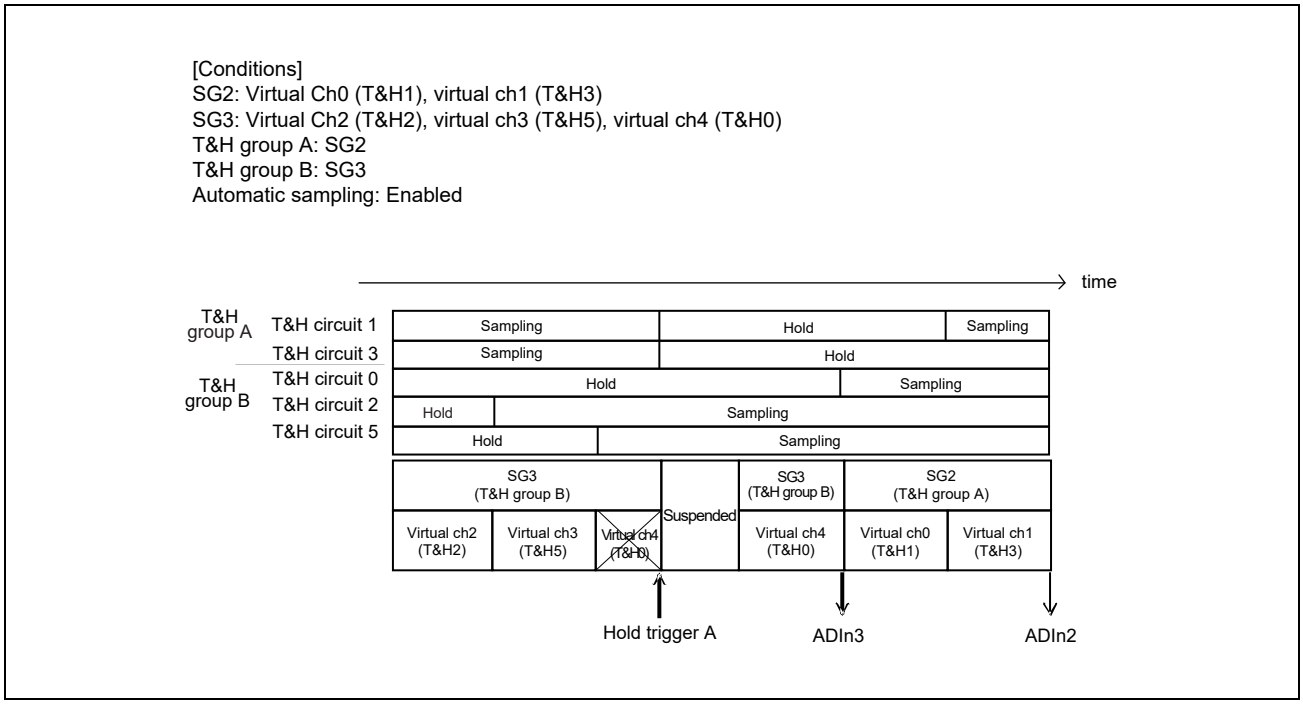


Figure 27.10 Operation Example of Simultaneous Track and Hold (Low-Priority SG Hold Trigger Input)

Also, when another trigger signal (hold trigger B) is input while A/D conversion is being forcibly stopped as shown in **Figure 27.11**, the stop time becomes longer than that in the case in which one trigger signal is input because of the waiting time until analog signals assigned to SG1 (T&H group B) are held.

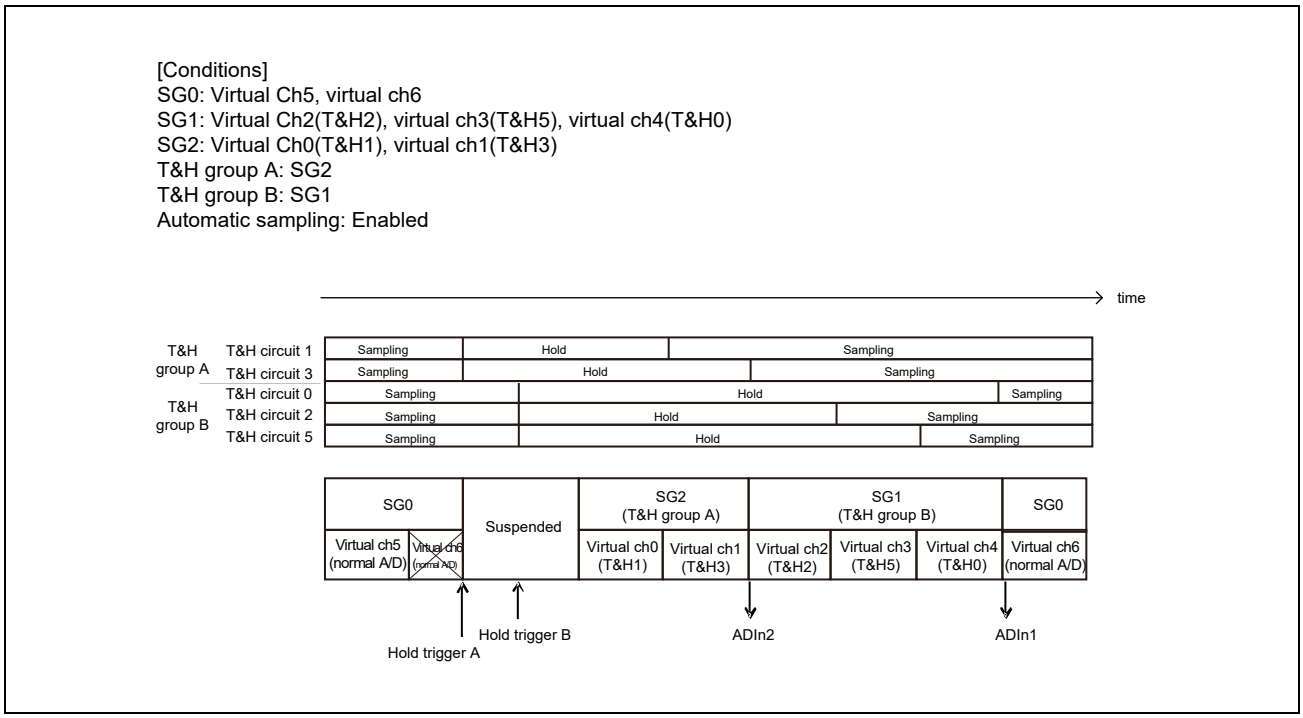


Figure 27.11 Operation Example of Simultaneous Track and Hold (Continuous Hold Trigger Input)

27.4.2.3 Addition A/D Conversion Function

This function performs A/D conversion two or four times in a row for analog signals of a physical channel and stores the added value in a data register. The addition count (two or four times) is common to all virtual channels.

27.4.2.4 Multicycle Scan Mode

This mode repeats A/D conversion for the specified number of times (1 to 256 times) for virtual channels assigned to a target SGx when a trigger is input.

27.4.2.5 Continuous Scan Mode

This mode repeats A/D conversion unlimitedly for virtual channels assigned to a target SGx when a trigger is input.

If you want to stop the continuous scan mode, stop A/D conversion according to the procedure for stopping A/D conversion. For details about the A/D conversion stop procedure, see **Section 27.5.3, Procedure for Stopping A/D Conversion**. You can stop all SGx and A/D timers by using the A/D halt register (ADCCnADHALTR register).

If a trigger for a lower-priority scan group is input to a scan group that is set to the continuous scan mode, the trigger is not accepted. Therefore, it is assumed that the continuous scan mode is set for SG0, which has the lowest priority.

27.4.3 Trigger Function

27.4.3.1 Input Selection of Triggers for Scan Groups

The following triggers can be selected as an A/D conversion start trigger for each scan group. Disable trigger input for unused scan groups.

Table 27.51 List of Trigger Supports

	HW Trigger		SW Trigger		
	SGx Trigger (SGx_TRG)	A/D Timer Trigger (SGy_TRG)	A/D Conversion Trigger	Hold Trigger	A/D Timer Trigger
SG0	√	×	√	×	×
SG1, SG2	√	×	√	√	×
SG3, SG4	√	√	√	√	√

Note: √: Supported, ×: Not supported

27.4.3.2 Starting Scan Groups by HW triggers

HW triggers include SGx triggers and A/D timer triggers.

(1) Starting scan groups by SGx triggers

A/D conversion can be started by starting SGx by a SGx_TRG signal. The simultaneous track and hold function holds analog signals by a hold trigger and then performs A/D conversion for them.

A SGx_TRG signal is input from PIC2D. PIC2D controls trigger signals from each unit or pin with masks and outputs them as SGx_TRG signals. For details about trigger factors, see **Section 24.3.3.1, ADCC Trigger Select Function**.

(2) Starting scan groups by A/D timer triggers

A SGy_TRG signal starts an A/D timer (free run) and an A/D timer trigger signal is output for each underflow of the counter as shown in **Figure 27.12**. This A/D timer trigger signal allows A/D conversion for SGy to start at regular intervals. However, when you use the simultaneous track and hold function, A/D timer trigger cannot be used.

An A/D timer is counted by clkad.

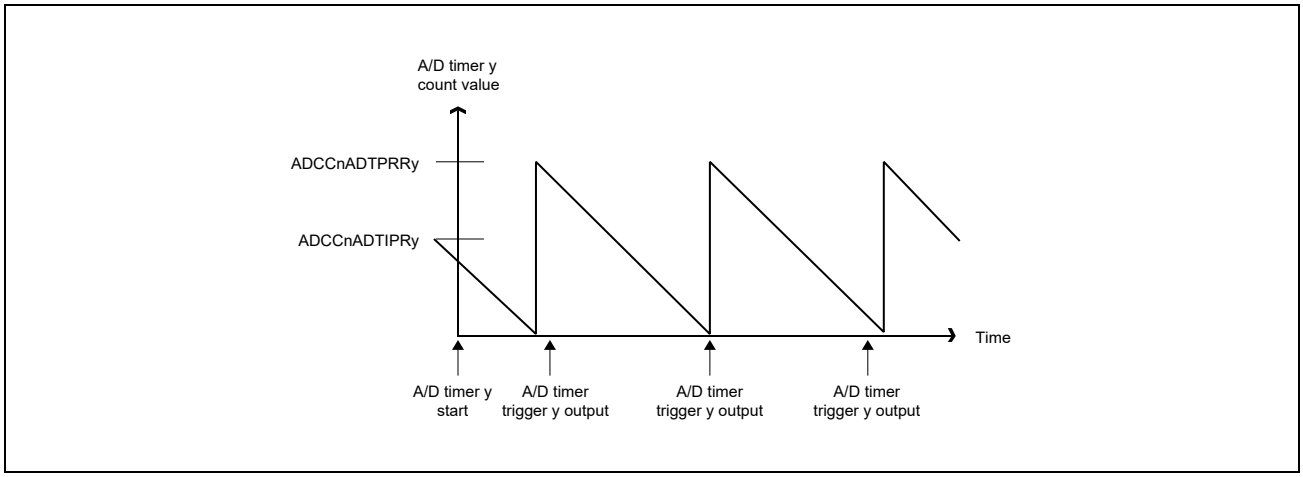


Figure 27.12 Operation Example of an A/D Timer

CAUTION

If you set the A/D timer initial phase register y (ADCCnADTIPRy register) to 0, an A/D timer trigger is output simultaneously with the start of an A/D timer. Also, if you set the A/D timer cycle register (ADCCnADTPRRy register) to 0, an A/D timer trigger is output for each clock.

27.4.3.3 Starting Scan Groups by SW Triggers

SW triggers include A/D conversion triggers, hold triggers, and A/D timer triggers. Note that before enabling each trigger, you should confirm that the target SG trigger is disabled and the target scan group is stopped.

(1) A/D conversion trigger

This trigger function can simultaneously start multiple SGx in ADCC0, ADCC1, and ADCC2, and start A/D conversion as shown in **Figure 27.13**. Also, this trigger function can start A/D conversion for SGx individually.

(2) A/D timer trigger

This trigger function can simultaneously start multiple A/D timer y in ADCC0, ADCC1, and ADCC2 as shown in **Figure 27.13**. Also, this trigger function can start A/D timer y individually.

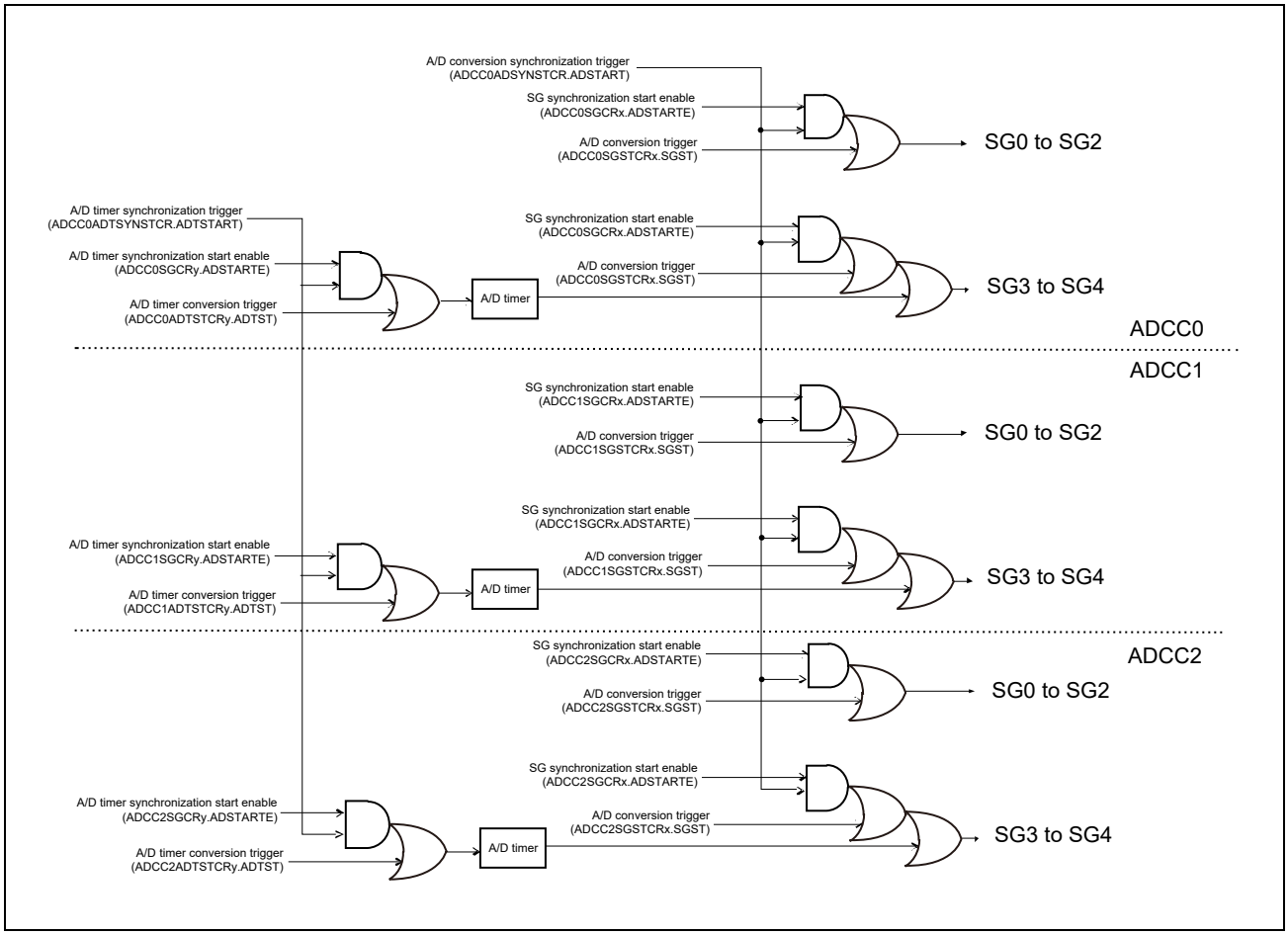


Figure 27.13 Functional Diagram of the SW Trigger Function

(3) Hold trigger

This trigger function can start hold at any timing as shown in **Figure 27.14**. Make sure that you perform 30 clkad or more samplings before starting hold.

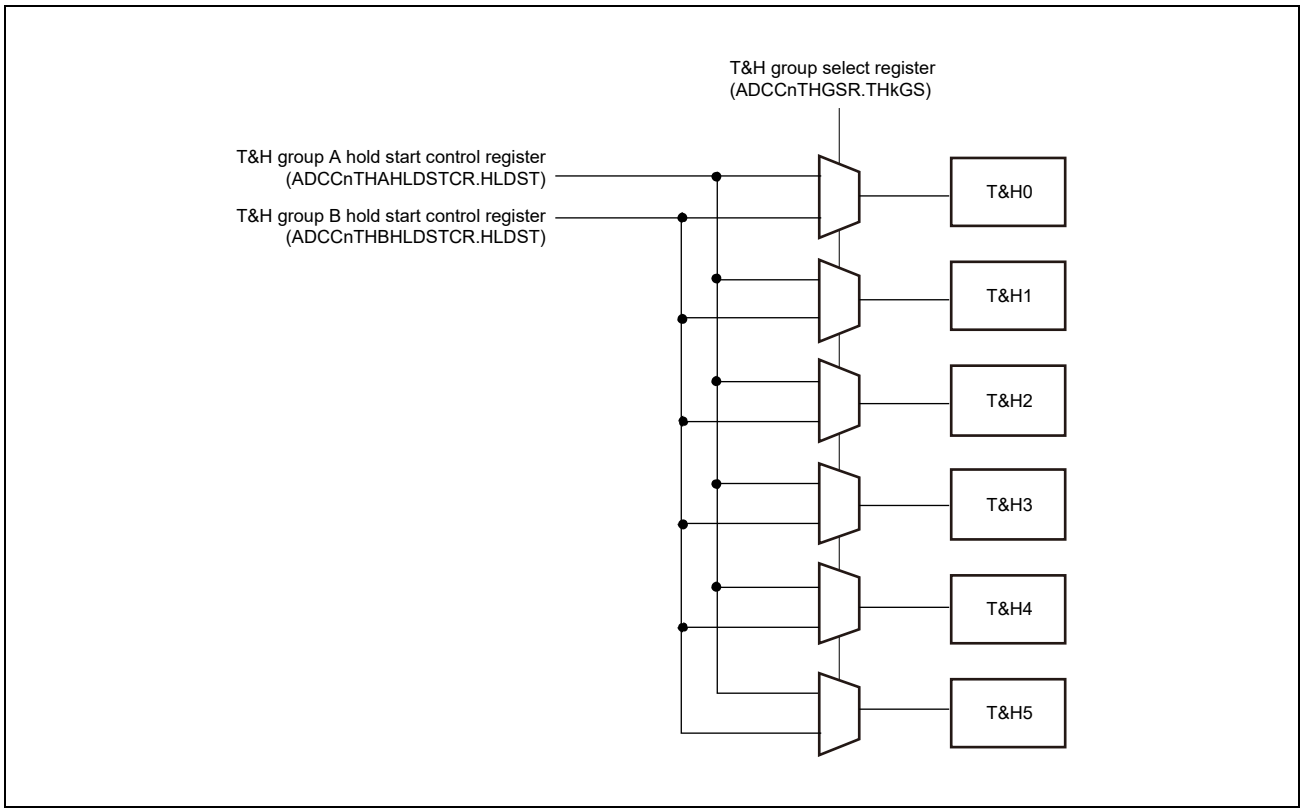


Figure 27.14 Functional Diagram of the Hold Trigger Function

27.4.4 Suspend Function

When a higher-priority scan group request is received during the processing of a lower-priority scan group, the suspend function suspends the lower-priority A/D conversion and performs the higher-priority A/D conversion. There are three types of suspend operation.

27.4.4.1 Synchronous Suspend Operation

If an A/D conversion trigger for a higher-priority scan group than the scan group for which A/D conversion is being performed occurs, after the conversion of the virtual channel for which A/D conversion is being performed is completed, A/D conversion for the higher-priority scan group is performed. After the A/D conversion for the higher-priority scan group is completed, A/D conversion resumes from the suspended virtual channel.

The following figures show operation examples.

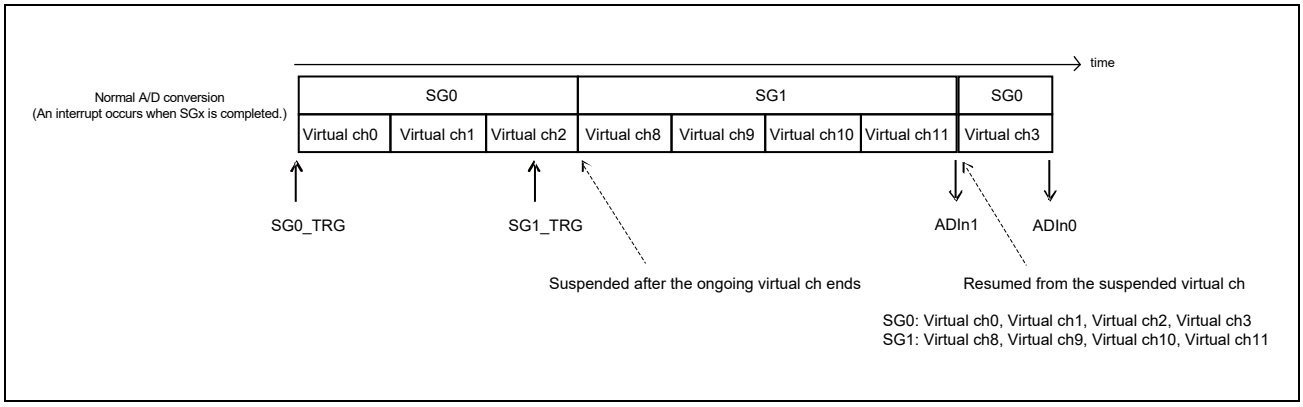


Figure 27.15 Operation Example of Synchronous Suspend (Normal A/D Conversion)

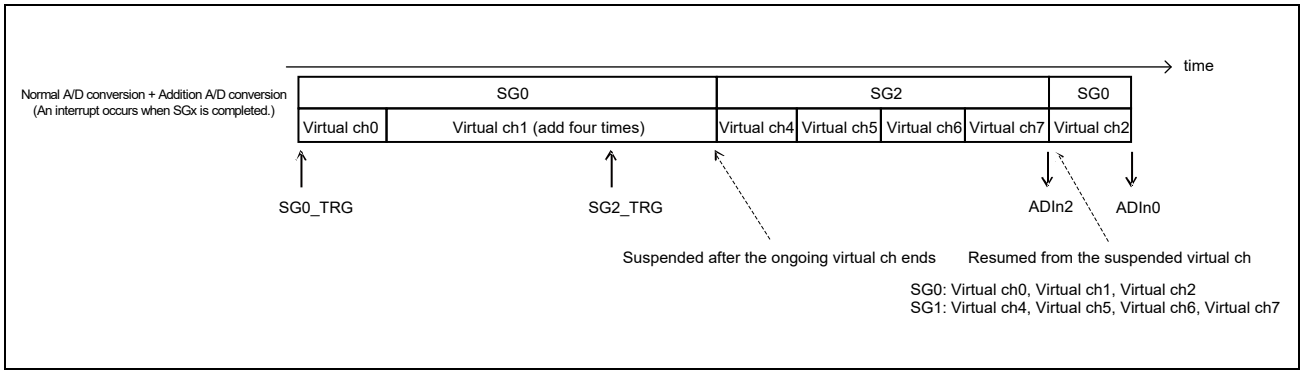


Figure 27.16 Operation Example of Synchronous Suspend (Normal A/D Conversion + Addition A/D Conversion)

27.4.4.2 Asynchronous Suspend Operation

If an A/D conversion trigger for a higher-priority scan group than the scan group for which A/D conversion is being performed occurs, the conversion for the virtual channel for which A/D conversion is being performed is immediately suspended, and A/D conversion for the higher-priority scan group is performed. After the A/D conversion for the higher-priority scan group is completed, A/D conversion for the suspended virtual channel starts from the beginning.

The following figures show operation examples.

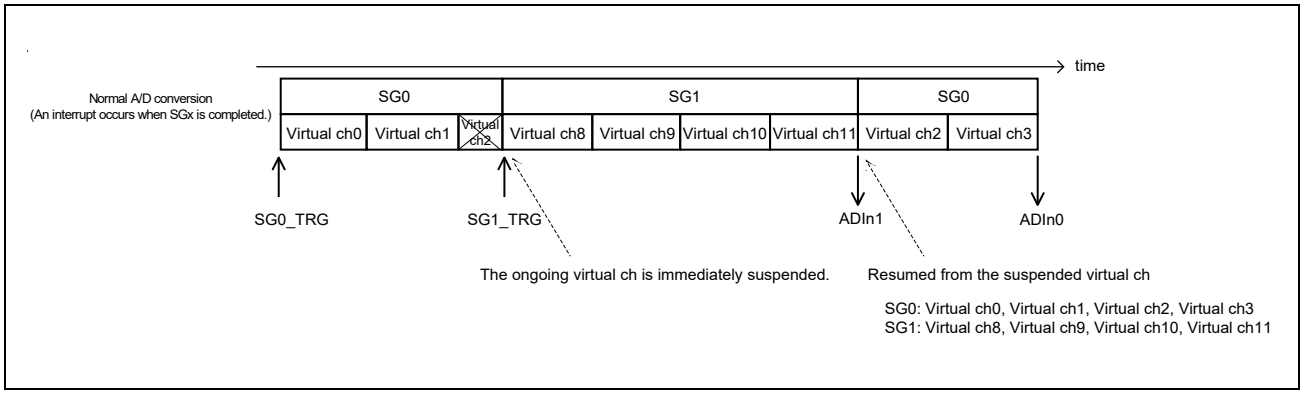


Figure 27.17 Operation Example of Asynchronous Suspend (Normal A/D Conversion)

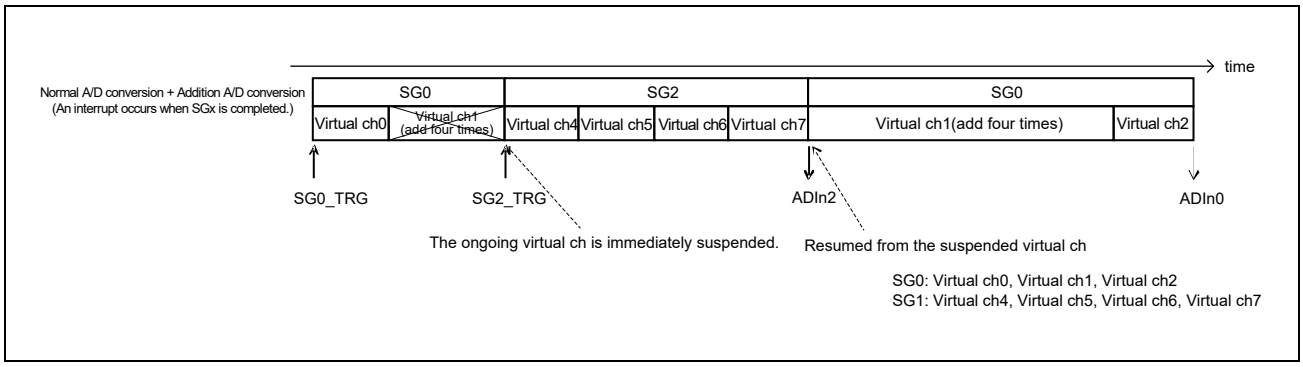


Figure 27.18 Operation Example of Asynchronous Suspend (Normal A/D Conversion + Addition A/D Conversion)

27.4.4.3 Synchronous and Asynchronous Combination Suspend Operation

If an A/D conversion trigger for a higher-priority scan group occurs during A/D conversion for SG0, asynchronous suspend operation is performed. If an A/D conversion trigger for a higher-priority scan group occurs during A/D conversion for a scan group other than SG0, synchronous suspend operation is performed. After the A/D conversion for the higher-priority scan group is completed, A/D conversion for the suspended virtual channel starts from the beginning.

The following figure shows an operation example.

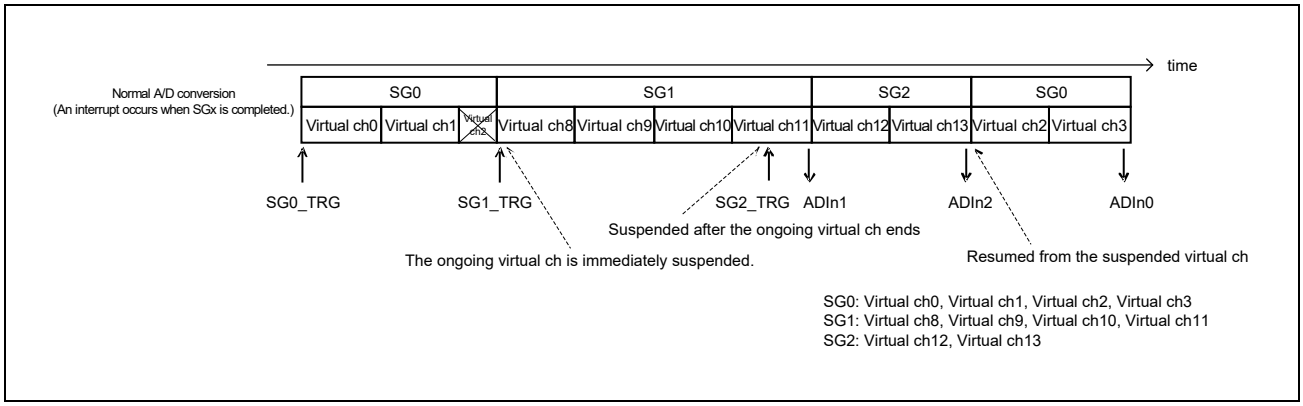


Figure 27.19 Operation Example of Synchronous and Asynchronous Combination Suspend

27.4.5 Interrupt Request Function

This function includes scan group x end interrupts and A/D error interrupts. An interrupt request signal is a pulse. You can start DMA/DTS by a scan group x end interrupt.

Interrupt output can be masked. Even if interrupt output is masked, the status register is set to 1 (an interrupt occurs).

27.4.5.1 Scan Group X End Interrupt

An end interrupt can occur at the following timing.

1. An interrupt occurs for each virtual channel.
   After A/D conversion for a virtual channel is completed, an interrupt occurs.
2. An interrupt occurs for each scan group.
   After A/D conversion for all the virtual channel assigned to the scan group is completed, an interrupt occurs.

In multiscan mode, an interrupt occurs in each A/D conversion for scan groups for a one-time scan. Therefore, if you perform multiscan two times in the above setting 2, an end interrupt occurs two times.

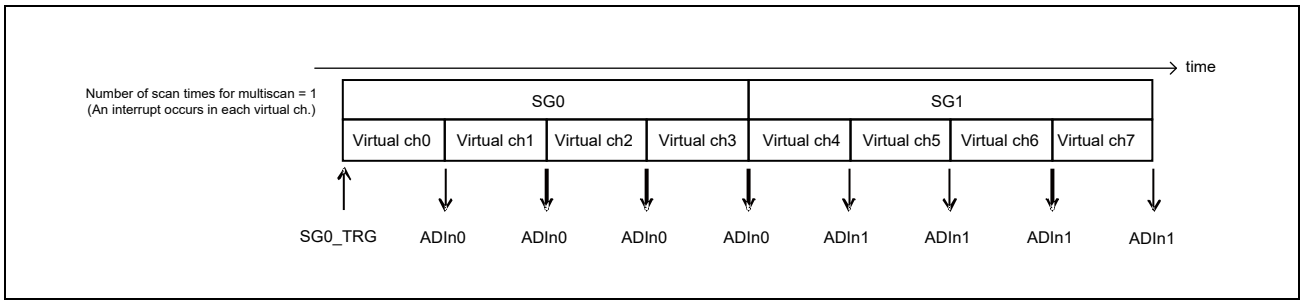


Figure 27.20 Scan Group X End Interrupt (An Interrupt Occurs for each Virtual Channel)

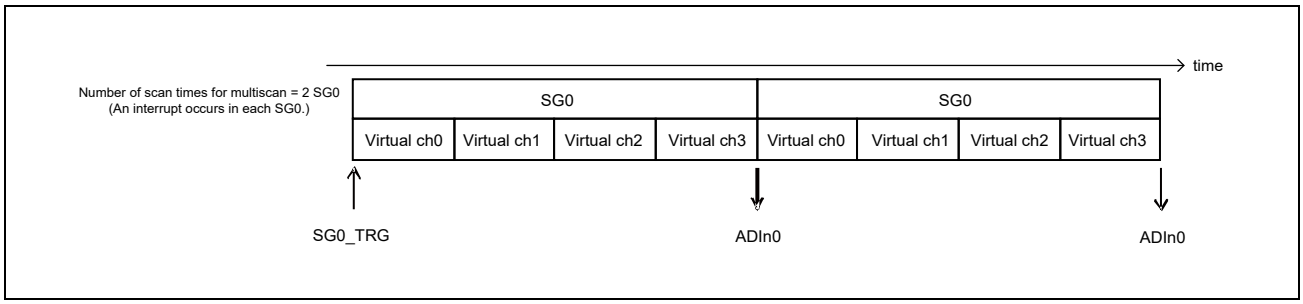


Figure 27.21 Scan Group X End Interrupt (An Interrupt Occurs for each SG)

27.4.5.2 A/D Error Interrupt Request

An A/D error interrupt can occur when the following A/D errors occur. The timing of the occurrence of A/D error interrupts is the same as that of end interrupts.

An A/D error interrupt request occurs when one of an upper-limit/lower-limit error, overwrite error, and an ID error occurs (OR condition). After an interrupt occurs, perform error status clear in the interrupt handler.

1. Upper-limit/lower-limit error
2. Overwrite error
3. ID error

(1) Upper-limit/lower-limit error

When an A/D converted value exceeds the specified upper-limit or lower-limit value, an error occurs. The number of the virtual channel in which an error occurs is retained in a register (ADCCnULER.ULECAP[5:0] bits). You can select one from three tables for the upper-limit or lower-limit value.

An upper-limit/lower-limit error is judged by the addition result that is stored in the data register.

(2) Overwrite error

When an A/D converted value has not been read (ADCCnDIRj.WFLG bit = 1) and the A/D converted value is updated (overwritten), an error occurs. The number of the virtual channel in which an error occurs is retained in a register (ADCCnOWER.OWECAP[5:0] bits).

(3) ID error

When a physical channel assigned to a virtual channel does not match with the physical channel for which conversion is actually performed, an error occurs. The number of the virtual channel in which an error occurs is retained in a register (ADCCnIDER.IDECAP[5:0] bits).

CAUTION

When an error has not been cleared, if the same error occurs again, the subsequent error information is discarded.

27.4.5.3 A/D Parity Error Trigger

A parity error interrupt can occur when a parity error occurs. A parity error trigger occurs when a data register (ADCCnDRj register) is read.

An A/D parity error trigger occurs by a parity error. An A/D parity error trigger is reported to the ECM.

When a data register is read, the contents are checked by the parity (ADCCnDIRj.PRTY bit) in the data supplementary information register. If a parity error is detected, the conversion is considered as an error. The number of the virtual channel in which an error occurs is retained in a register (ADCCnPER.PECAP[5:0] bits).

CAUTION

When an error has not been cleared, if the same error occurs again, the subsequent error information is discarded.

27.4.6 Function for Transferring Results of A/D Conversion to EMU

Each ADCC can output A/D conversion completion signals for virtual channels 0, 1, and 2, A/D converted data, and SG4 scan end signals to the EMU. An A/D conversion completion signal is a signal that indicates that the A/D conversion result value is stored in a data register and the A/D conversion is completed.

Data stored in data registers ADCCnDR0, 1, and 2 is output as A/D converted data. Note that the upper 4 bits of data to be output are fixed to 0 and the lower 12 bits are A/D converted data regardless of the data format setting (ADCCnADCR2.DFMT bit). Also, when the read and clear enable is enabled (ADCCnSFTCR.RDCLRE bit = 1), if the ADCCnDRj register or the ADCCnDIRj register is read, A/D converted data is cleared to 0000_H similarly to the ADCCnDR0, 1, and 2 registers.

CAUTION

When you use this function, virtual channels 0, 1, and 2 should be assigned to SG4.

Specify the settings so that a scan group end interrupt occurs for each scan group (The settings in which an interrupt occurs for each virtual channel are prohibited).

Do not use this function in the addition A/D conversion settings (ADCCnVCRj.CNVCLS[2:0] bits = 4_H).

27.4.7 Self-Diagnostic Functions

Each ADCC has the following self-diagnostic functions. The self-diagnostic function compares the conversion result with the expected value and confirms whether the result is as expected.

- Pin-level self-diagnostic function
- A/D conversion circuit self-diagnostic function
- Wiring-break detection self-diagnostic circuit function

27.4.7.1 Pin-Level Self-Diagnostic Function

This is a function to check an abnormal path from pins. This function checks an abnormal path from pins by specifying an even physical subchannel and odd physical subchannel as a set, setting a different voltage by the pin-level self-diagnostic control register (ADCCnTDCR register), and then executing A/D conversion. As shown in **Figure 27.22**, disconnect ADCC from analog input pins, apply the specified voltage, and then perform A/D conversion.

You can use a combination of AVSS, AVCC, and 1/2 × AVCC for different voltage. Also, you can use any physical channel for diagnosis.

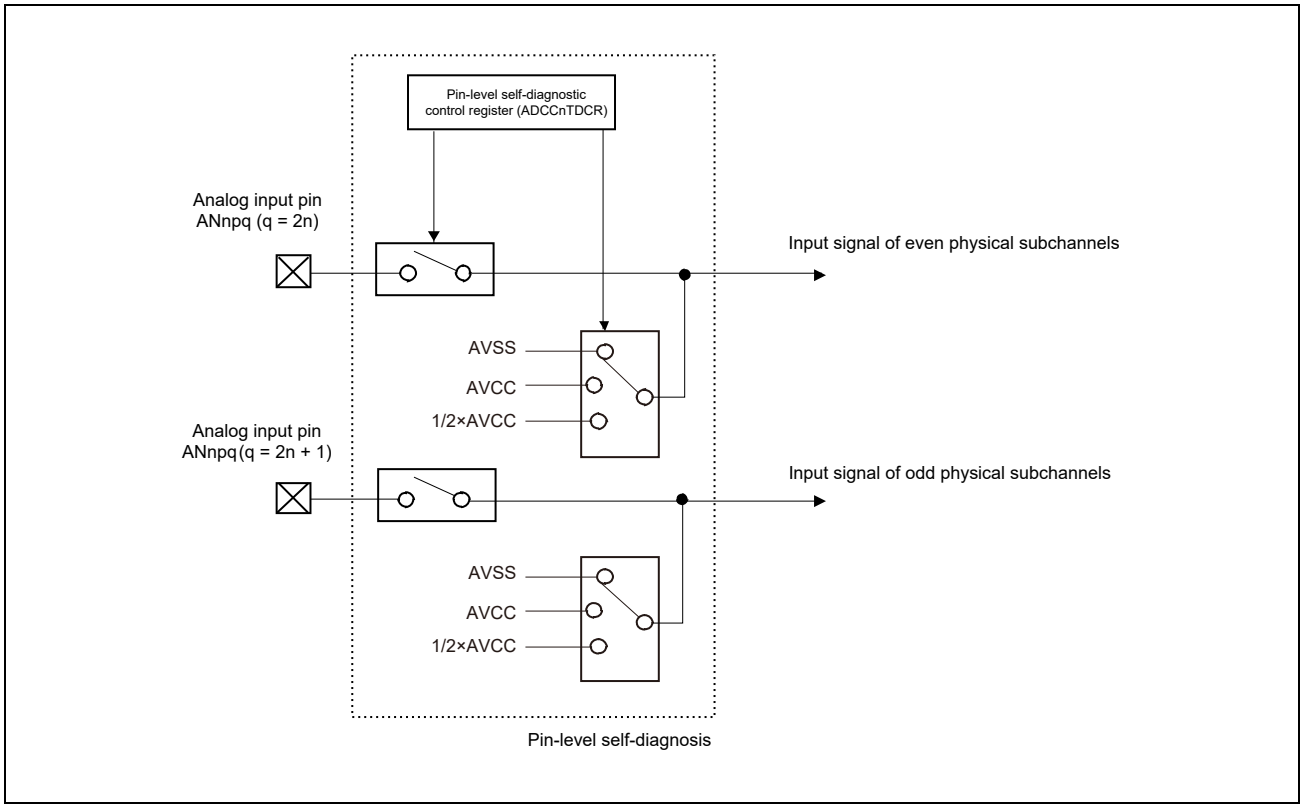


Figure 27.22 Functional Diagram of Pin-Level Self-Diagnosis

27.4.7.2 A/D Conversion Circuit Self-Diagnosis Function

This is a function to input a self-diagnostic voltage level and check the A/D conversion circuit from the result of A/D conversion. You can select a self-diagnostic voltage level from AVREFH × 1, AVREFH × 3/4, AVREFH × 1/2, AVREFH × 1/4, and AVREFH × 0.

27.4.7.3 Wiring-Break Detection Self-Diagnosis Circuit Function

This is a function to detect a wiring-break in a pin due to solder separation.

Discharge the target analog pin for the specified time in the wiring-break detection control register (ADCCnODCR register) and then perform A/D conversion. If the conversion result attenuates to approximately 0 V, you can determine that a wiring-break is present.

27.5 Procedure

27.5.1 Procedure for Setting A/D Conversion

Figure 27.23 shows the flow of A/D conversion settings, and **Figure 27.24** shows the flow of initial settings. Perform the initial settings in a state in which trigger factors for all scan groups are disabled and all scan groups and T&H are stopped. If they are active, perform the A/D conversion stop settings.

Set the value after reset for the setting values of unused functions.

In the following flow, before starting A/D conversion, pin-level self-diagnosis is performed in a state in which A/D conversion is disabled, and then A/D conversion start settings are performed.

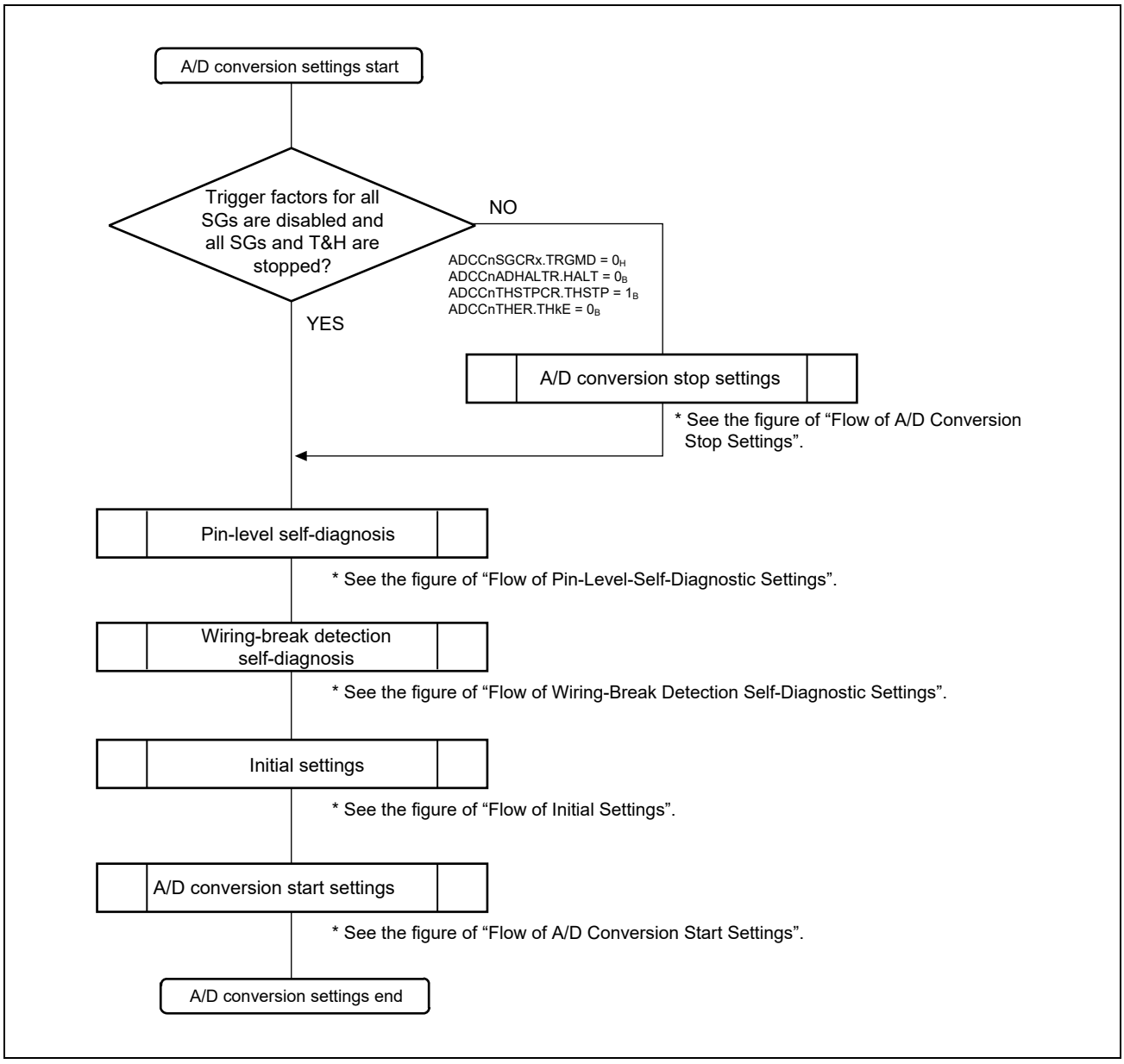


Figure 27.23 Flow of A/D Conversion Settings

The following initial settings flow is a flow for performing the basic settings of ADCC such as the A/D conversion mode settings.

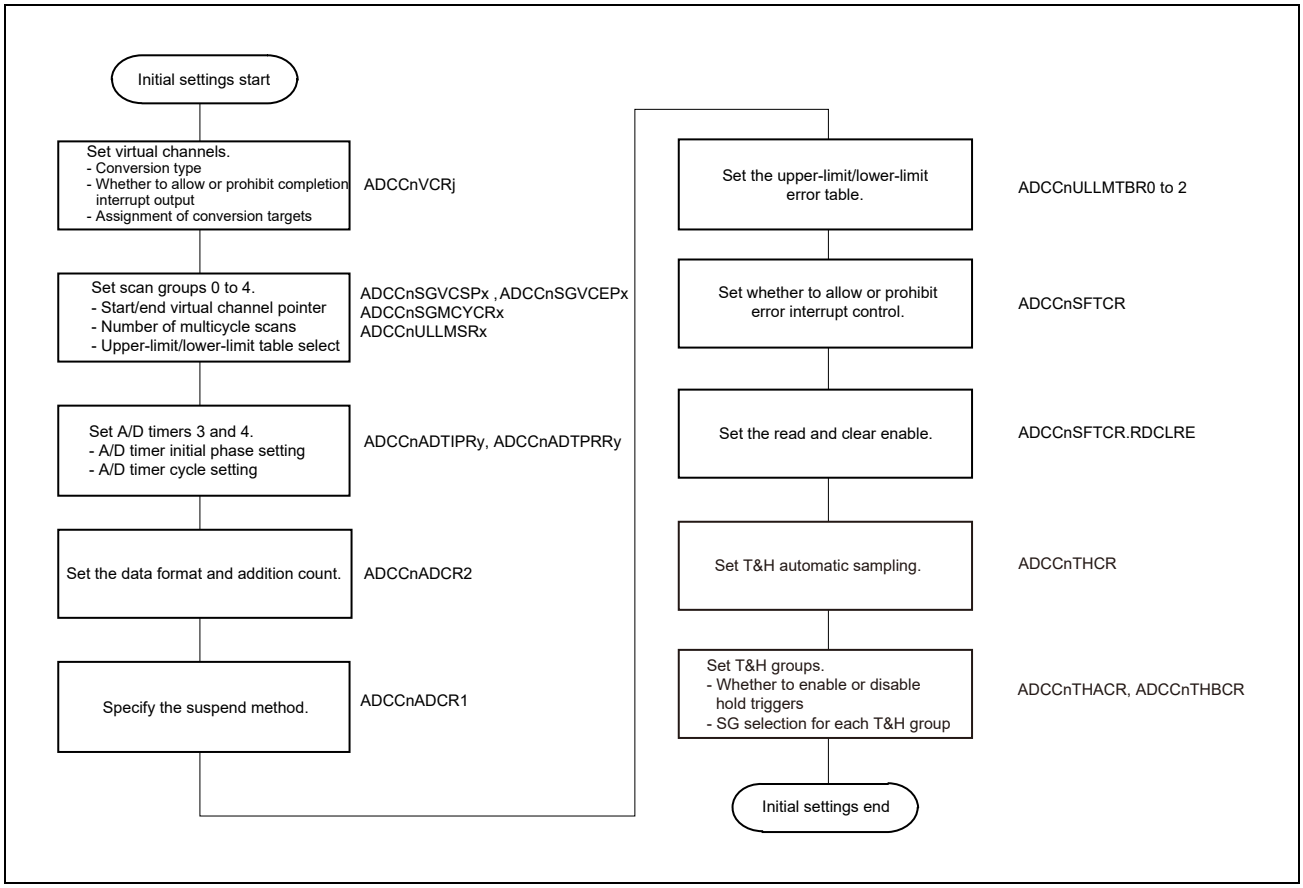


Figure 27.24 Flow of Initial Settings

27.5.2 Procedure for Starting A/D Conversion

Figure 27.25 shows the flow of A/D conversion start settings.

The following flow is a flow for starting A/D conversion operation by using a HW trigger. When you use the simultaneous track and hold function, you should perform hold after starting T&H sampling and then waiting 30 clkad or more samplings.

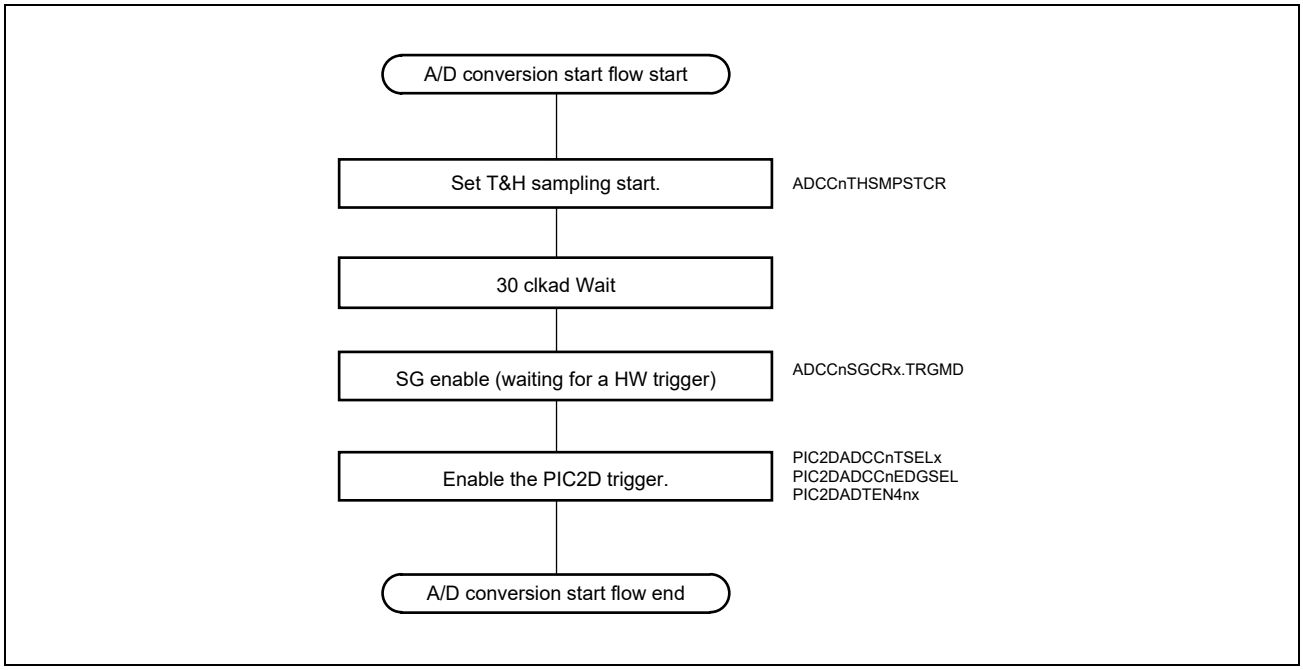


Figure 27.25 Flow of A/D Conversion Start Settings

27.5.3 Procedure for Stopping A/D Conversion

Figure 27.26 shows the flow of A/D conversion stop settings.

The following flow is a flow for stopping A/D conversion operation by disabling triggers for all scan groups and stopping operation of all scan groups and T&H.

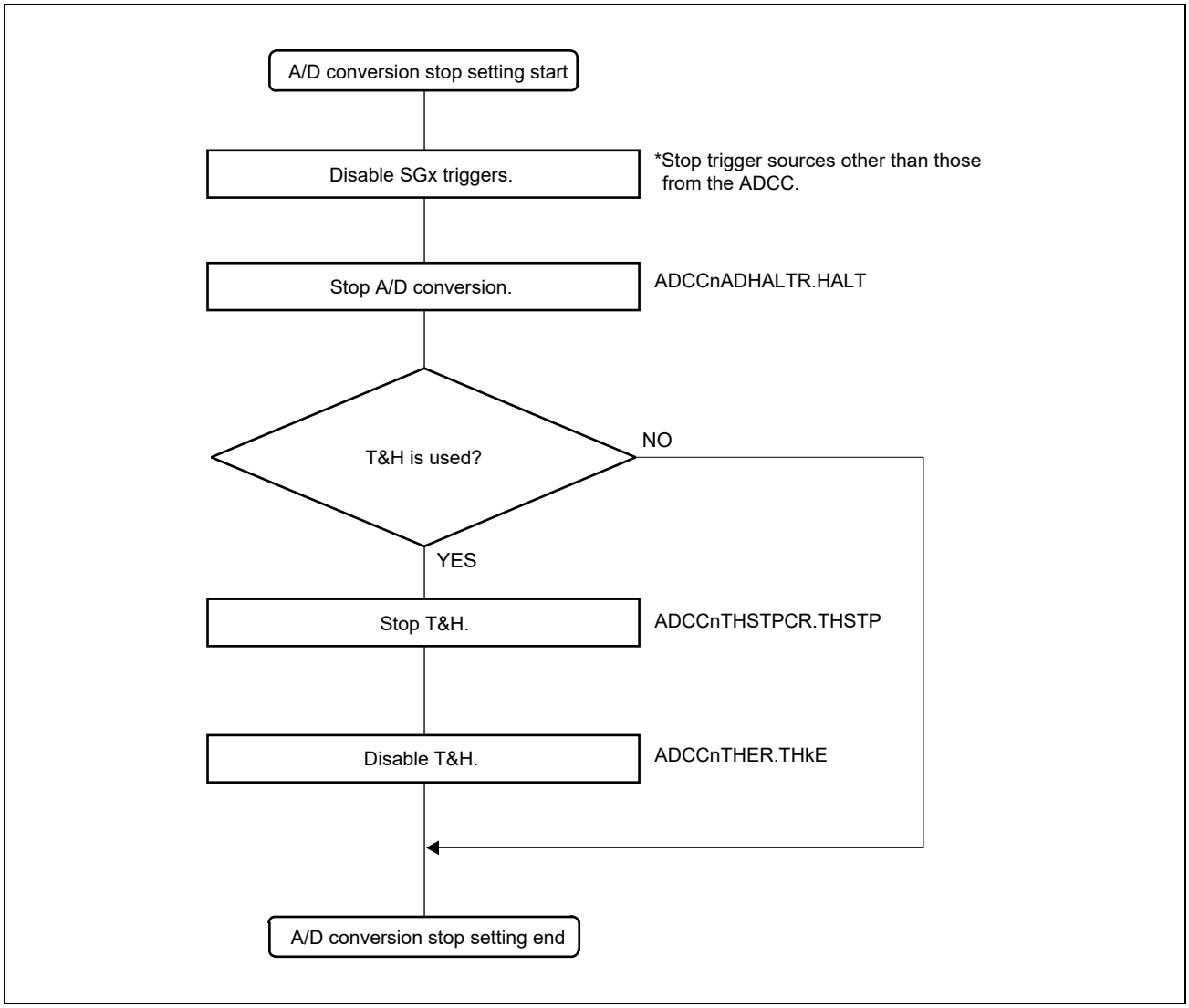


Figure 27.26 Flow of A/D Conversion Stop Settings

27.5.4 Procedure for Setting Pin-Level Self-Diagnosis

Figure 27.27 shows the flow of pin-level self-diagnostic settings.

The following flow uses an example in which all the pins of ADCC0 are assigned to virtual channels, and pin-level self-diagnosis is performed by switching applied voltage to even physical subchannels and odd physical subchannels. Also, it is assumed that the flow is performed before starting A/D conversion. Do not perform a self-diagnosis while the current is injected into the analog input pin.

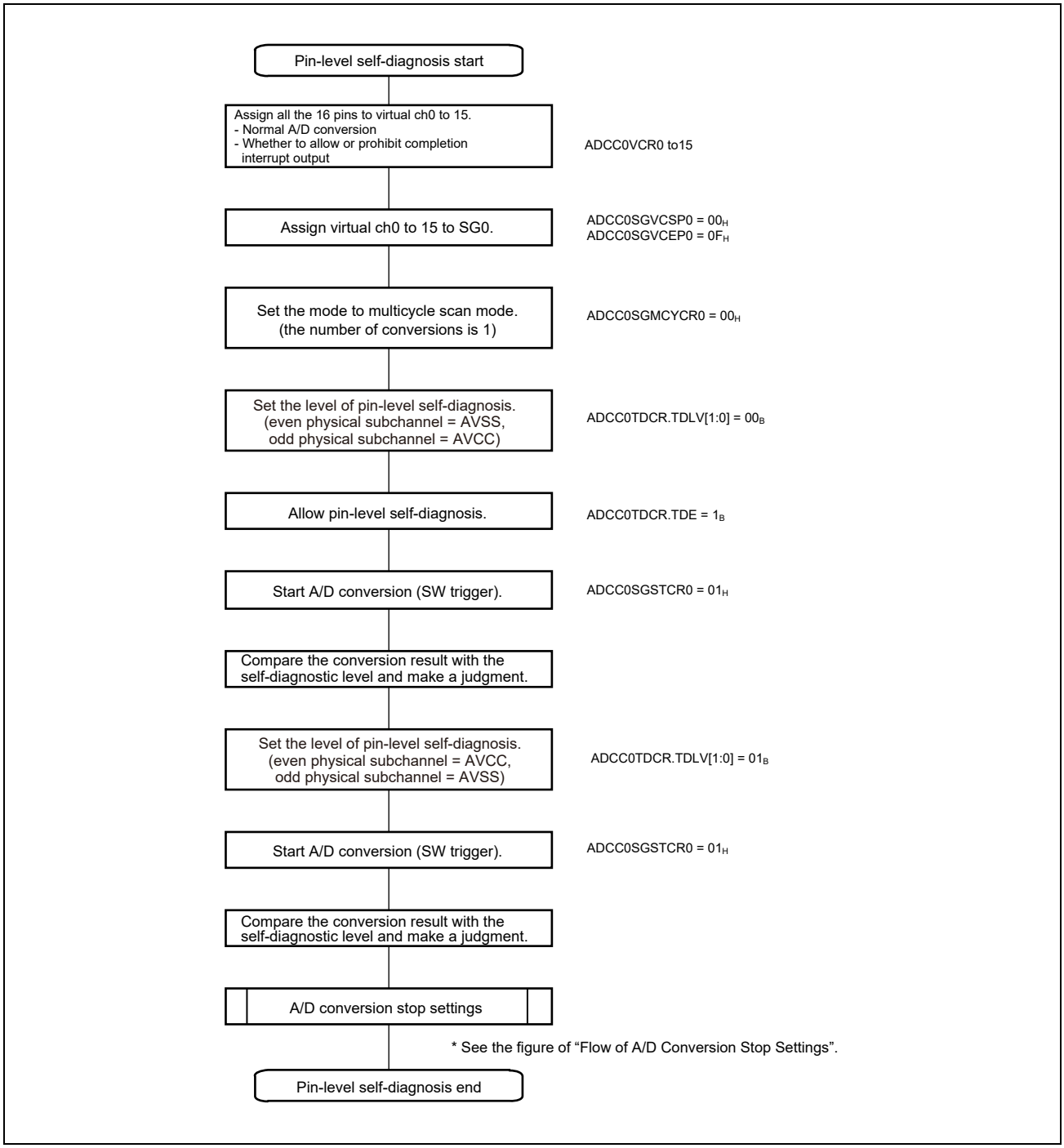
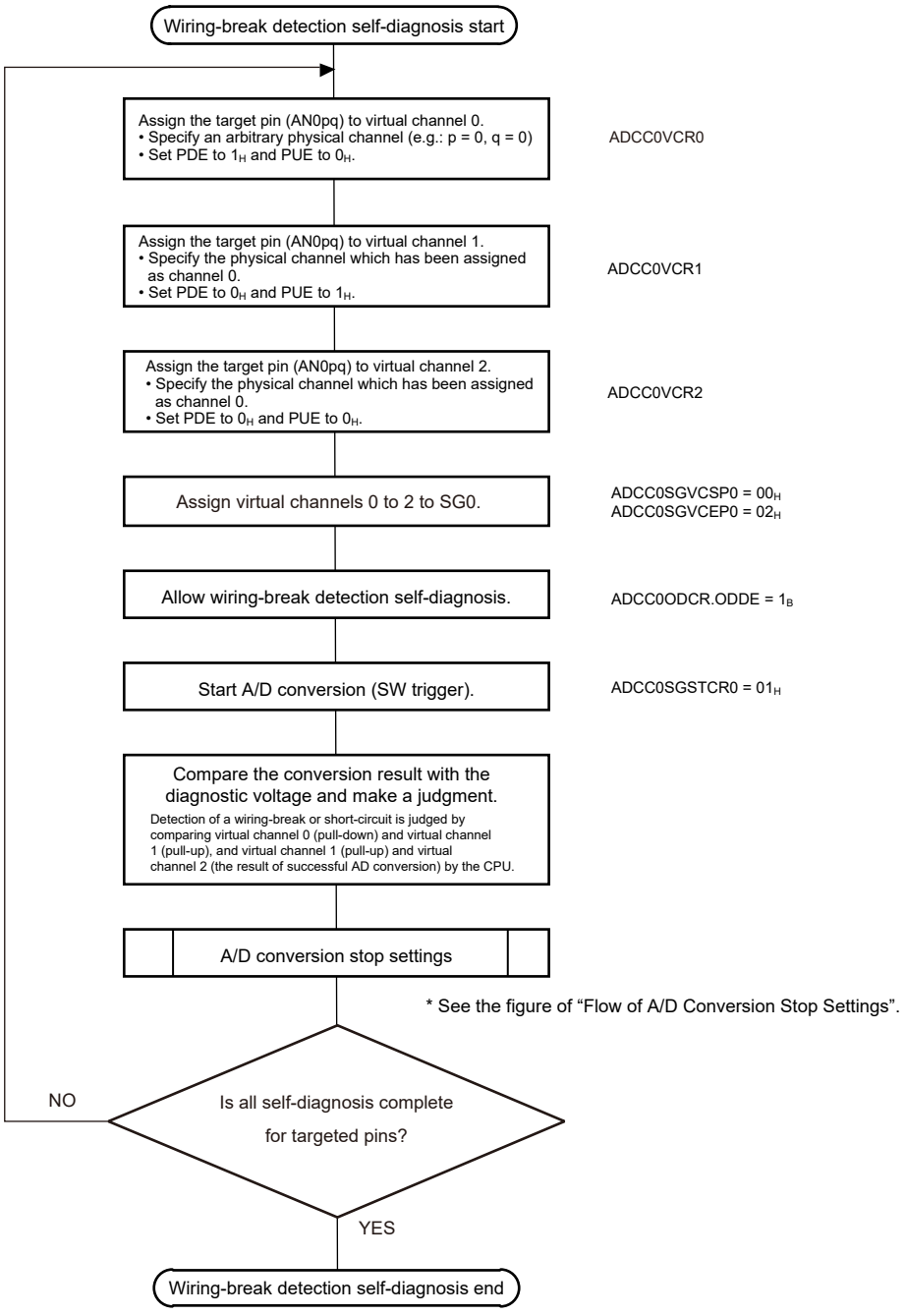


Figure 27.27 Flow of Pin-Level-Self-Diagnostic Settings

27.5.5 Procedure for Setting Wiring-Break Detection Self-Diagnosis

Wiring-break detection is a facility for detecting wiring breaks in ANI. Both pull-down and pull-up methods are used for detection. In pull-down method, if a wiring-break occurs, the result of the AD conversion is attenuated to approximately 0V, which is detected as an abnormal value. Therefore, the user can judge that a wiring-break has been detected. In pull-up method, if a wiring-break occurs, the result of the AD conversion is booted to approximately 5V, which is detected as an abnormal value. Therefore, the user can judge that a wiring-break has been detected.

Figure 27.28 shows the flow of wiring-break detection self-diagnostic settings. The following flow uses an example in which each pin of ADCC0 is assigned to virtual channels, and wiring-break detection self-diagnosis is performed by applying diagnostic voltage (other than the neighborhood of 0 V) to targeted pins. Also, it is assumed that the flow is performed before starting A/D conversion.



Note: Perform wiring-break detection self-diagnosis on each pin one by one.

Figure 27.28 Flow of Wiring-Break Detection Self-Diagnostic Settings

27.6 Definition of A/D Conversion Accuracy

A/D conversion accuracy is defined below.

- Resolution
  Number of digital output codes of the A/D converter
- Quantization error
  An error essentially contained in A/D converters, which is given as 1/2 LSB (**Figure 27.29**).
- Offset error
  Deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from the minimum voltage value 000_H to 001_H. However, the quantization error is not included (**Figure 27.29**).
- Full-scale error
  Deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from FFE_H to FFF_H. However, the quantization error is not included (**Figure 27.29**).
- DNL (Differential nonlinear error)
  Deviation between the ideal digital output code width (Vq) and the actual digital output code width (Va), which is given as (Va - Vq) / Vq. However, the offset error, the full-scale error, and the quantization error are not included (**Figure 27.29**).
- INL (Integral nonlinear error)
  Deviation of the actual value from the ideal A/D conversion characteristics, from the zero voltage to the full-scale voltage, which is given as an integral of DNL from 000_H to a digital output code. However, the offset error, the full-scale error, and the quantization error are not included (**Figure 27.29**).
- Absolute accuracy
  Deviation between the digital value and the analog input value. The offset error, the full-scale error, the quantization error, DNL, and INL are included (**Figure 27.29**).

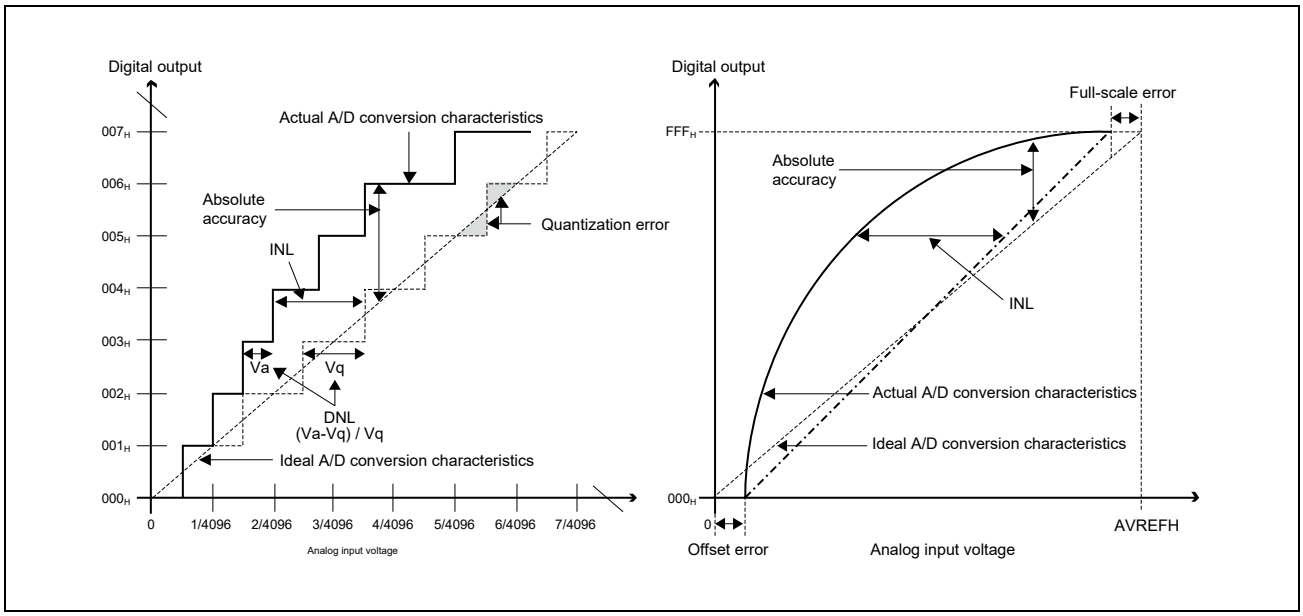


Figure 27.29 Definition of A/D Conversion Accuracy

27.7 Notes

27.7.1 Notes on Register Settings

1. When you use the simultaneous track and hold function, be sure to select asynchronous suspend.
2. When you use the simultaneous track and hold function, be sure to select the multicycle scan mode and set the number of scans to 1.
3. When you use the simultaneous track and hold function, be sure to hold first to perform A/D conversion.
4. When you use the simultaneous track and hold function, be sure to disable an AD timer trigger input to SGx to be selected in T&H group A or B.
5. Do not perform wiring-break detection in A/D conversion using the simultaneous track and hold function. Set the ADCCnTHER.THkE bit to 0 or the ADCCnTHSTPCR.THSTP bit to 1 before using the wiring-break detection function.
6. Do not perform a self-diagnosis while the current is injected into the analog input pin.

The following table shows notes on setting registers.

Before setting values for the following registers, set and check the contents of Check Steps.

Table 27.52 Notes on Setting Registers

Registers	Check Steps
ADCCnVCRj ADCCnADCR1 ADCCnADCR2 ADCCnSFTCR ADCCnTDCR ADCCnODCR ADCCnULLMTBR0 to ADCCnULLMTBR2	1. The hold trigger enable (ADCCnTHACR.HLDTE bit and ADCCnTHBCR.HLDTE bit) of T&H groups A and B is 0. 2. The scan group synchronization start enable (ADCCnSGCRx.ADSTARTE bit) of all scan groups is 0 and the trigger mode (ADCCnSGCRx.TRGMD bit) of all scan groups is 0. 3. The scan group status (ADCCnSGSRx.SGACT bit) of all scan groups is 0 (before starting scan groups).
ADCCnTHCR ADCCnTHGSR	1. The hold trigger enable (ADCCnTHACR.HLDTE bit and ADCCnTHBCR.HLDTE bit) of T&H groups A and B is 0. 2. The scan group status (ADCCnSGSRx.SGACT bit) of SGx specified in the scan group select (ADCCnTHACR.SGS[1:0] bits and ADCCnTHBCR.SGS[1:0] bits) of T&H groups A and B is 0 (before starting scan groups). 3. The ADCCnTHER.THkE bit of all T&H is 0 (all T&H are stopped).
ADCCnTHACR.HLDCTE bit, SGS[1:0] bit ADCCnTHBCR.HLDCTE bit, SGS[1:0] bit	1. The hold trigger enable (ADCCnTHACR.HLDTE bit and ADCCnTHBCR.HLDTE bit) of T&H groups A and B is 0. 2. The scan group synchronization start enable (ADCCnSGCRx.ADSTARTE bit) of all scan groups is 0 and the trigger mode (ADCCnSGCRx.TRGMD bit) of all scan groups is 0. 3. The scan group status (ADCCnSGSRx.SGACT bit) of all scan groups is 0 (before starting scan groups). 4. The ADCCnTHER.THkE bit of all T&H is 0 (all T&H are stopped).
ADCCnTHER	1. The hold trigger enable (ADCCnTHACR.HLDTE bit and ADCCnTHBCR.HLDTE bit) of T&H groups A and B is 0. 2. The scan group status (ADCCnSGSRx.SGACT bit) of SGx specified in the scan group select (ADCCnTHACR.SGS[1:0] bits and ADCCnTHBCR.SGS[1:0] bits) of T&H groups A and B is 0 (before starting scan groups).
ADCCnSGCRx.SCANMD bit, ADIE bit ADCCnSGMCYCRx ADCCnULLMSRx ADCCnSGVCSPx ADCCnSGVCEPx	1. When the scan group select (ADCCnTHACR.SGS[1:0] bits) of T&H group A is specified to SGx, the hold trigger enable (ADCCnTHACR.HLDTE bit) of T&H group A is 0. 2. When the scan group select (ADCCnTHBCR.SGS[1:0] bits) of T&H group B is specified to SGx, the hold trigger enable (ADCCnTHBCR.HLDTE bit) of T&H group B is 0. 3. The scan group synchronization start enable (ADCCnSGCRx.ADSTARTE bit) of SGx is 0 and the trigger mode (ADCCnSGCRx.TRGMD bit) of SGx is 0. 4. The scan group status (ADCCnSGSRx.SGACT bit) of SGx is 0 (before starting scan groups).
ADCCnTHCR ADCCnTHACR ADCCnTHBCR ADCCnTHER ADCCnTHGSR ADCCnSGCRx ADCCnSGVCSPx ADCCnSGVCEPx	When setting the registers listed at left, write to the registers after they have been read. If this procedure is not followed, the written register value may not be correctly reflected in operations.

Section 28 A/D Converter Option (ADPA)

This section contains a generic description of the A/D converter option (ADPA).

The first part of the section describes all RH850/C1M-A specific properties such as the number of units, the register base address, etc. The remainder of the section describes the functions and registers of ADPA.

28.1 Features of RH850/C1M-A ADPA

28.1.1 Number of Units

This LSI has the following number of units of ADPA.

Table 28.1 Number of Units

Product	RH850/C1M-A2	RH850/C1M-A1
Number of units	1	1
Name	ADPAn (n = 0)	ADPAn (n = 0)

Table 28.2 Indices

Index	Meaning
m	Throughout this section, the individual TSG3 units are identified by the index “m” (m = 0 to 2).
j	Throughout this section, the individual counters incorporated in ADPA are identified by the index “j” (j = 0 to 23).
k	The individual ADC units referred to from this section are identified by the index “k” (k = 0 to 2).
n	Throughout this section, the individual ADPA units are identified by the index “n” (n = 0).
q	Throughout this section, the DMA request numbers is indicated by the index “q” (q = 0 to 23).

28.1.2 Register Base Address

ADPA base addresses are listed in the following table.

ADPA register addresses are given as offsets from the individual base address.

Table 28.3 Register Base Address

Base Address Name	Base Address
<ADPA0_base>	FFF4 0000_H

28.1.3 Clock Supply

The ADPA clock supply is shown below.

Table 28.4 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
ADPAn	PCLK	CLK_LSB (low-speed peripheral clock)

28.1.4 Interrupt Requests

The ADPA interrupt requests are listed below.

Table 28.5 Interrupts and DMA Requests

Unit Interrupt Signal	Interrupt/DMA Request Name (Outline)	Interrupt Number	DMA Trigger Number*1		DTS Trigger Number*1	
			1st	2nd	1st	2nd
INTADPA0	ADPA control notification interrupt 0	333	—	—	—	—
INTADPA1	ADPA control notification interrupt 1	334	—	—	—	—
INTADPA2	ADPA control notification interrupt 2	335	—	—	—	—
DRQADPA0	ADPA DMA request 0	—	—	19	—	19
DRQADPA1	ADPA DMA request 1	—	—	20	—	20
DRQADPA2	ADPA DMA request 2	—	—	21	—	21
DRQADPA3	ADPA DMA request 3	—	—	22	—	22
DRQADPA4	ADPA DMA request 4	—	—	23	—	23
DRQADPA5	ADPA DMA request 5	—	—	24	—	24
DRQADPA6	ADPA DMA request 6	—	—	49	—	49
DRQADPA7	ADPA DMA request 7	—	—	50	—	50
DRQADPA8	ADPA DMA request 8	—	—	53	—	53
DRQADPA9	ADPA DMA request 9	—	—	54	—	54
DRQADPA10	ADPA DMA request 10	—	—	55	—	55
DRQADPA11	ADPA DMA request 11	—	—	56	—	56
DRQADPA12	ADPA DMA request 12	—	—	57	—	57
DRQADPA13	ADPA DMA request 13	—	—	58	—	58
DRQADPA14	ADPA DMA request 14	—	—	59	—	59
DRQADPA15	ADPA DMA request 15	—	—	60	—	60
DRQADPA16	ADPA DMA request 16	—	—	68	—	68
DRQADPA17	ADPA DMA request 17	—	—	69	—	69
DRQADPA18	ADPA DMA request 18	—	—	70	—	70
DRQADPA19	ADPA DMA request 19	—	—	71	—	71
DRQADPA20	ADPA DMA request 20	—	—	103	—	103
DRQADPA21	ADPA DMA request 21	—	—	104	—	104
DRQADPA22	ADPA DMA request 22	—	—	105	—	105
DRQADPA23	ADPA DMA request 23	—	—	106	—	106

Note 1. 1st: Primary channel, 2nd : Secondary channel

28.1.5 Reset Sources

ADPA reset sources are listed in the following table. ADPA is initialized by the reset source below.

Table 28.6 Reset Source

Unit Name	Reset Source
ADPAn	All reset sources

28.1.6 External Input/Output Signals

ADPA does not have external I/O signals.

28.2 Overview

28.2.1 Functional Overview

ADPA provides the following functionalities.

(1) Hi-Z control based on the upper and lower threshold values for AD conversion results

This function controls the outputs from the TSG3 units, whether to be placed into a high-impedance state or not, according to the number of counts of threshold judgment results based on the values read from ADC with the successive approximation method and the settings of upper and lower thresholds for those read values. Output from TSG3 are set for or released from a hi-Z state by the hardware. It can be also configured to be released from a hi-Z state by the software.

A hi-Z state is controlled according to the values of the error determination counter and normal recovery counter.

(2) Counting the number of met conditions for upper/lower threshold for ADC values

ADPA has twenty-four counters that count the number of met conditions. The individual counters are used for counting the results from the corresponding virtual channel registers of ADCC (VCR0 to VCR7) as follows.

- Counters 0 to 7 respectively corresponds to ADCC0.VCR0 to 7.
- Counters 8 to 15 respectively corresponds to ADCC1.VCR0 to 7.
- Counters 16 to 23 respectively corresponds to ADCC2.VCR0 to 7.

The judgment result from any counter above can be used for controlling outputs through any channel of TSG3s.

See **Figure 28.1** and **Figure 28.2** for connections between the modules.

28.2.2 Block Diagram

The relationships between ADPA, ADC, and TSG3 are shown below.

Outputs through the TSG3 units can be set for or release from a hi-Z state by using any judgment result read from the ADC. However, if hi-Z control by the timer option (TAPA) is also available, controlling by the timer option is prioritized. The result of VCRn threshold value judgment is output at every A/D conversion according to the settings of the vertical channel threshold value register. For details, see **Section 27, AD Converter (ADCC)**.

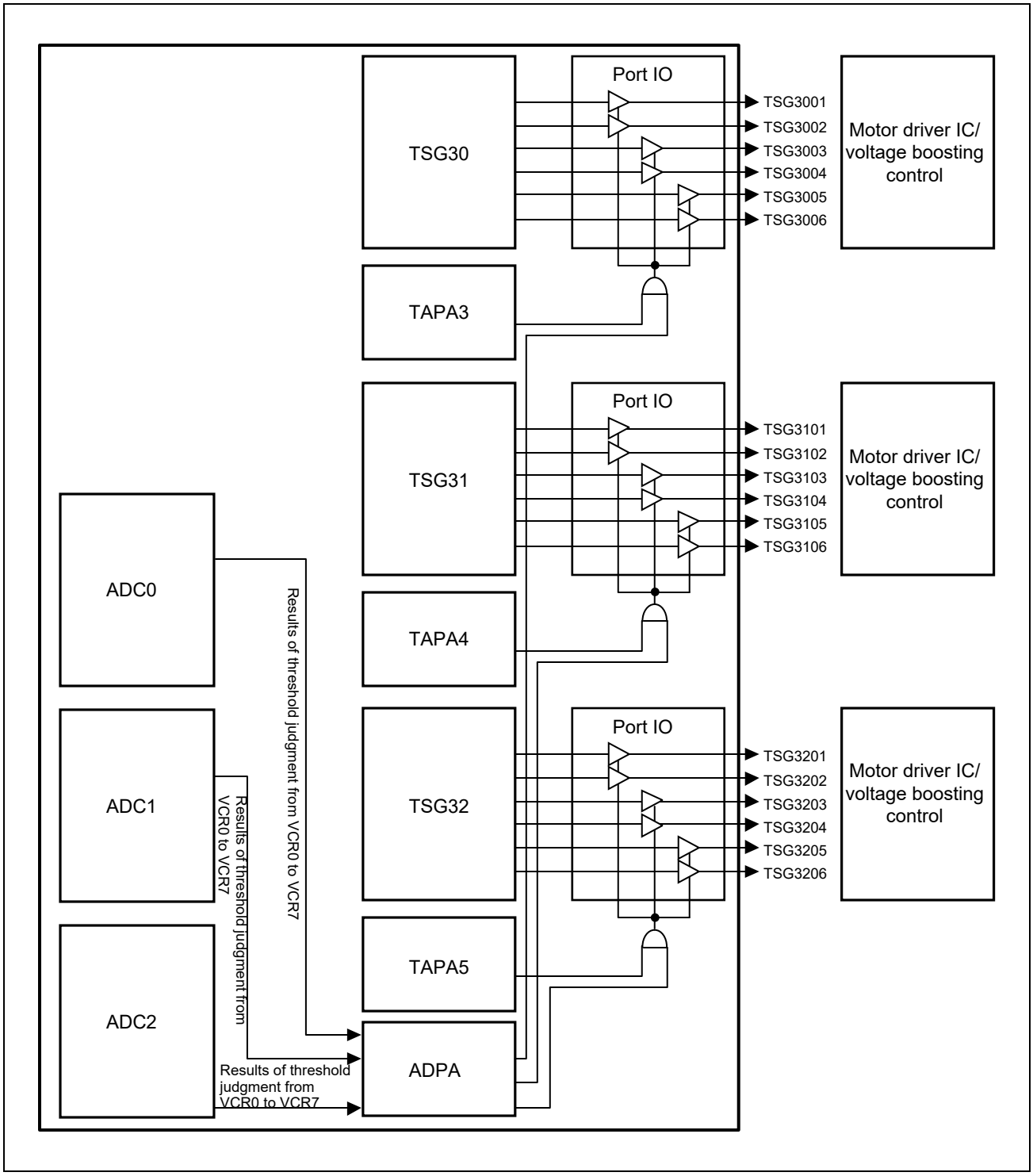


Figure 28.1 Block Diagram

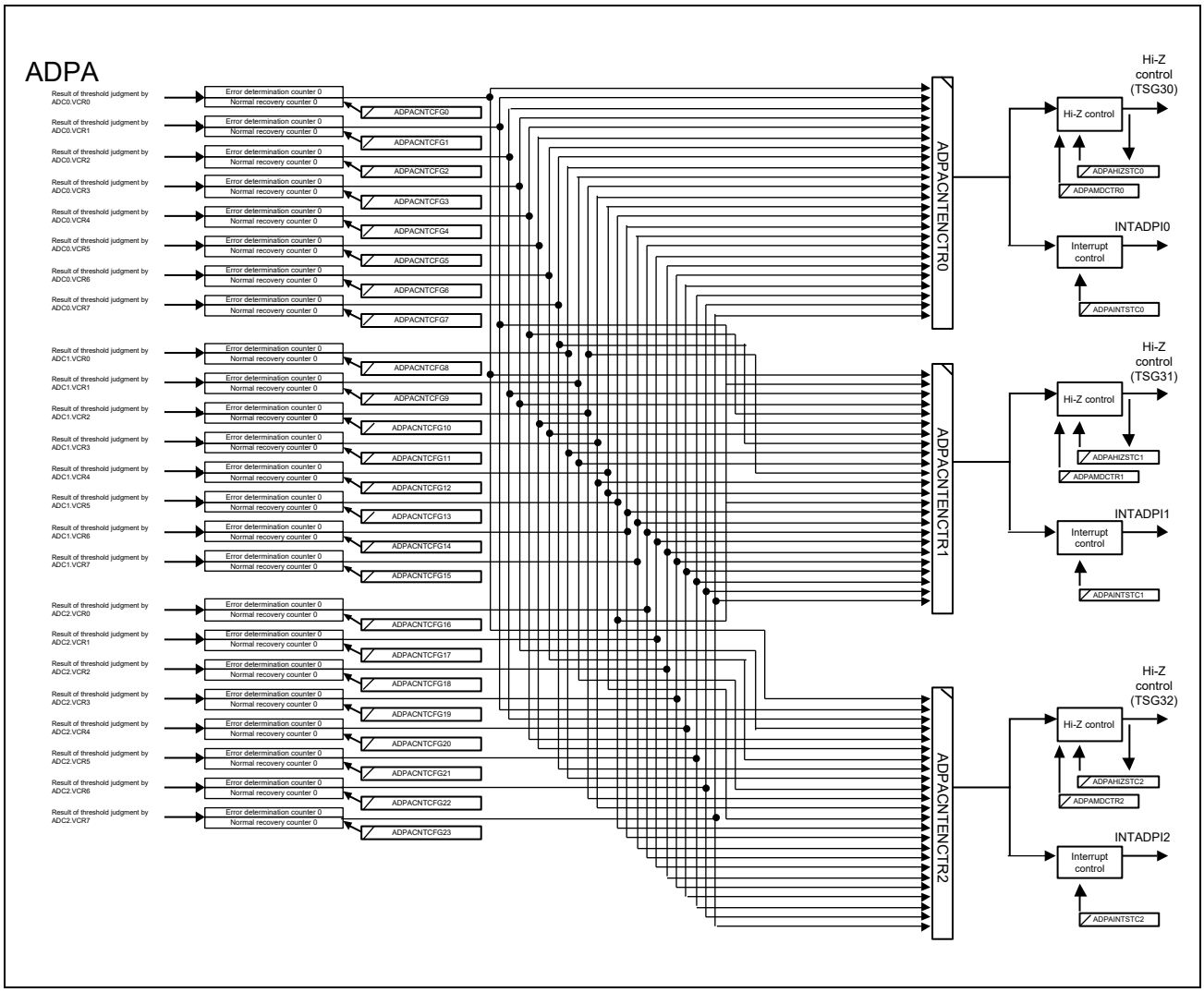


Figure 28.2 ADPA Internal Diagram

28.3 Registers

28.3.1 List of Registers

The ADPA registers are listed in the table below.

All registers of ADPA can be accessed only in 32-bit units. Do not make accesses to those registers in 8 or 16-bit units.

For <ADPAn_base>, see the register base address.

Table 28.7 List of Registers

Module Name	Register Name	Symbol	Address
ADPA	Hi-Z Control State Register m	ADPAHIZSTCm	<ADPAn_base> + 000_H + 100_H × m
ADPA	Interrupt Control Register m	ADPAINTSTCm	<ADPAn_base> + 008_H + 100_H × m
ADPA	Counter State Register m	ADPACNTSTSm	<ADPAn_base> + 010_H + 100_H × m
ADPA	Hi-Z Control Release Mode Selection Register m	ADPAMDCTRm	<ADPAn_base> + 020_H + 100_H × m
ADPA	Counter Enable Register m	ADPACNTENCTRm	<ADPAn_base> + 040_H + 100_H × m
ADPA	Hi-Z Control Request State Register m	ADPACNTRQSTSm	<ADPAn_base> + 044_H + 100_H × m
ADPA	Counter Setting Register j	ADPACNTCFGj	<ADPAn_base> + 800_H + 4_H × j
ADPA	Counter Control Register j	ADPACNTCTRj	<ADPAn_base> + 900_H + 4_H × j
ADPA	DMA Resource Selection Register q	ADPADMASELq	<ADPAn_base> + $C00_H$ + 4_H × q
ADPA	Test Pulse Injection Register 0	ADPATPUL0	<ADPAn_base> + $F00_H$
ADPA	Test Pulse Injection Register 1	ADPATPUL1	<ADPAn_base> + $F04_H$
ADPA	Test Pulse Injection Register 2	ADPATPUL2	<ADPAn_base> + $F08_H$

28.3.2 ADPAHIZSTCm – Hi-Z Control State Register m

This register indicates the state of hi-Z control. Writing 1 to this register while all bits of the hi-Z control request state register (ADPACNTRQSTSm) indicate “a request for the hi-Z state was not issued” releases the relevant output from the hi-Z state, if releasing by the software is selected by the mode register.

Access: Readable/writable in 32-bit units.

Address: <ADPAn_base> +100_H × m

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 28.8 ADPAHIZSTC Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When writing, write the value after a reset.
0	STS	Hi-Z Control State This bit indicates whether the output from TSG3x is in a hi-Z state or not. If releasing by the software is selected, writing 1 to this bit releases the output from the hi-Z state. 0 : Output from TSG3m is in a normal state. 1 : Output from TSG3m is in a hi-Z state.

28.3.3 ADPAINTSTCm – Interrupt Control Register m

This register controls the timing to notify interrupts.

Access: Readable/writable in 32-bit units.

Address: <ADPAn_base> + 8 + 100_H × m

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTEN	
Value after rese	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 28.9 ADPAINTSTC Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When writing, write the value after a reset.
1, 0	INTEN	Interrupt Control Register 00 : Interrupts are not notified. 01 : An interrupt is notified when the relevant output is released from the hi-Z state. 10 : An interrupt is notified when the relevant output is placed into a hi-Z state. 11 : An interrupt is notified when the relevant output is placed into or released from the hi-Z state.

28.3.4 ADPACNTSTSm – Counter State Register m

This register indicates the state of the individual counters.

Access: Readable in 32-bit units.

Address: <ADPAn_base> + 10_H + 100_H × m

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CNTSTS							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTSTS															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 28.10 ADPACNTSTS Register Contents

Bit Position	Bit Name	Function
31 to 24	Reserved	When read, the value after reset is returned.
23 to 0	CNTSTS	Counter State Register Each of these bits indicates whether the corresponding counters have made a request for the hi-Z state or not. 0 : The counter has made no request for the hi-Z state. 1 : The counter has made a request for the hi-Z state.

28.3.5 ADPAMDCTRm – Hi-Z Control Release Mode Selection Register m

This register is used to select whether to release the output from the hi-Z state by the hardware or the software.

Access: Readable/writable in 32-bit units.

Address: <ADPAn_base> +20_H + 100_H × m

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SFTEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 28.11 ADPAMDCTR Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When writing, write the value after a reset.
0	SFTEN	Hi-Z Control Release Mode Selection This bit selects whether to release the relevant output from a hi-Z state by the hardware or by the software. Releasing by the software does not make any effect if all counters (which have been enabled to make a request for the hi-Z state) have made a request for the hi-Z state. 0 : Hardware mode: the relevant output is released from the hi-Z state according to the control made by the counters. 1 : Software mode: the relevant output is released from the hi-Z state by the software.

28.3.6 ADPACNTENCTRm – Counter Enable Register m

This register enables and disables the counter to make a request for the hi-Z state.

Access: Readable/writable in 32-bit units.

Address: <ADPAn_base> + 40_H + 100_H × m

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CNTEN							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTEN															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.12 ADPACNTENCTR Register Contents

Bit Position	Bit Name	Function
31 to 24	Reserved	When writing, write the value after a reset.
23 to 0	CNTEN	Hi-Z Control Enable Counter Setting Each of these bits enables or disables the corresponding counter to make a request for the hi-Z state. 0 : The counter is disabled to make a request for the hi-Z state. 1 : The counter is enabled to make a request for the hi-Z state.

28.3.7 ADPACNTRQSTSm –Hi-Z Control Request Status Register m

This register indicates whether a request for the hi-Z state from the counter was issued or not.

Access: Readable in 32-bit units.

Address: <ADPAn_base> + 44_H + 100_H × m

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CNTRQ							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTRQ															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 28.13 ADPACNTRQSTS Register Contents

Bit Position	Bit Name	Function
31 to 24	Reserved	When read, the value after reset is returned.
23 to 0	CNTRQ	Hi-Z Control Request Each of these bits indicates whether a request for the hi-Z state from the corresponding counter was issued or not. 0 : No request for the hi-Z state has been issued. 1 : A request for the hi-Z state has been issued.

28.3.8 ADPACNTCFGj – Counter Setting Register j

Each of these registers makes settings of the corresponding counter. The normal recovery counter and the error determination counter are used to decide the period for eliminating noise. Rewriting this register resets the values of the internal down counter.

Access: Readable/writable in 32-bit units.

Address: <ADPAn_base> + 800_H + 4_H × j

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	SEL		—	—	—	—	—	—	—	—	—	—	—	CNMD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	NRMCNT				—	—	—	—	ERRCNT			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 28.14 ADPACNTCFG Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	Reserved	When writing, write the value after a reset.
29, 28	SEL	Count Type Selection These bits select the condition for counting by counter j. 00 : Values in excess of the upper and lower limits are counted as normal. 01 : Values in excess of the lower limit are counted as errors and values in excess of the upper limit are counted as normal. 10 : Values in excess of the upper limit are counted as errors and values in excess of the lower limit are counter as normal. 11 : Values in excess of the upper and lower limits are counted as errors.
27 to 17	Reserved	When writing, write the value after a reset.
16	CNMD	Noise Count Method Selection 0 : Error determination counter is reset once the value of counter j falls within the preset thresholds. 1 : Error determination counter is reset once the normal recovery counter value becomes 0.
15 to 12	Reserved	When writing, write the value after a reset.
11 to 8	NRMCNT	Normal Recovery Counter Setting The number of counts of the normal recovery counter is selected by these bits. After reaching the selected number of counts, the relevant output is released from the hi-Z state and this counter is reset. 0_H : Normal Recovery Counter is disabled. 1_H : The relevant output is released from the hi-Z state once the value of counter j falls below the preset thresholds. . . . F_H : The relevant output is released from the hi-Z state when the value of counter j falls below the preset thresholds fifteen times consecutively. Hi-Z control is not performed when the normal recovery counter is disabled.

Table 28.14 ADPACNTCFG Register Contents (2/2)

Bit Position	Bit Name	Function
7 to 4	Reserved	When writing, write the value after a reset.
3 to 0	ERRCNT	Error Determination Counter Setting The number of counts by the error determination counter is selected by these bits. After reaching the selected number of counts, the relevant output is placed into a hi-Z state. The counter value is reset according to the setting of the CNMD register. 0_H : Error determination counter is disabled. 1_H : The concerned output is placed into a hi-Z state once the value of the counter j exceeds the preset thresholds. . . . F_H : The concerned output is placed into a hi-Z state when the value of the counter j exceeds the preset thresholds fifteen times in total. Hi-Z control is not performed when the error determination counter is disabled.

28.3.9 ADPACNTCTRj ─ Counter Control Register j

Each of these registers makes initial settings for the corresponding counter.

Access: Writable in 32-bit units.

Address: <ADPAn_base> + 900_H + 4_H × j

Value after reset: —

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLR	
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 28.15 ADPACNTCTR Register Contents

Bit Position	Bit Name	Function
31 to 18	Reserved	The write value should be 0.
17, 16	CLR	Counter Clear Register These bits initializes counter j by reloading the configuration for the counter and cancelling the given request for the hi-Z state. 00 : Has no effect. 11 : Initializes counter j. Reloads the configuration for the counter and cancels the request for the hi-Z state which has been made. Other than above: Setting prohibited.
15 to 0	Reserved	The write value should be 0.

28.3.10 ADPADMASELq ─ DMA Resource Selection q

ADPA is configured to output twenty four DMA requests. Resources for a DMA request and the counter which issues the request are selected by this register. Selecting the counter eventually selects the conditions for counting threshold judgment results.

Access: Readable/writable in 32-bit units.

Address: <ADPAn_base> + $C00_H$ + 4_H × q

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TIMSEL		—	—	—	CNTSEL				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 28.16 ADPADMASEL Register Contents

Bit Position	Bit Name	Function
31 to 10	Reserved	When writing, write the value after a reset.
9, 8	TIMSEL	DMA Output Timing Select These bits select the timing to issue a DMA request. 00 : Issuance of DMA request is disabled. 01 : A DMA request is issued when the relevant output is released from the hi-Z state. 10 : A DMA request is issued when the relevant output is placed into a hi-Z state. 11 : A DMA request is issued when the relevant output is placed into or released from the hi-Z state.
7 to 5	Reserved	When writing, write the value after a reset.
4 to 0	CNTSEL	DMA Request Issue Counter Selection These bits select the counter which issues a DMA request. 00_H : Counter 0 01_H : Counter 1 17_H :Counter 23 18_H or above : Setting prohibited

28.3.11 ADPATPUL0 ─ Test Pulse Injection Register 0

This register generates test pulses.

Access: Writable in 32-bit units.

Address: <ADPAn_base> + $F00_H$

Value after reset: —

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TPULNRM							
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPULNRM															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 28.17 ADPATPUL0 Register Contents

Bit Position	Bit Name	Function
31 to 24	Reserved	The write value should be 0.
23 to 0	TPULNRM	Injection of Pulse for Successful Conversion Each of these bits injects a pulse notifying the corresponding counter that the result of AD conversion was within the preset threshold. For example, writing 1 to bit 0 injects a pulse to counter 0. A single writing injects one pulse. Writing 0 to these bits is ignored.

28.3.12 ADPATPUL1 ─ Test Pulse Injection Register 1

This register generates test pules.

Access: Writable in 32-bit units.

Address: <ADPAn_base> + $F04_H$

Value after reset: —

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TPULOVR							
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPULOVR															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 28.18 ADPATPUL1 Register Contents

Bit Position	Bit Name	Function
31 to 24	Reserved	The write value should be 0.
23 to 0	TPULOVR	Injection of Pulse for Excess of Upper Threshold Each of these bits injects a pulse notifying the corresponding counter that the result of AD conversion exceeded the preset upper threshold. For example, writing 1 to bit 0 injects a pulse to counter 0. A single writing injects one pulse. Writing 0 to these bits is ignored.

28.3.13 ADPATPUL2 ─ Test Pulse Injection Register 2

This register generates test pulses.

Access: Writable in 32-bit units.

Address: <ADPAn_base> + $F08_H$

Value after reset: —

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	TPULUND							
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPULUND															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 28.19 ADPATPUL2 Register Contents

Bit Position	Bit Name	Function
31 to 24	Reserved	The write value should be 0.
23 to 0	TPULUND	Injection of Pulse in Excess of Lower Threshold Each of these bits injects a pulse notifying the corresponding counter that the result of AD conversion exceeded the preset lower threshold. For example, writing 1 to bit 0 injects a pulse to counter 0. A single writing injects one pulse. Writing 0 to these bits is ignored.

28.4 Function

28.4.1 Noise Count Method

This part of the section describes how to count noises in ADPA. The number of times the result of AD conversion exceeds the given upper or lower threshold is counted and used for controlling hi-Z state of outputs from TSG3. Two counting methods are provided with different conditions for initializing the error determination counter as shown in figures below.

Noise count method when

ADPACNTCFGj.CNMD = 0,

ADPACNTCFGj.NRMCNT = 3, and

ADPACNTCFGj.ERRCNT = 3

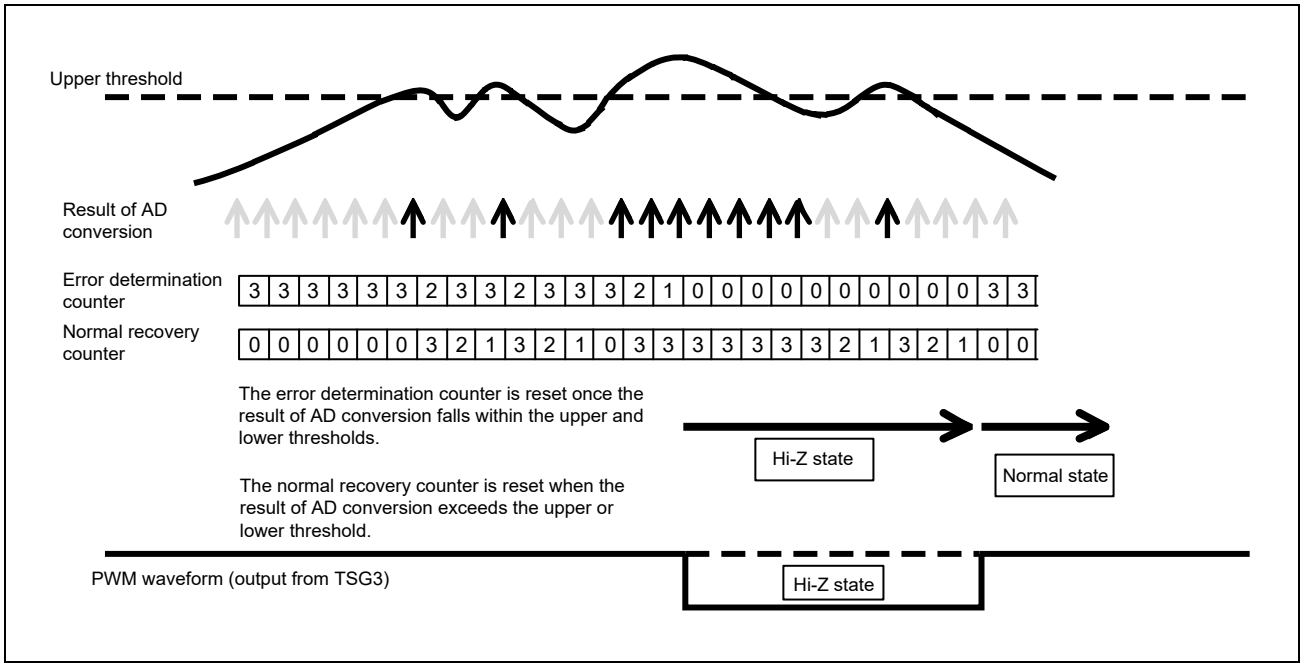


Figure 28.3 Noise Count Method 1

Noise count method when

ADPACNTCFGj.CNMD = 1,

ADPACNTCFGj.NRMCNT = 3, and

ADPACNTCFGj.ERRCNT = 3

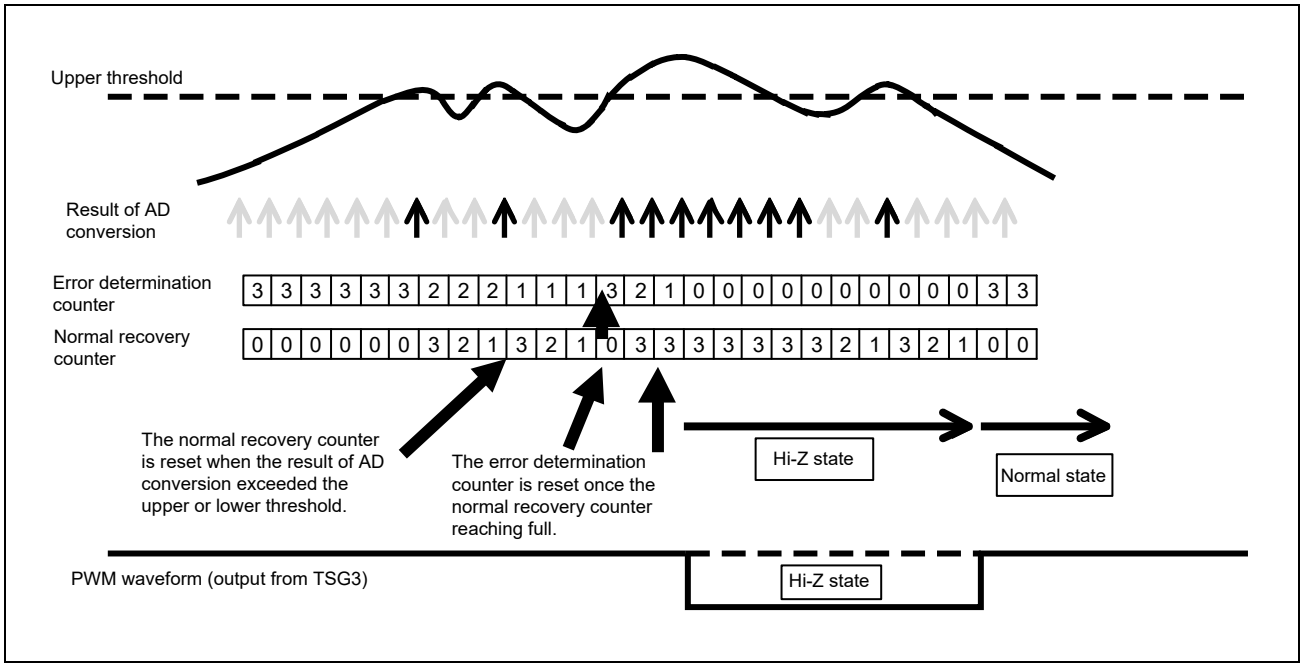


Figure 28.4 Noise Count Method 2

28.5 Operational Procedures

28.5.1 Basic Operation

(1) Configuration of ADC (see **Section 27, AD Converter (ADCC)** for details.)

Make necessary settings of the VCULME, VCLLME, and VCULLMTBS bits of the ADCC virtual channel register (ADCCkVCRj) and the virtual channel threshold table registers (ADCCkVCULLMTBR0 to 6) and perform threshold judgement.

CAUTION

The above settings are not related to the scan group upper limit/lower limit table select register.

(2) Configuration of TSG3

See **Section 20, Motor Control Timer (TSG3)** for details.

(3) Configuration of ADPA

Configure the following registers.

Counter setting registers (ADPACNTCFGj)

Counter enable registers (ADPACNTENCTRm)

Interrupt control registers (ADPAINTSTCm)

Hi-Z release mode selection registers (ADPAMDCTRm)

(4) Start AD conversion

Start AD conversion after configuring ADPA. When the settings for ADPA counter setting registers are changed, the error determination counter and normal recovery counter reload the new settings.

28.5.2 Start-Up Diagnosis

ADPA is also available in combination with self-diagnosis function provided for ADC. Details on this function are described in **Section 27, AD Converter (ADCC)**. Furthermore, by using the test pulse injection registers, users can inject desired conversion results to the target register for testing.

As a start-up diagnosis, whether hi-Z control works properly or not is checked by reading the hi-Z control state register after performing AD conversion for test by using either self-diagnosis function of ADC or test pulse injection registers.

Section 29 Functional Safety

This section describes the overview of the safety facilities of the RH850/C1M-A microcontrollers.

These microcontrollers were developed to meet the requirements for a Safety Element out of Context (SEooC) as described in the ISO26262. Contact our sales office for the details regarding the development process and safety facilities.

The following lists the failure detection functions provided by these products.

29.1 Overview

ECC and EDC

Detects failures of memories and data transfer path; correct some types of failures.

Lockstep

Detects failures of CPU1 in the early stage.

Memory Protection

Detects erroneous access to the memories and peripheral circuits to protect the data in these elements against erroneous access.

MISG

Monitors write access to a specific address by the CPU, generates the signatures based on the written data, and automatically compares the generated signatures with each other.

Clock Monitor

Monitors the clock operation to detect abnormal operation.

Duplex Configuration

For detecting failures on the failure detection circuits themselves, the functions shown in the table below have the dual hardware configuration. Failures on these functions are notified to the ECM.

Duplexed Functionality	Remark
Lockstep comparator	A lockstep comparator for CPU1
ECC decoder	
Address parity decoder	

For the ECC decoder for the peripheral RAM and data flash, the hardware including the control registers has dual configuration. When using the duplex failure detection function, set the same values to the control registers of the master and checker. For the other ECC decoders and address parity checker, control registers are not in dual configuration.

Error Control Module (ECM)

The ECM monitors various failures detected in the LSI and specifies the operation for the detected failure.

29.2 ECC and EDC

29.2.1 Overview

29.2.1.1 ECC

This product incorporates ECC for the following memories. The ECC enables detection and correction of errors of the data retained in the memories. The ECC also enables detection and correction of errors produced between the ECC encoder and memories; and memories and the ECC decoder.

Table 29.1 ECC Overview

Applicable Memory	Applicable Data Width [bit]	Operation upon Error Detection				Failure Insertion
		Detection/Correction	Notice to ECM	Error State	Address Capture	
Code flash	128	SEC-DED	Possible	Possible	Possible	Possible
Data flash	32	SEC-DED	Possible	Possible	Possible	Possible
Local RAM (CPU1, CPU2, SubCPU)	32	SEC-DED	Possible	Possible	Possible	Possible
Global RAM	32	SEC-DED	Possible	Possible	Possible	Possible
Instruction cache (data)	64	SEC-DED	Possible	Possible	Possible	Possible
Instruction cache (TAG)	16	SED-DED	Possible	Possible	Possible	Possible
RAM for DTS	32	SEC-DED	Possible	Possible	Possible	Possible
Peripheral RAM (32 bits)	32	SEC-DED	Possible	Possible	Possible	Possible

- Applicable data width

ECC encoding is applied to the data of the shown width.
If narrower data is to be written, the following processes are required. Here, ECC is also checked when data is read in (1).

(1) Read data to which ECC encoding is applied, including data to be overwritten.

(2) Replace data to be overwritten.

(3) Write back the data generated in (2).

- Detection/Correction

  SEC-DED: Detection and correction of 1-bit errors and detection of 2-bit errors are possible.

  SED-DED: Detection of 1-bit errors and 2-bit errors is possible.

- Notice to ECM

  A detected error is notified to the ECM.

- Error status

  The state of the detected error is retained.

- Address capture

  The address at which the error is detected is retained.

- Failure insertion

  Self-diagnosis of the ECC decoder is possible by intentionally generating an ECC error.

29.2.1.2 Address Parity

This product incorporates address parity for the following memories. The address parity enables detection of errors during address decoding. The address parity also enables detection of errors produced at addresses between the parity encoder and memories.

Table 29.2 Address Parity Overview

Applicable Memory	Parity Bit	Notice to ECM	Error Statues	Address Capture	Failure Insertion
Code flash	2 bits*1	Possible	Possible	Possible	Possible
Local RAM	2 bits*2	Possible	Possible	Possible	Possible
Global RAM	2 bits*2	Possible	Possible	Possible	Possible

Note 1. An address parity bit is written to one location in the memory at every 128 bits of data. Since the data are read in 256-bit units, twice the size of the data to be written, the address parity bits consist of 2 bits where one bit is an inversion of the other.

Note 2. The parity bits corresponding to the written address are written to two locations in the memory. When they are compared to the parity bits corresponding to the read address and errors are detected in both of the two parity bits stored in the memory, it is handled as an address parity error. When an error is detected only at one of the parity bits, it is handled as a parity bit error.

29.2.1.3 Data Parity

This product incorporates data parity for the particular data transfer. The data parity enables detection of errors of the transferred data. For details, see **Section 29.2.9, Data Parity for Data Transfer Path**.

29.2.2 Code Flash ECC and Address Parity

29.2.2.1 Overview

The Code Flash ECC is summarized in the table.

Table 29.3 Overview of the Code Flash ECC

Item	Description
Detection and correction of ECC errors	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. This function is enabled after a reset, that is, detection and notification of 2-bit errors and detection and correction of 1-bit errors are enabled.
Address parity	Address parity check can be either enabled or disabled. Address parity is checked during data read. This function is enabled after a reset.
Error notification	Errors are notified to the ECM on occurrence of ECC errors and parity errors. ECC error: • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. After a reset, error notification upon detection of a 2-bit ECC error is enabled and error notification upon detection of a 1-bit ECC error is disabled Parity error: • Error notification can be either enabled or disabled upon detection of an address parity error. After a reset, error notification upon detection of an address parity error is enabled. The error notification signal is issued to the ECM, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error is handled as one source.
Error states	A status register is provided, which indicates the statuses of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status is set, the applicable status is set. The error status can be cleared using the clear register.
Address capture	If an ECC error or a parity error occurs while no error status is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error, a 1-bit ECC error, or an address parity error is detected. The error status serves as the enable bit of the captured address.
Self-diagnosis	Data in the ROM and the ECC and address parity bits can all be read directly. Desired values can also be written to the ROM and to the ECC and address parity bits.
Others	The ECM can initiate a transition to the safe state in response to the detection of a 2-bit ECC error during the fetching of an instruction.

The ECC decoder and address parity generator are provided each for the read ports (CPU1, CPU2, SubCPU, and interconnect) connected to the Code Flash interface. The address parity checker is provided in the access controller for the code flash. For details, see **Figure 29.1**. The error information of the CPU1, CPU2, and SubCPU are retained in the registers, by each instruction and by each data. The instruction and data of the error information of the interconnect are retained in the same register. See **Figure 29.2**.

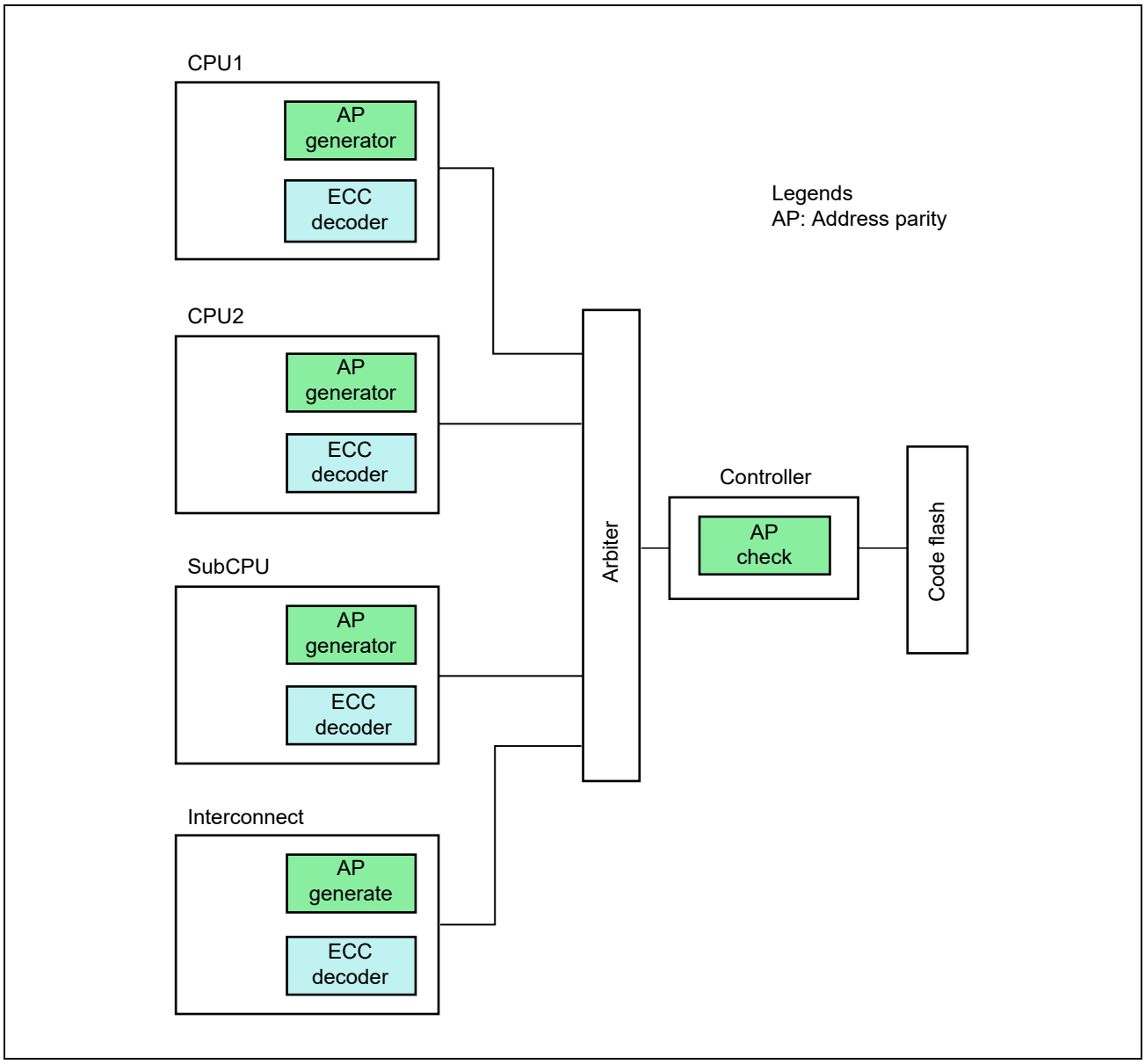


Figure 29.1 ECC and Address Parity of Code Flash

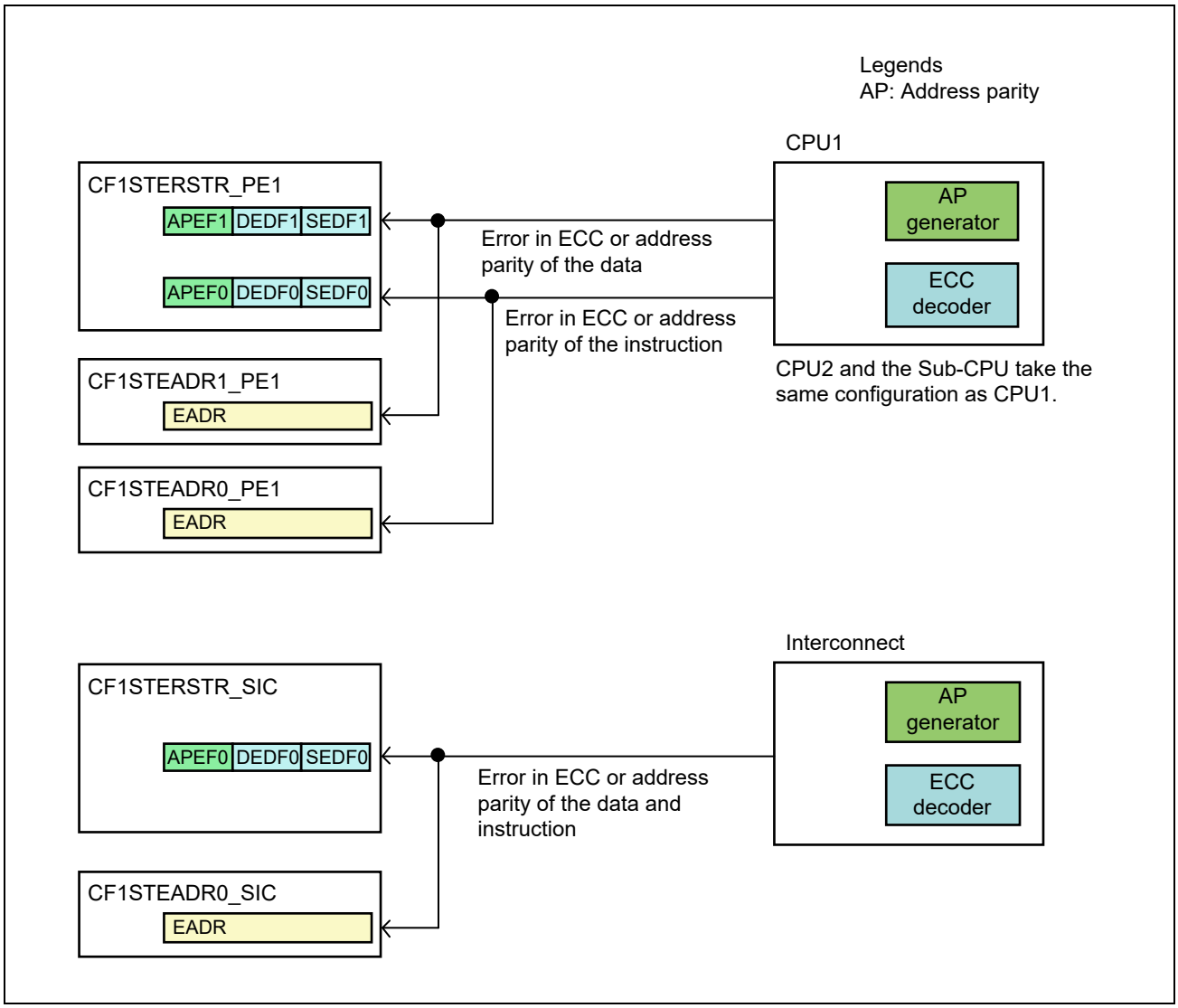


Figure 29.2 Structure of Retaining Error Information of the Code Flash

29.2.2.2 List of Registers

Table 29.4 List of Registers

Module Name	Register Name	Symbol*1	Address
ECCFLI	Code flash address parity control register	CFAPCTL	FFC6 2000H
ECCFLI	Code flash ECC control register (SIC)	CFECCCTL_SIC	FFC6 2200H
ECCFLI	Code flash error information control register (SIC)	CFERRINT_SIC	FFC6 2204H
ECCFLI	Code flash status clear register (SIC)	CFSTCLR_SIC	FFC6 2208H
ECCFLI	Code flash error count overflow status register (SIC)	CFOVFSTR_SIC	FFC6 220CH
ECCFLI	Code flash 1st error status register (SIC)	CF1STERSTR_SIC	FFC6 2210H
ECCFLI	Code flash 1st error address register (SIC)	CF1STEADR0_SIC	FFC6 2250H
ECCFLI	Code flash sub-test control register (SIC)	CFSTSTCTL_SIC	FFC6 2350H
ECCFLI	Code flash ECC control register (PE1)	CFECCCTL_PE1	FFC6 2400H
ECCFLI	Code flash error information control register (PE1)	CFERRINT_PE1	FFC6 2404H
ECCFLI	Code flash status clear register (PE1)	CFSTCLR_PE1	FFC6 2408H
ECCFLI	Code flash error count overflow status register (PE1)	CFOVFSTR_PE1	FFC6 240CH
ECCFLI	Code flash 1st error status register (PE1)	CF1STERSTR_PE1	FFC6 2410H
ECCFLI	Code flash 1st error address register 0 (PE1)	CF1STEADR0_PE1	FFC6 2450H
ECCFLI	Code flash 1st error address register 1 (PE1)	CF1STEADR1_PE1	FFC6 2454H
ECCFLI	Code flash sub-test control register (PE1)	CFSTSTCTL_PE1	FFC6 2550H
ECCFLI	Code flash ECC control register (PE2)	CFECCCTL_PE2	FFC6 2600H
ECCFLI	Code flash error information control register (PE2)	CFERRINT_PE2	FFC6 2604H
ECCFLI	Code flash status clear register (PE2)	CFSTCLR_PE2	FFC6 2608H
ECCFLI	Code flash error count overflow status register (PE2)	CFOVFSTR_PE2	FFC6 260CH
ECCFLI	Code flash 1st error status register (PE2)	CF1STERSTR_PE2	FFC6 2610H
ECCFLI	Code flash 1st error address register 0 (PE2)	CF1STEADR0_PE2	FFC6 2650H
ECCFLI	Code flash 1st error address register 1 (PE2)	CF1STEADR1_PE2	FFC6 2654H
ECCFLI	Code flash sub-test control register (PE2)	CFSTSTCTL_PE2	FFC6 2750H
ECCFLI	Code flash ECC control register (PE3)	CFECCCTL_PE3	FF75 3000H
ECCFLI	Code flash error information control register (PE3)	CFERRINT_PE3	FF75 3004H
ECCFLI	Code flash status clear register (PE3)	CFSTCLR_PE3	FF75 3008H
ECCFLI	Code flash error count overflow status register (PE3)	CFOVFSTR_PE3	FF75 300CH
ECCFLI	Code flash 1st error status register (PE3)	CF1STERSTR_PE3	FF75 3010H
ECCFLI	Code flash 1st error address register 0 (PE3)	CF1STEADR0_PE3	FF75 3050H
ECCFLI	Code flash 1st error address register 1 (PE3)	CF1STEADR1_PE3	FF75 3054H
ECCFLI	Code flash sub-test control register (PE3)	CFSTSTCTL_PE3	FF75 3150H

Note 1. The registers with symbols “_SIC”, “_PE1”, “_PE2”, and “_PE3” as suffixes are provided to the particular ECC controllers: the registers with “_SIC” are provided to the ECC controller for access from the system interconnect to the code flash, the registers with “_PE1” are provided to the ECC controller for access from CPU1, the registers with “_PE2” are provided to the ECC controller for access from CPU2, and the registers with "_PE3" are provided to the ECC controller for access from the SubCPU.

29.2.2.3 Details of Registers

(1) CFAPCTL — Code Flash Address Parity Control Register

CFAPCTL enables or disables address parity check. Set the PROT[1:0] bits to 01_B when writing to this register. This register is initialized by an internal reset or an external reset.

Access: CFAPCTL is readable/writable in 32-bit units.
CFAPCTLL is readable/writable in 16-bit units.

Address: CFAPCTL : FFC6 2000H
CFAPCTLL : FFC6 2000H :

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	APARIDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.5 CFAPCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are read as 0.
13 to 1	Reserved	The read value is 0. The write value should be 0.
0	APARIDIS	Address Parity Check Disable Enables or disables address parity check by the address parity circuit. 0: Enables address parity check. 1: Disables address parity check

(2) CFECCCTL_SIC/PE1/PE2/PE3 — Code Flash ECC Control Register

CFECCCTL enables or disables ECC error detection and correction and 1-bit error correction. Set the PROT[1:0] bits to 01_B when writing to this register. This register is initialized by an internal reset or an external reset.

Access: CFECCCTL_SIC, CFECCCTL_PE1, CFECCCTL_PE2, and CFECCCTL_PE3 are readable/writable in 32-bit units.
CFECCCTL_SICL, CFECCCTL_PE1L, CFECCCTL_PE2L, and CFECCCTL_PE3L are readable/writable in 16-bit units.

Address: CFECCCTL_SIC: FFC6 2200H, CFECCCTL_PE1 : FFC6 2400H,
CFECCCTL_PE2: FFC6 2600H, CFECCCTL_PE3 : FF75 3000H,
CFECCCTL_SICL: FFC6 2200H, CFECCCTL_PE1L : FFC6 2400H,
CFECCCTL_PE2L: FFC6 2600H, CFECCCTL_PE3L : FF75 3000H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.6 CFECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	Reserved	The read value is 0. The write value should be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(3) CFERRINT_SIC/PE1/PE2/PE3 — Code Flash Error Information Control Register

CFERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, or an address parity error.

This register is initialized by an internal reset or an external reset.

Access: CFERRINT_SIC, CFERRINT_PE1, CFERRINT_PE2, and CFERRINT_PE3 are readable/writable in 32-bit units.
CFERRINT_SICL, CFERRINT_PE1L, CFERRINT_PE2L, and CFERRINT_PE3L are readable/writable in 16-bit units.
CFERRINT_SICLL, CFERRINT_PE1LL, CFERRINT_PE2LL, and CFERRINT_PE3LL are readable/writable in 8-bit units.

Address: CFERRINT_SIC : FFC6 2204H, CFERRINT_PE1 : FFC6 2404H,
CFERRINT_PE2 : FFC6 2604H, CFERRINT_PE3 : FF75 3004H,
CFERRINT_SICL : FFC6 2204H, CFERRINT_PE1L : FFC6 2404H,
CFERRINT_PE2L : FFC6 2604H, CFERRINT_PE3L : FF75 3004H,
CFERRINT_SICLL : FFC6 2204H, CFERRINT_PE1LL : FFC6 2404H,
CFERRINT_PE2LL : FFC6 2604H, CFERRINT_PE3LL : FF75 3004H

Value after reset: 0000 0006H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	APEIE	DEDIE	SEDIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 29.7 CFERRINT Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	The read value is 0. The write value should be 0.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(4) CFSTCLR_SIC/PE1/PE2/PE3 — Code Flash Status Clear Register

CFSTCLR clears the error flags in the error status register (CF1STERSTR), the overflow flag in the error count overflow status register (CFOVFSTR), and the error address register (CF1STEADR). CFSTCLR is a write-only register and is always read as 0.

Access: CFSTCLR_SIC, CFSTCLR_PE1, CFSTCLR_PE2, and CFSTCLR_PE3 are writable in 32-bit units.
CFSTCLR_SICL, CFSTCLR_PE1L, CFSTCLR_PE2L, and CFSTCLR_PE3L are writable in 16-bit units.
CFSTCLR_SICLL, CFSTCLR_PE1LL, CFSTCLR_PE2LL, and CFSTCLR_PE3LL are writable in 8-bit units.

Address: CFSTCLR_SIC : FFC6 2208H, CFSTCLR_PE1 : FFC6 2408H,
CFSTCLR_PE2 : FFC6 2608H, CFSTCLR_PE3 : FF75 3008H,
CFSTCLR_SICL : FFC6 2208H, CFSTCLR_PE1L : FFC6 2408H,
CFSTCLR_PE2L : FFC6 2608H, CFSTCLR_PE3L : FF75 3004H,
CFSTCLR_SICLL : FFC6 2208H, CFSTCLR_PE1LL : FFC6 2408H,
CFSTCLR_PE2LL : FFC6 2608H, CFSTCLR_PE3LL : FF75 3008H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR1	STCLR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 29.8 CFSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	STCLR1	Error Overflow Flag Clear for Data Writing 1 to this bit clears the APEF1, DEDF1, and SEDF1 flags in CF1STERSTR; ERROVF1 flag in CFOVFSTR; and CF1STEADR1.
0	STCLR0	Error Overflow Flag Clear for Instruction and SIC Writing 1 to this bit clears the APEF0, DEDF0, and SEDF0 flags in CF1STERSTR; ERROVF0 flag in CFOVFSTR; and CF1STEADR0.

(5) CFOVFSTR_SIC/PE1/PE2/PE3 — Code Flash Error Count Overflow Status Register

CFOVFSTR indicates overflows of the error counter. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the CFSTCLR.STCLR bit to 1.

Access: CFOVFSTR_SIC, CFOVFSTR_PE1, CFOVFSTR_PE2, and CFOVFSTR_PE3 are readable in 32-bit units.
CFOVFSTR_SICL, CFOVFSTR_PE1L, CFOVFSTR_PE2L, and CFOVFSTR_PE3L are readable in 16-bit units.
CFOVFSTR_SICLL, CFOVFSTR_PE1LL, CFOVFSTR_PE2LL, and CFOVFSTR_PE3LL are readable in 8-bit units.

Address: CFOVFSTR_SIC : FFC6 220C$_H$, CFOVFSTR_PE1 : FFC6 240C$_H$,
CFOVFSTR_PE2 : FFC6 260C$_H$, CFOVFSTR_PE3 : FF75 300C$_H$,
CFOVFSTR_SICL : FFC6 220C$_H$, CFOVFSTR_PE1L : FFC6 240C$_H$,
CFOVFSTR_PE2L : FFC6 260C$_H$, CFOVFSTR_PE3L : FF75 300C$_H$,
CFOVFSTR_SICLL : FFC6 220C$_H$, CFOVFSTR_PE1LL : FFC6 240C$_H$,
CFOVFSTR_PE2LL : FFC6 260C$_H$, CFOVFSTR_PE3LL : FF75 300C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF1	ERROVF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.9 CFOVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	ERROVF1	Error Overflow Flag for Data This flag is set if the second error occurs while any of the error flags (APEF1, DEDF1, and SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag for Instruction and SIC This flag is set if the second error occurs while any of the error flags (APEF0, DEDF0, and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(6) CF1STERSTR_SIC/PE1/PE2/PE3 — Code Flash 1st Error Status Register

CF1STERSTR monitors occurrence of the first error. The error status is set if an error occurs while the error flag is 0. The error flag is overwritten if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error monitor flag is set.

If multiple errors occur simultaneously, all the corresponding error flags are set. CF1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit in CFSTCLR to 1.

Access: CF1STERSTR_SIC, CF1STERSTR_PE1, CF1STERSTR_PE2, and CF1STERSTR_PE3 are readable in 32-bit units.
CF1STERSTR_SICL, CF1STERSTR_PE1L, CF1STERSTR_PE2L, and CF1STERSTR_PE3L are readable in 16-bit units.
CF1STERSTR_SICLL, CF1STERSTR_SICLH, CF1STERSTR_PE1LL, CF1STERSTR_PE1LH, CF1STERSTR_PE2LL, CF1STERSTR_PE2LH, CF1STERSTR_PE3LL, and CF1STERSTR_PE3LH are readable in 8-bit units.

Address: CF1STERSTR_SIC : FFC6 2210$_H$, CF1STERSTR_PE1 : FFC6 2410$_H$,
CF1STERSTR_PE2 : FFC6 2610$_H$, CF1STERSTR_PE3 : FF75 3010$_H$,
CF1STERSTR_(SIC/PE1/PE2/PE3)L : CF1STERSTR_(SIC/PE1/PE2/PE3) + 00$_H$,
CF1STERSTR_(SIC/PE1/PE2/PE3)LL : CF1STERSTR_(SIC/PE1/PE2/PE3) + 00$_H$,
CF1STERSTR_(SIC/PE1/PE2/PE3)LH : CF1STERSTR_(SIC/PE1/PE2/PE3) + 01$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	APEF1 *1	DEDF1 *1	SEDF1 *1	—	—	—	—	—	APEF0	DEDF0	SEDF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. These bits are reserved in the CF1STERSTR_SIC register. They are always read as 0. The write value should always be 0.

Table 29.10 CF1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 11, 7 to 3	Reserved	The read value is 0. The write value should be 0.
2 + 8n	APEFn	Address Parity Error Monitor Flag Clearing conditions: A reset is generated or the CFSTCLR.STCLRn bit is set to 1. Setting conditions: An address parity error has occurred while the error flags DEDFn and APEFn are 0.
1 + 8n	DEDFn	ECC 2-bit Error Monitor Flag Clearing conditions: A reset is generated or the CFSTCLR.STCLRn bit is set to 1. Setting conditions: A 2-bit error has occurred while the error flags DEDFn and APEFn are 0.
0 + 8n	SEDFn	ECC 1-bit Error Monitor Flag Clearing conditions: A reset is generated or the CFSTCLR. STCLRn bit is set to 1. Setting conditions: A 1-bit error has occurred while the error flags DEDFn, SEDFn, and APEFn are 0.

Note: SIC: n = 0
PE1/PE2/PE3: n = 0 (for instructions), n = 1 (for data)

(7) CF1STEADRn_SIC/PE1/PE2/PE3 — Code flash 1st error address register n
SIC: n = 0, PE1/PE2/PE3: n = 0, 1

CF1STEADR holds the address at which an error has occurred.

The error address is set if an error occurs while all the error flags are 0 in CF1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

EADR[27:5] of this register correspond to bits [27:5] of the real address. The real address can be calculated by appending the higher-order address bits [31:28] as a base address.

CF1STEADR0 is cleared by an internal reset, an external reset, or setting the STCLR bit in CFSTCLR to 1.

Access: CF1STEADR0_SIC, CF1STEADRn_PE1, CF1STEADRn_PE2, and CF1STEADRn_PE3 are readable in 32-bit units.
CF1STEADRn_(SIC/PE1/PE2/PE3)_(L/H) is readable in 16-bit units.
CF1STEADRn_(SIC/PE1/PE2/PE3)_(LL/LH/HL/HH) is readable in 8-bit units.

Address: CF1STEADR0_SIC : FFC6 2250$_H$,
CF1STEADR0_PE1 : FFC6 2450$_H$, CF1STEADR1_PE1 : FFC6 2454$_H$,
CF1STEADR0_PE2 : FFC6 2650$_H$, CF1STEADR1_PE2 : FFC6 2654$_H$,
CF1STEADR0_PE3 : FF75 3050$_H$, CF1STEADR1_PE3 : FF75 3054$_H$,
CF1STEADR0_(SIC/PE1/PE2/PE3)_L : CF1STEADR0_(SIC/PE1/PE2/PE3) + 00$_H$,
CF1STEADR0_(SIC/PE1/PE2/PE3)_H : CF1STEADR0_(SIC/PE1/PE2/PE3) + 02$_H$,
CF1STEADR0_(SIC/PE1/PE2/PE3)_LL : CF1STEADR0_(SIC/PE1/PE2/PE3) + 00$_H$,
CF1STEADR0_(SIC/PE1/PE2/PE3)_LH : CF1STEADR0_(SIC/PE1/PE2/PE3) + 01$_H$,
CF1STEADR0_(SIC/PE1/PE2/PE3)_HL : CF1STEADR0_(SIC/PE1/PE2/PE3) + 02$_H$,
CF1STEADR0_(SIC/PE1/PE2/PE3)_HH : CF1STEADR0_(SIC/PE1/PE2/PE3) + 03$_H$,
CF1STEADR1_(SIC/PE1/PE2/PE3)_L : CF1STEADR1_(SIC/PE1/PE2/PE3) + 00$_H$,
CF1STEADR1_(SIC/PE1/PE2/PE3)_H : CF1STEADR1_(SIC/PE1/PE2/PE3) + 02$_H$,
CF1STEADR1_(SIC/PE1/PE2/PE3)_LL : CF1STEADR1_(SIC/PE1/PE2/PE3) + 00$_H$,
CF1STEADR1_(SIC/PE1/PE2/PE3)_LH : CF1STEADR1_(SIC/PE1/PE2/PE3) + 01$_H$,
CF1STEADR1_(SIC/PE1/PE2/PE3)_HL : CF1STEADR1_(SIC/PE1/PE2/PE3) + 02$_H$,
CF1STEADR1_(SIC/PE1/PE2/PE3)_HH : CF1STEADR1_(SIC/PE1/PE2/PE3) + 03$_H$

Value after reset: 0000 0000$_H$

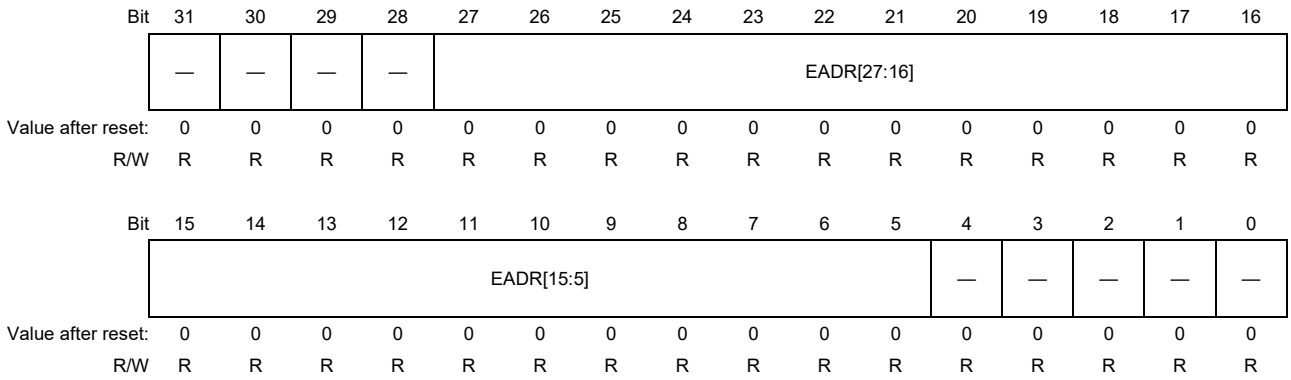


Table 29.11 CF1STEADRn Register Contents

Bit Position	Bit Name	Function
31 to 28	Reserved	The read value is 0. The write value should be 0.
27 to 5	EADR[27:5]	1st Error Address These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in CF1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.
4 to 0	Reserved	The read value is 0. The write value should be 0.

Note: SIC: n = 0
PE1/PE2/PE3: n = 0 (for instruction), n = 1 (for data)

(8) CFSTSTCTL_SIC/PE1/PE2/PE3 — Code flash sub-test control register

CFSTSTCTL is used for the ECC test (self-diagnosis). This register is dedicated for the code flash. After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly. Set the PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Access: CFSTSTCTL_SIC, CFSTSTCTL_PE1, CFSTSTCTL_PE2, and CFSTSTCTL_PE3 are readable/writable in 32-bit units. CFSTSTCTL_SICL, CFSTSTCTL_PE1L, CFSTSTCTL_PE2L, and CFSTSTCTL_PE3L are readable/writable in 16-bit units.

Address: CFSTSTCTL_SIC : FFC6 2350$_H$, CFSTSTCTL_PE1 : FFC6 2550$_H$, CFSTSTCTL_PE2 : FFC6 2750$_H$, CFSTSTCTL_PE3 : FF75 3150$_H$, CFSTSTCTL_SICL : FFC6 2350$_H$, CFSTSTCTL_PE1L : FFC6 2550$_H$, CFSTSTCTL_PE2L : FFC6 2750$_H$, CFSTSTCTL_PE3L : FF75 3150$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	ECCTST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.12 CFSTSTCTL_SIC/PE1/PE2/PE3 Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 1	Reserved	The read value is 0. The write value should be 0.
0	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly.

Correctly reading instructions from the code flash access path is not possible while ECC test mode is selected (ECCTST = 1). While the access path for the CPU is set to test mode (including during changes to the value of the ECCTST bit), the CPU must run a program from the local RAM or global RAM and must not fetch instructions from the code flash memory.

The CPU has a small data buffer. If an old value remains in this buffer, the correct value cannot be read even when the ECCTST bit is switched. When switching the ECCTST bit, be sure to clear the data buffer. For how to clear the data buffer, see **Section 3, CPU System**.From the code flash access path with ECC test mode selected, access must be made by reading 4 bytes aligned to 16n address. The results of code flash reading are as follows:

Table 29.13 Results of Code Flags Reading

Bit Position	Contents
bit[31:10]	Always 0
bit[9]	Address parity bit
bit[8:0]	ECC bits

29.2.2.4 Test Function

Through appropriate register setting, the code flash, ECC bits, and address parity bits can be read.

(1) Reading code flash data

1. Set the ECCDIS bit in the CFECCCTL register to 1 to disable ECC error detection and correction.
2. When ECCDIS = 1, neither error detection nor correction proceeds when the code flash is read; the data output from the code flash is read unchanged.

How to exit this test mode:

3. Set the ECCDIS bit in the CFECCCTL register to 0 to enable ECC error detection and correction

(2) Reading the ECC bits and address parity bits

1. Set the ECCDIS bit in the CFECCCTL register to 1 to disable ECC error detection and correction.
2. Set the ECCTST bit in the CFSTSTCTL register to 1 to set test mode.
3. When the code flash is read, the ECC and address parity bits are read instead of the code flash data.

How to exit this test mode:

4. Set the ECCDIS bit in the CFECCCTL register to 0 to enable ECC error detection and correction.
5. Set the ECCTST bit in the CFSTSTCTL register to 0 to set normal mode.

(3) Self-diagnosis

Self-diagnosis of the ECC decoder and address parity decoder for the access paths is possible by writing incorrect data to the code flash memory beforehand (fault injection) and then reading this data. In writing of the address parity, since an address-parity bit is generated for each 128-bit unit, two address-parity bits correspond to the 256-bit unit of reading, so always write inverse values to the two bits even when injecting an error.

For details on programming of the code flash, refer to *RH850/C1M-A Flash Memory User's Manual: Hardware Interface*.

29.2.3 Data Flash ECC

29.2.3.1 Overview

The data flash ECC is summarized in the table below.

Table 29.14 Overview of Data Flash ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. This function is enabled after a reset, that is, detection and notification of 2-bit errors and detection and correction of 1-bit errors are enabled.
Error notification	Upon occurrence of an ECC error, the ECM is notified of the error. ECC Error: • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. In the state after reset, error notification is enabled upon detection of a 2-bit ECC error, and error notification is disabled upon detection of a 1-bit ECC error. The error notification signal is output, where a 2-bit ECC error is handled as one source, and a 1-bit ECC error is handled as one source. ECC errors of data flash are included in the system error exemption (SYSERR exception). For details, see **Section 3, CPU System**.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection and 1-bit ECC error detection. If an error occurs while no error status flag is set, the corresponding status is set. The error status can be cleared by using the clear register.
Address capture	If an ECC error occurs while no error status flag is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error or a 1-bit ECC error is detected. The error status serves as the enable bit of the capture address.
Self-diagnosis	Data in the ROM and the ECC bits can be read directly. Desired values can be written as ROM data and to the ECC bits.

29.2.3.2 List of Registers

(1) List of ECC modules

Table 29.15 List of ECC Modules

ECC Module Name and Register Base Address			
Master*1		Checker*1	
Module Name	Base Address <Base_addr>	Module Name	Base Address
ECCEEP	FFC6 2C00$_H$	ECCEEPC	FFC6 2E00$_H$

Note 1. Two modules (one for master and the other for checker) are provided for duplex configuration.

(2) List of registers

The following registers are provided for the individual ECC modules.

Table 29.16 List of Registers

Module Name	Register Name	Symbol	Address
ECCEEP	Data Flash ECC control register	DFECCCTL	<Base_addr>
ECCEEP	Data flash error status register	DFERSTR	<Base_addr> + 04$_H$
ECCEEP	Data flash error status clear register	DFERSTC	<Base_addr> + 08$_H$
ECCEEP	Data flash error overflow status register	DFOVFSTR	<Base_addr> + 0C$_H$
ECCEEP	Data flash error overflow status clear register	DFOVFSTC	<Base_addr> + 10$_H$
ECCEEP	Data flash error notification control register	DFERRINT	<Base_addr> + 14$_H$
ECCEEP	Data flash 1st error address register	DFEADR	<Base_addr> + 18$_H$
ECCEEP	Data flash test control register	DFTSTCTL	<Base_addr> + 1C$_H$

29.2.3.3 Details of Registers

(1) DFECCCTL — Data flash ECC control register

DFECCCTL enables or disables ECC error detection and correction and 1-bit error correction. DFECCCTL is initialized by an internal reset or an external reset.

Set the PROT[1:0] bits to 01_B when writing to DFECCCTL.

Access: DFECCCTL register is readable/writable in 16-bit units.

Address: DFECCCTL : <Base_addr>

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.17 DFECCCTL Register Contents

Bit Position	Bit Name	Function
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	Reserved	The read value is 0. The write value should be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. In the initial state, ECC error detection and correction are enabled. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(2) DFERSTR — Data flash error status register

DFERSTR monitors occurrence of errors.

DFERSTR is initialized by an internal reset, an external reset, or setting the clear bit in the data flash error status clear register.

The SEDF bit is set if a 1-bit ECC error is detected while ECC error detection and correction are enabled, and the DEDF bit is set if a 2-bit ECC error is detected.

Access: DFERSTR register is readable in 8-bit units.

Address: DFERSTR : <Base_addr> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEDF	SEDF
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 29.18 DFERSTR Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	The read value is 0. The write value should be 0.
1	DEDF	2-Bit ECC Error Monitor This bit is set if a 2-bit ECC error is generated while SEDF and DEDF are 0. Clearing conditions: A reset is generated or the ERRCLR bit in the DFERSTC register is set. Setting condition: A 2-bit ECC error is generated with both SEDF and DEDF being 0.
0	SEDF	1-Bit ECC Error Monitor This bit is set if a 1-bit ECC error is generated while SEDF and DEDF are 0. Clearing conditions: A reset is generated or the ERRCLR bit is set in data flash error status clear register. Setting condition: A 1-bit ECC error is generated with both SEDF and DEDF being 0.

(3) DFERSTC — Data flash error status clear register

DFERSTC clears the error flags in the data flash error status register. DFERSTC is a write-only register and is always read as 0.

Access: DFERSTC register is writable in 8-bit units.

Address: DFERSTC : <Base_addr> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERRCLR
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 29.19 DFERSTC Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	The read value is 0. The write value should be 0.
0	ERRCLR	SEDF/DEDF Flag Clear Writing 1 to this bit clears the SEDF/DEDF flag.

(4) DFOVFSTR — Data flash error overflow status register

DFOVFSTR monitors occurrence of data flash error overflow. The ERROVF flag is cleared by an internal reset, an external reset, or setting the ERROVFCLR bit to 1 in DFOVFSTC.

Access: DFOVFSTR register is readable in 8-bit units.

Address: DFOVFSTR : <Base_addr> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERROVF
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 29.20 DFOVFSTR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	The read value is 0. The write value should be 0.
0	ERROVF	Error Overflow Flag This flag is set if an ECC error occurs while the error address register is full.

(5) DFOVFSTC — Data flash error overflow status clear register

DFOVFSTC clears the data flash error overflow flag. Setting the ERROVFCLR bit to 1 clears this flag.

Access: DFOVFSTC register is writable in 8-bit units.

Address: DFOVFSTC : <Base_addr> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERROVFCLR
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 29.21 DFOVFSTC Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	The read value is 0. The write value should be 0.
0	ERROVFCLR	Error Overflow Flag Clear Writing 1 to this bit clears the ERROVF flag. This bit is always read as 0.

(6) DFERRINT — Data flash error notification control register

DFERRINT enables or disables generation of the error notification signal upon detection of a 2-bit ECC error or a 1-bit ECC error.

Access: DFERRINT register is readable/writable in 8-bit units.

Address: DFERRINT : <Base_addr> + 14_H

Value after reset: 02_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset:	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 29.22 DFERRINT Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	The read value is 0. The write value should be 0.
1	DEDIE	2-Bit ECC Error Notification Control Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Control Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(7) DFEADR — Data flash 1st error address register

DFEADR holds the address at which an ECC error has occurred while both of the SEDF and DEDF bits in the data flash error status register are 0.

Access: DFEADR register is readable in 32-bit units.

Address: DFEADR : <Base_addr> + 18_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	DFEADR[20:16]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DFEADR[15:2]														—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.23 DFEADR Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	The read value is 0. The write value should be 0.
20 to 2	DFEADR[20:2]	ECC Error Address These bits are read-only and used to monitor the address at which an ECC error has occurred. Since this register holds the internal address, add the base address ($FF20\ 0000_H$) of the data flash memory to transform the internal address to the real address.
1, 0	Reserved	The read value is 0. The write value should be 0.

(8) DFTSTCTL — Data flash test control register

DFTSTCTL is used for the ECC test. After ECC test mode is enabled by setting ECCTST = 1, the ECC bits can be read.

Set the PROT[1:0] bits to 01B when writing to DFTSTCTL.

Access: DFTSTCTL register is readable/writable in 16-bit units.

Address: DFTSTCTL : <Base_addr> + $1C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	ECCTST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.24 DFTSTCTL Register Contents

Bit Position	Bit Name	Function
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 1	Reserved	When read, 0 is always returned. The write value should be 0.
0	ECCTST	ECC Test Sets ECC test mode.

29.2.3.4 Test Function

Data in the ROM and the ECC bits can be read through the setting of the data flash test control register (DFTSTCTL).

(1) Reading the ROM data

1. Set the ECCDIS bit in the data flash ECC control register to 1 to disable ECC error detection and correction.
2. When ECCDIS = 1, neither error detection nor correction proceeds when the data flash is read; the data output from the data flash is read unchanged.

How to exit this test mode:

3. Set the ECCDIS bit in the data flash ECC control register to 0 to enable ECC error detection and correction.

(2) Reading the ECC data

1. Set the ECCDIS bit in the DFECCCTL register to 1 to disable ECC error detection and correction.
2. Set the ECCTST bit in the DFTSTCTL register to 1 to set test mode.
3. When the data flash is read, the 7 lower-order bits of read data are read as ECC data.

How to exit this test mode:

4. Set the ECCDIS bit in the DFECCCTL register to 0 to enable ECC error detection and correction.
5. Set the ECCTST bit in the DFTSTCTL register to 0 to set normal mode.

(3) Self-diagnosis

Self-diagnosis of the ECC decoder is possible by writing incorrect data to the data flash memory beforehand (fault injection) and then reading this data. A 1- or 2-bit ECC error fault can be injected by generating correct ECC bits once and inverting only the appropriate bits.

For details on programming of the data flash, refer to *RH850/C1M-A Flash Memory User's Manual: Hardware Interface*.

29.2.4 Local RAM (CPU1, CPU2, SubCPU) ECC and Address Parity

29.2.4.1 Overview

The local RAM ECC of CPU1, CPU2, and SubCPU are summarized in the table below.

Table 29.25 Overview of the Local RAM CPU1, CPU2, and SubCPU

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. This function is enabled after a reset, that is, detection of 2-bit errors and detection and correction of 1-bit errors are enabled.
Address parity	Address parity check can be either enabled or disabled. During data write, a parity bit generated based on the written address is written simultaneously with the write data. At this time, the same parity bit is written to two locations in the RAM. During data read, comparison is performed between a parity bit generated based on the read address and a parity bit read from the RAM. The error decoding specifications are shown in **Table 29.26, Address Parity Definitions**. This function is enabled after a reset.
Error notification	Indicators of the occurrence of ECC and address parity errors are conveyed to ECM. ECC Error: • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. The register values after a reset enable notification of the 2-bit error and disable notification of the 1-bit error. Parity Error: • Error notification can be either enabled or disabled upon detection of an address parity error. • Error notification can be either enabled or disabled upon detection of a parity bit error. The register values after a reset enable notification of the address parity error and disable notification of the parity bit error. The error notification signal is output, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error and a parity bit error are handled as one source.
Error status	A status register is provided, which indicates the states of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status is set, the corresponding status is set. The error status can be cleared using the clear register.
Address capture	If an ECC error or a parity error occurs while no error status is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error, a 1-bit ECC error, or an address parity error is detected. The error status serves as the enable bit of the capture address.
Self-diagnosis	Desired values can be written as RAM data and to the ECC and address parity bits. Data in the RAM and the ECC and address parity bits can be read directly.

The definitions of the address parity error during read accesses are shown in the table below.

Table 29.26 Address Parity Definitions

RAM Macro Address Parity Bit 1	RAM Macro Address Parity Bit 2	Read Address Parity	Error Determination and Error Name
0	0	0	No error
0	0	1	Address parity error
0	1	0	Parity bit error
0	1	1	Parity bit error
1	0	0	Parity bit error
1	0	1	Parity bit error
1	1	0	Address parity error
1	1	1	No error

The local RAM of CPU1, CPU2, and the SubCPU has structures that can read and write a maximum of 128 bits of data simultaneously. In the meanwhile, ECC or address parity is provided for each 32-bit data, and the 32-bit data is divided into four words (word 0 to word 3). The relationship between addresses and words is described below.

Table 29.27 Relationship between Addresses and Words

Four Lower-Order Bits of Address (Hexadecimal Notation)	F_H to C_H	B_H to 8_H	7_H to 4_H	3_H to 0_H
Word number	Word 3	Word 2	Word 1	Word 0

The error information of the CPU1, CPU2, and SubCPU are retained by each instruction and by each data.

29.2.4.2 List of Registers

Table 29.28 List of Registers

Module Name	Register Name	Symbol*1	Address
ECCCPU1	Local RAM address parity control register (PE1)	LRAPCTL_PE1	FFC6 5000H
ECCCPU1	Local RAM test control register (PE1)	LRTSTCTL_PE1	FFC6 5004H
ECCCPU1	Local RAM test data read buffer 0 (PE1)	LRTDATBF0_PE1	FFC6 5008H
ECCCPU1	Local RAM test data read buffer 1 (PE1)	LRTDATBF1_PE1	FFC6 500CH
ECCCPU2	Local RAM address parity control register (PE2)	LRAPCTL_PE2	FFC6 5020H
ECCCPU2	Local RAM test control register (PE2)	LRTSTCTL_PE2	FFC6 5024H
ECCCPU2	Local RAM test data read buffer 0 (PE2)	LRTDATBF0_PE2	FFC6 5028H
ECCCPU2	Local RAM test data read buffer 1 (PE2)	LRTDATBF1_PE2	FFC6 502CH
ECCCPU1	Local RAM ECC control register (PE1)	LRECCCTL_PE1	FFC6 5400H
ECCCPU1	Local RAM error information control register (PE1)	LRERRINT_PE1	FFC6 5404H
ECCCPU1	Local RAM status clear register (PE1)	LRSTCLR_PE1	FFC6 5408H
ECCCPU1	Local RAM error count overflow status register (PE1)	LROVFSTR_PE1	FFC6 540CH
ECCCPU1	Local RAM 1st error status register (PE1)	LR1STERSTR_PE1	FFC6 5410H
ECCCPU1	Local RAM 1st error address register 0(PE1)	LR1STEADR0_PE1	FFC6 5450H
ECCCPU1	Local RAM 1st error address register 1(PE1)	LR1STEADR1_PE1	FFC6 5454H
ECCCPU2	Local RAM ECC control register (PE2)	LRECCCTL_PE2	FFC6 5600H
ECCCPU2	Local RAM error information control register (PE2)	LRERRINT_PE2	FFC6 5604H
ECCCPU2	Local RAM status clear register (PE2)	LRSTCLR_PE2	FFC6 5608H
ECCCPU2	Local RAM error count overflow status register (PE2)	LROVFSTR_PE2	FFC6 560CH
ECCCPU2	Local RAM 1st error status register (PE2)	LR1STERSTR_PE2	FFC6 5610H
ECCCPU2	Local RAM 1st error address register 0 (PE2)	LR1STEADR0_PE2	FFC6 5650H
ECCCPU2	Local RAM 1st error address register 1 (PE2)	LR1STEADR1_PE2	FFC6 5654H
ECCCPU3	Local RAM address parity control register (PE3)	LRAPCTL_PE3	FF75 4000H
ECCCPU3	Local RAM test control register (PE3)	LRTSTCTL_PE3	FF75 4004H
ECCCPU3	Local RAM test data read buffer 0 (PE3)	LRTDATBF0_PE3	FF75 4008H
ECCCPU3	Local RAM test data read buffer 1(PE3)	LRTDATBF1_PE3	FF75 400CH
ECCCPU3	Local RAM ECC control register (PE3)	LRECCCTL_PE3	FF75 5000H
ECCCPU3	Local RAM error information control register (PE3)	LRERRINT_PE3	FF75 5004H
ECCCPU3	Local RAM status clear register (PE3)	LRSTCLR_PE3	FF75 5008H
ECCCPU3	Local RAM error count overflow status register (PE3)	LROVFSTR_PE3	FF75 500CH
ECCCPU3	Local RAM 1st error status register (PE3)	LR1STERSTR_PE3	FF75 5010H
ECCCPU3	Local RAM 1st error address register 0 (PE3)	LR1STEADR0_PE3	FF75 5050H
ECCCPU3	Local RAM 1st error address register 1(PE3)	LR1STEADR1_PE3	FF75 5054H

Note 1. The registers with "_PE1" are provided for access to the local RAM of CPU1, the registers with "_PE2" are provided for access to the local RAM of CPU2, and the registers with "_PE3" are provided for access to the local RAM of the SubCPU.

29.2.4.3 Details of Registers

(1) LRAPCTL_PE1/PE2/PE3 — Local RAM address parity control register

LRAPCTL enables or disables address parity check. Set the PROT[1:0] bits to 01_B when writing to LRAPCTL.

LRAPCTL is initialized by an internal reset or an external reset.

Access: LRAPCTL_PE1, LRAPCTL_PE2, LRAPCTL_PE3 are readable/writable in 32-bit units.
LRAPCTL_PE1L, LRAPCTL_PE2L, LRAPCTL_PE3L are readable/writable in 16-bit units.

Address: LRAPCTL_PE1 : FFC6 5000H , LRAPCTL_PE2 : FFC6 5020H , LRAPCTL_PE3 : FF75 4000H ,
LRAPCTL_PE1L : FFC6 5000H , LRAPCTL_PE2L : FFC6 5020H , LRAPCTL_PE3L : FF75 4000H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	APARIDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.29 LRAPCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 1	Reserved	The read value is 0. The write value should be 0.
0	APARIDIS	Address Parity Check Disable Enables or disables address parity check by all the address parity circuits (word 0 to word 3). 0: Enables address parity check. 1: Disables address parity check.

(2) LRTSTCTL_PE1/PE2/PE3 — Local RAM test control register

This register is used for the ECC test (self-diagnosis) and address parity checker test (self-diagnosis). After ECC test mode is enabled by setting ECCTST = 1, desired values can be written to the ECC and address parity bits. The DATSEL bit is used to select RAM data or the ECC and address parity bits.

By setting address parity test mode (APTESTi = 1; i = 0, 1, 2, 3), the parity to be input to the address parity checker is inverted. Set the PROT[1:0] bits to 01B when writing to LRTSTCTL.

LRTSTCTL is initialized by an internal reset or an external reset.

Access: LRTSTCTL_PE1, LRTSTCTL_PE2, and LRTSTCTL_PE3 are readable/writable in 32-bit units.
LRTSTCTL_PE1L, LRTSTCTL_PE2L, and LRTSTCTL_PE3L are readable/writable in 16-bit units.

Address: LRTSTCTL_PE1 : FFC6 5004H, LRTSTCTL_PE2 : FFC6 5024H, LRTSTCTL_PE3 : FF75 4004H,
LRTSTCTL_PE1L : FFC6 5004H, LRTSTCTL_PE2L : FFC6 5024H, LRTSTCTL_PE3L : FF75 4004H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	APTEST3	APTEST2	APTEST1	APTEST0	ECCTST	DATSEL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.30 LRTSTCTL_PE1/PE2 Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, 0 is always returned. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 6	Reserved	When read, 0 is always returned. The write value should be 0.
5	APTEST3	Address Parity Checker (Word 3) Test Sets the address parity checker to test mode. When APTEST3 = 1, the parity generated through the address parity generator is inverted.
4	APTEST2	Address Parity Checker (Word 2) Test Sets the address parity checker to test mode. When APTEST2 = 1, the parity generated through the address parity generator is inverted.
3	APTEST1	Address Parity Checker (Word 1) Test Sets the address parity checker to test mode. When APTEST1 = 1, the parity generated through the address parity generator is inverted.
2	APTEST0	Address Parity Checker (Word 0) Test Sets the address parity checker to test mode. When APTEST0 = 1, the parity generated through the address parity generator is inverted.
1	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly.
0	DATSEL	Data Select This bit is valid when ECCTST = 1. This bit selects the RAM bit which can be accessed when writing. 0: RAM data is selected. 1: The ECC bits and address parity bit are selected.

CAUTION

When ECC test mode for the local RAM is enabled (ECCTST = 1), access to the local RAM should be in 4-byte units.

(3) LRTDATBFn_PE1/PE2/PE3 — Local RAM test data read buffer n (n = 0, 1)

In ECC test mode (self-diagnosis), the ECC and address parity bits can be read. If the local RAM is read while ECCTST = 1 in the local RAM test control register LRTSTCTL, reading from the local RAM reads out the ECC and address parity bits, and these bits are stored in this buffer.

This register is initialized by an internal reset or an external reset.

Access: LRTDATBFn_(PE1/PE2/PE3) register is readable in 32-bit units.
LRTDATBFn_(PE1/PE2/PE3)(L/H) register is readable in 16-bit units.
LRTDATBFn_(PE1/PE2/PE3)(LL/LH/HL/HH) register is readable in 8-bit units.

Address: LRTDATBF0_PE1 : FFC6 5008H, LRTDATBF1_PE1 : FFC6 500CH,
LRTDATBF0_PE2 : FFC6 5028H, LRTDATBF1_PE2 : FFC6 502CH,
LRTDATBF0_PE3 : FF75 4008H, LRTDATBF1_PE3 : FF75 400CH,
LRTDATBFn_(PE1/PE2/PE3)L : LRTDATBFn_(PE1/PE2/PE3)+00H,
LRTDATBFn_(PE1/PE2/PE3)H : LRTDATBFn_(PE1/PE2/PE3)+02H,
LRTDATBFn_(PE1/PE2/PE3)LL : LRTDATBFn_(PE1/PE2/PE3)+00H,
LRTDATBFn_(PE1/PE2/PE3)LH : LRTDATBFn_(PE1/PE2/PE3)+01H,
LRTDATBFn_(PE1/PE2/PE3)HL : LRTDATBFn_(PE1/PE2/PE3)+02H,
LRTDATBFn_(PE1/PE2/PE3)HH : LRTDATBFn_(PE1/PE2/PE3)+03H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	LRDATABF								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	LRDATABF								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.31 LRDATBFn Register Contents

Bit Position	Bit Name	Function
31 to 25	Reserved	When read, 0 is always returned. The write value should be 0.
24 to 16	LRDATABF	These bits are valid when ECCTST = 1 (selecting test mode) in the local RAM test control register. When reading from the corresponding bank in the local RAM, the ECC bits for the local RAM (word (2n + 1)) and the address parity bit are respectively stored in LRTDATABF[22:16] and LRTDATABF[24:23].
15 to 9	Reserved	When read, 0 is always returned. The write value should be 0.
8 to 0	LRDATABF	These bits are valid when ECCTST = 1 (selecting test mode) in the local RAM test control register. When reading from the corresponding word in the local RAM, the ECC bits for the local RAM (word (2n)) and the address parity bit are respectively stored in LRTDATABF[6:0] and LRTDATABF[8:7].

(4) LRECCCTL_PE1/PE2/PE3 — Local RAM ECC control register

LRECCCTL enables or disables ECC error detection and correction and 1-bit error correction. Set the PROT[1:0] bits to 01B when writing to LRECCCTL.

LRECCCTL is initialized by an internal reset or an external reset.

Access: LRECCCTL_PE1, LRECCCTL_PE2, and LRECCCTL_PE3 are readable/writable in 32-bit units.
LRECCCTL_PE1L, LRECCCTL_PE2L, and LRECCCTL_PE3L are readable/writable in 16-bit units.

Address: LRECCCTL_PE1 : FFC6 5400H, LRECCCTL_PE2 : FFC6 5600H, LRECCCTL_PE3 : FF75 5000H,
LRECCCTL_PE1L : FFC6 5400H, LRECCCTL_PE2L : FFC6 5600H, LRECCCTL_PE3L : FF75 5000H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.32 LRECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	Reserved	The read value is 0. The write value should be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(5) LRERRINT_PE1/PE2/PE3 — Local RAM error information control

LRERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, an address parity error, or a parity bit error.

LRERRINT is initialized by an internal reset or an external reset.

Access: LRERRINT_PE1, LRERRINT_PE2, and LRERRINT_PE3 are readable/writable in 32-bit units.
LRERRINT_PE1L, LRERRINT_PE2L, and LRERRINT_PE3L are readable/writable in 16-bit units.
LRERRINT_PE1LL, LRERRINT_PE2LL, and LRERRINT_PE3LL are readable/writable in 8-bit units.

Address: LRERRINT_PE1 : FFC6 5404H, LRERRINT_PE2 : FFC6 5604H, LRERRINT_PE3 : FF75 5004H,
LRERRINT_PE1L : FFC6 5404H, LRERRINT_PE2L : FFC6 5604H, LRERRINT_PE3L : FF75 5004H,
LRERRINT_PE1LL : FFC6 5404H, LRERRINT_PE2LL : FFC6 5604H, LRERRINT_PE3LL : FF75 5004H

Value after reset: 0000 0006H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PBEIE	APEIE	DEDIE	SEDIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.33 LRERRINT Register Contents

Bit Position	Bit Name	Function
31 to 4	Reserved	The read value is 0. The write value should be 0.
3	PBEIE	Parity Bit Error Notification Enable Enables or disables generation of the error notification signal upon detection of a parity bit error when address parity check is enabled. 0: Disables notification of the parity bit error. 1: Enables notification of the parity bit error.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(6) LRSTCLR_PE1/PE2/PE3 — Local RAM status clear register

LRSTCLR clears the error flags in the error status register (LR1STERSTR) and the overflow flag in the error count overflow status register (LROVFSTR). LRSTCLR is a write-only register and is always read as 0.

Access: LRSTCLR_PE1, LRSTCLR_PE2, and LRSTCLR_PE3 are writable in 32-bit units.
LRSTCLR_PE1L, LRSTCLR_PE2L, and LRSTCLR_PE3L are writable in 16-bit units.
LRSTCLR_PE1LL, LRSTCLR_PE2LL, and LRSTCLR_PE3LL are writable in 8-bit units.

Address: LRSTCLR_PE1 : FFC6 5408$_H$, LRSTCLR_PE2 : FFC6 5608$_H$, LRSTCLR_PE3 : FF75 5008$_H$,
LRSTCLR_PE1L : FFC6 5408$_H$, LRSTCLR_PE2L : FFC6 5608$_H$, LRSTCLR_PE3L : FF75 5008$_H$,
LRSTCLR_PE1LL : FFC6 5408$_H$, LRSTCLR_PE2LL : FFC6 5608$_H$, LRSTCLR_PE3LL : FF75 5008$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR1	STCLR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 29.34 LRSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	STCLR1	Error Overflow Flag Clear (for data) Writing 1 to this bit clears the PBEF1, APEF1, DEDF1, and SEDF1 flags in LR1STERSTR, the ERROVF1 flag in LROVFSTR, and LR1STEADR1.
0	STCLR0	Error Overflow Flag Clear (for instruction) Writing 1 to this bit clears the PBEF0, APEF0, DEDF0, and SEDF0 flags in LR1STERSTR and ERROVF0 flag in LROVFSTR and LR1STEADR0.

(7) LROVFSTR_PE1/PE2/PE3 — Local RAM error count overflow status register

LROVFSTR monitors occurrence of error overflow. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in LRSTCLR.

Access: LROVFSTR_PE1, LROVFSTR_PE2, and LROVFSTR_PE3 are readable in 32-bit units.
LROVFSTR_PE1L, LROVFSTR_PE2L, and LROVFSTR_PE3L are readable in 16-bit units.
LROVFSTR_PE1LL, LROVFSTR_PE2LL, and LROVFSTR_PE3LL are readable in 8-bit units.

Address: LROVFSTR_PE1 : FFC6 540C$_H$, LROVFSTR_PE2 : FFC6 560C$_H$, LROVFSTR_PE3 : FF75 500C$_H$,
LROVFSTR_PE1L : FFC6 540C$_H$, LROVFSTR_PE2L : FFC6 560C$_H$, LROVFSTR_PE3L : FF75 500C$_H$,
LROVFSTR_PE1LL : FFC6 540C$_H$, LROVFSTR_PE2LL : FFC6 560C$_H$,
LROVFSTR_PE3LL : FF75 500C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF1	ERROVF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.35 LROVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	ERROVF1	Error Overflow Flag (for Data) This flag is set if the second error occurs while any of the error flags (PBEF1, APEF1, DEDF1, SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag (for Instruction) This flag is set if the second error occurs while any of the error flags (PBEF0, APEF0, DEDF0, SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(8) LR1STERSTR_PE1/PE2/PE3 — Local RAM 1st error status register

LR1STERSTR monitors occurrence of the first error. The error status is set if an error occurs while the error flag is 0. The error flag is overwritten if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag or parity bit error flag is set.

If more than one error occurs simultaneously, all the corresponding error flags are set. LR1STERSTR is cleared by an internal reset, an external reset, or setting 1 to the STCLR bit in LRSTCLR.

Access: LR1STERSTR_PE1, LR1STERSTR_PE2, and LR1STERSTR_PE3 are readable in 32-bit units.
LR1STERSTR_PE1L, LR1STERSTR_PE2L, and LR1STERSTR_PE3L are readable in 16-bit units.
LR1STERSTR_PE1LL, LR1STERSTR_PE1LH, LR1STERSTR_PE2LL, LR1STERSTR_PE2LH,
LR1STERSTR_PE3LL, and LR1STERSTR_PE3LH are readable in 8-bit units.

Address: LR1STERSTR_PE1 : FFC6 5410$_H$, LR1STERSTR_PE2 : FFC6 5610$_H$,
LR1STERSTR_PE3 : FF75 5010$_H$,
LR1STERSTR_PE1L : FFC6 5410$_H$, LR1STERSTR_PE2L : FFC6 5610$_H$,
LR1STERSTR_PE3L : FF75 5010$_H$,
LR1STERSTR_PE1LL : FFC6 5410$_H$, LR1STERSTR_PE1LH : FFC6 5411$_H$,
LR1STERSTR_PE2LL : FFC6 5610$_H$, LR1STERSTR_PE2LH : FFC6 5611$_H$
LR1STERSTR_PE3LL : FF75 5010$_H$, LR1STERSTR_PE3LH : FF75 5011$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PBEF1	APEF1	DEDF1	SEDF1	—	—	—	—	PBEF0	APEF0	DEDF0	SEDF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.36 LR1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 12 7 to 4	Reserved	The read value is 0. The write value should be 0.
3 + 8n	PBEFn	Parity Bit Error Monitor Flag 0: Cleared to 0 by setting 1 to the STCLRn bit in LRSTCLR. 1: Indicates that a parity bit error has occurred while all the error flags PBEFn, APEFn, DEDFn, and SEDFn are 0.
2 + 8n	APEFn	Address Parity Error Monitor Flag 0: Cleared to 0 by setting 1 to the STCLRn bit in LRSTCLR. 1: Indicates that an address parity error has occurred while the error flags DEDFn and APEFn are 0. This flag is set equally for read and write; the error generation source is not considered.
1 + 8n	DEDFn	2-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting 1 to the STCLRn bit in LRSTCLR. 1: Indicates that a 2-bit ECC error has occurred while the error flags DEDFn and APEFn are 0.
0 + 8n	SEDFn	1-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting 1 to the STCLRn bit in LRSTCLR. 1: Indicates that a 1-bit ECC error has occurred while all the error flags PBEFn, APEFn, DEDFn, and SEDFn are 0.

Note: n = 0: Error detection in the instruction; n = 1: Error detection in the data

(9) LR1STEADRn_PE1/PE2/PE3 — Local RAM 1st error address register n (n = 0, 1)

LR1STEADRn holds the address at which an error has occurred while the error flag in LR1STERSTR is set to 0. The address information is overwritten if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag or parity bit error flag is set.

Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

EADR[22:2] of this register correspond to bits [22:2] of the real address. The real address can be calculated by appending the higher-order address bits [31:23] as a base address.

LR1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit in CFSTCLR to 1.

Access: LR1STEADRn_(PE1/PE2/PE3) is readable in 32-bit units.
LR1STEADRn_(PE1/PE2/PE3)(L/H) is readable in 16-bit units.
LR1STEADRn_(PE1/PE2/PE3)(LL/LH/HL/HH) is readable in 8-bit units.

Address: LR1STEADR0_PE1 : FFC6 5450H, LR1STEADR1_PE1 : FFC6 5454H,
LR1STEADR0_PE2 : FFC6 5650H, LR1STEADR1_PE2 : FFC6 5654H,
LR1STEADR0_PE3 : FF75 5050H, LR1STEADR1_PE3 : FF75 5054H,
LR1STEADRn_(PE1/PE2/PE3)L : LR1STEADRn_(PE1/PE2/PE3) +00H,
LR1STEADRn_(PE1/PE2/PE3)H : LR1STEADRn_(PE1/PE2/PE3) +02H,
LR1STEADRn_(PE1/PE2/PE3)LL : LR1STEADRn_(PE1/PE2/PE3) +00H,
LR1STEADRn_(PE1/PE2/PE3)LH : LR1STEADRn_(PE1/PE2/PE3) +01H,
LR1STEADRn_(PE1/PE2/PE3)HL : LR1STEADRn_(PE1/PE2/PE3) +02H,
LR1STEADRn_(PE1/PE2/PE3)HH : LR1STEADRn_(PE1/PE2/PE3) +03H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	EADR[22:16]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR[15:2]														—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.37 LR1STEADRn Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	The read value is 0. The write value should be 0.
22 to 2	EADR[22:2]	1st Error Address (for Word n) These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in LR1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.
1 to 0	Reserved	The read value is 0. The write value should be 0.

Note: n = 0 (for instruction), n = 1 (for data)

The second bit is always 0 for the following registers.

LR1STEADR0_PE1

LR1STEADR0_PE2

LR1STEADR0_PE3

For example, if an error address FEF0 0000_H is stored in the above registers, an error has occurred either at FEF0 0000_H or FEF0 0004_H.

29.2.4.4 Test Function

Through appropriate register setting, desired values can be written as RAM data and to the ECC and address parity bits. Also, data in the RAM and the ECC and address parity bits can all be read.

(1) Writing the RAM data

1. Set the ECCTST bit in the LRTSTCTL register to 1 to set test mode.
2. Set the corresponding DATSEL bit in the LRTSTCTL register to 0 to select RAM data for access when writing.
3. When data is written to the local RAM, only the RAM data can be modified without updating the ECC bits.

How to exit this test mode:

4. Set the ECCTST bit in the LRTSTCTL register to 0 to disable test mode (normal mode).

(2) Reading the RAM data

1. Set the ECCDIS bit in the LRECCCTL register to 1 to disable ECC error detection and correction.
2. Read the local RAM. Since neither error detection nor correction proceeds when the local RAM is read, the RAM data is read unchanged.

How to exit this test mode:

3. Set the ECCDIS bit in the LRECCCTL register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC and address parity bits

1. Set the ECCTST bit in the LRTSTCTL register to 1 to set test mode.
2. Set the corresponding DATSEL bit in the LRTSTCTL register to 1 to select the ECC bits and the address parity bit for access when writing.
3. When data is written to the local RAM, only the ECC and address parity bits can be modified without updating the RAM data. At that time, bits [6:0] and bits [8:7] are respectively written to the ECC bits and to the address parity bit.

How to exit this test mode:

4. Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(4) Reading the ECC and address parity bits

1. Set the ECCTST bit in the LRTSTCTL register to 1 to set test mode.
2. When the local RAM is read, the ECC and address parity bits are stored in the word corresponding to local RAM test data read buffer 0 or local RAM test data read buffer 1.

How to exit this test mode:

3. Set the ECCTST bit in the LRTSTCTL register to 0 to disable test mode (normal mode).

(5) Self-diagnosis of the ECC check function

Desired values can be written as the RAM data and/or ECC bits by following the procedure described in (1) or (3) above. Therefore, a fault can be injected by, for example, inverting appropriate bits of the RAM data and/or ECC bits. After that, self-diagnosis of the ECC decoder is possible by reading the local RAM in normal mode and checking the result of error correction or detection.

(6) Self-diagnosis of the address parity check function

Self-diagnosis is enabled by following either of the two procedures below.

1. Setting APTESTi (i = 0, 1, 2, 3) in the LRTSTCTL register to 1 inverts the result of address parity generation for the corresponding word. That is, a fault can be injected to the address parity generator. Writing to the corresponding word in the local RAM in this state and checking the result of parity error detection allows self-diagnosis of the address parity check function in writing.
2. Desired data can be written to the address parity bit through the procedure described in (4) above. This enables injection of a 1-bit or 2-bit fault to the address parity bit by inverting the address parity bit. After that, self-diagnosis of the address parity bit checking function for reading is possible by reading the local RAM in normal mode and checking the result of parity error detection.

29.2.5 Global RAM ECC and Address Parity

29.2.5.1 Overview

The Global RAM ECC is summarized in the table below.

Table 29.38 Overview of the Global RAM ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. This function is enabled after a reset, that is, detection of 2-bit errors and detection and correction of 1-bit errors are enabled.
Address parity	Address parity check can be either enabled or disabled. During data write, a parity bit generated based on the written address is written simultaneously with the write data. At this time, the parity bit is written to only one location in the RAM. During data read, comparison is performed between a parity bit generated based on the read address and a parity bit read from the memory. See **Table 29.39, Address Parity** for the specification of error decoding. This function is enabled after a reset.
Error notification	Indicators of the occurrence of ECC and address parity errors are conveyed to the ECM. ECC Error: • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. The register values after a reset enable error notification upon detection of a 2-bit ECC error, and disable error notification upon detection of a 1-bit ECC error. Parity Error: • Error notification can be either enabled or disabled upon detection of an address parity error. • Error notification can be either enabled or disabled upon detection of a parity bit error. The register values after a reset enable error notification upon detection of an address parity error, and disable error notification upon detection of a parity bit error. The error notification signal is output, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error and parity bit error is handled as one source.
Error status	A status register is provided, which indicates the states of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status is set, the corresponding status is set. The error status can be cleared using the clear register.
Address capture	If an ECC error or a parity error occurs while no error status is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error, a 1-bit ECC error, or an address parity error is detected. The error status serves as the enable bit of the capture address.
Self-diagnosis	Desired values can be written as RAM data and to the ECC and address parity bits. Data in the RAM and the ECC and address parity bits can be read directly.

The definitions of the address parity error during read accesses are shown in the table below.

Table 29.39 Address Parity Definitions

Address Parity Bit		Read Address Parity	Error Determination and Error Name
0	0	0	No error
0	0	1	Address parity error
0	1	0	Parity bit error
0	1	1	Parity bit error
1	0	0	Parity bit error
1	0	1	Parity bit error
1	1	0	Address parity error
1	1	1	No error

The ECC encoder, decoder, and address parity generator are provided each for the access path (CPU1, CPU2, SubCPU, and interconnect) connected to the global RAM. The address parity checker is provided in the banks A and B of the global RAM. Also, the ECC decoder and encoder which supports RMW processing are provided for each bank A and B. Access from the SubCPU to the global RAM is made via interconnect. For details, see **Figure 29.3**.

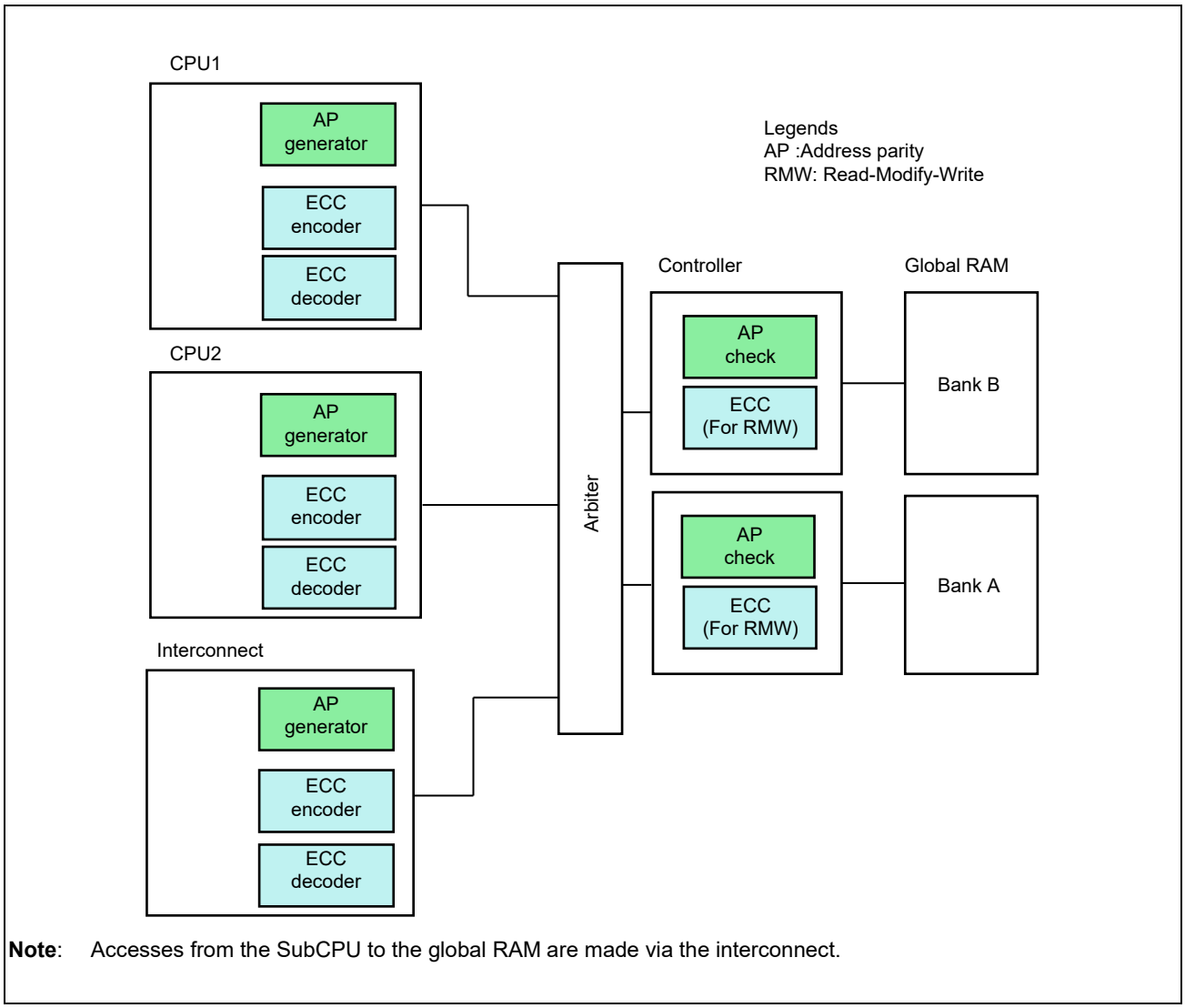
- RMW Processing

  Bit manipulation instructions, 2-byte writing and 1-byte writing are executed in three steps:

  1) Reading of 32-bit data;
  2) generation of write data by replacement of the predetermined data (modifying);
  3) writing of 32-bit data. In this section, such operations are referred to as read-modify-write (RMW) processing. RMW processing for the global RAM proceeds in the controller for each bank.

  In RWM processing, the ECC is decoded for the read operation in step 1), while the ECC is encoded for the write operation in step 3). See **Figure 29.4** for the configuration of the ECC for RMW.

  For the ECC decoder to use in updating the data, which is referred to as the write data in step 2), failure injection is possible by switching the input data from registers at which an arbitrary values can be set. See **Section 29.2.5.4, Test Function** for how to inject failures.



Note: Accesses from the SubCPU to the global RAM are made via the interconnect.

Figure 29.3 ECC and Address parity for the Global RAM

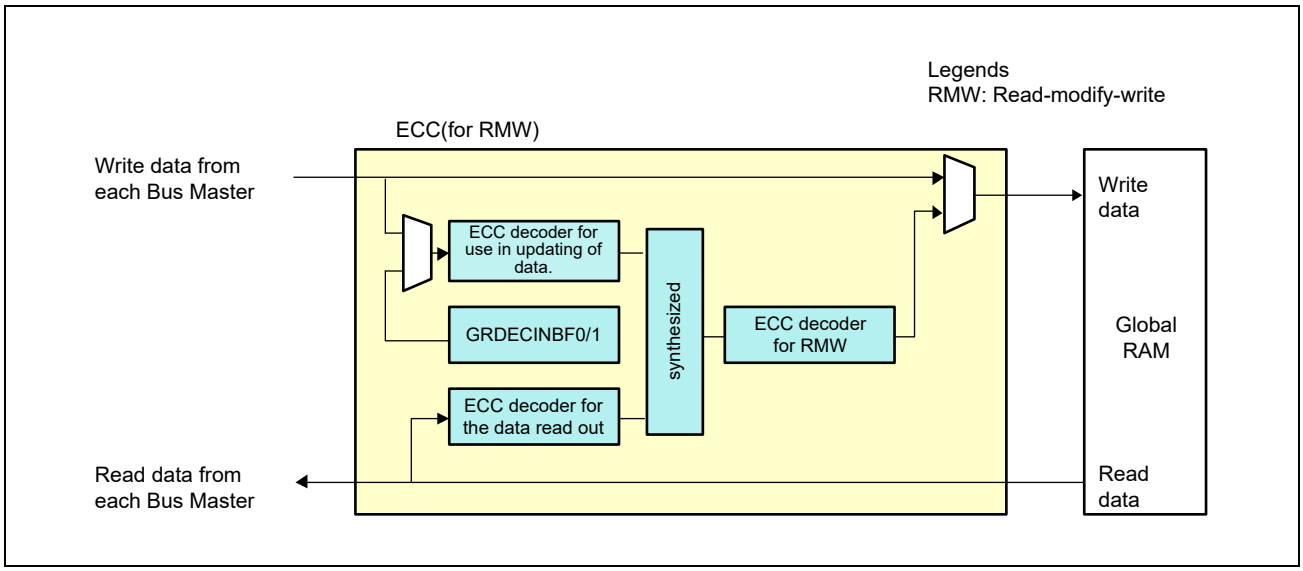


Figure 29.4 Global RAM ECC Configuration for RWM

Up to 64 bits of data can be simultaneously read from or written to the global RAM.

Each bus master (CPU1, CPU2, interconnect) has 64-bit data bus and the corresponding controllers have 128-bit data bus for each bank A and B. Meanwhile, the ECC and address parity are provided for each 32-bit data. That is, two each of the ECC decoder, ECC encoder, ECC circuit (for RMW), and address parity checker in **Figure 29.4** are provided for the individual bus masters, one for the 32 higher-order bits and the other for the 32 lower-order bits while four each of those are provided for the individual controllers, each in word 0 to 3, which are the 128 bits of data divided by four.

Word 0 to 3 are divided into even smaller units in bank A and bank B.

Table 29.40 Addresses and Corresponding ECC Circuits

4 Lower-Order Bits of the Address (in hexadecimal notation)	F_H to C_H	B_H to 8_H	7_H to 4_H	3_H to 0_H
Corresponding ECC circuit (at bus master)	Higher-order 32 bits	Lower-order 32 bits	Higher-order 32 bits	Lower-order 32 bits
Corresponding ECC circuit (at controller)	Word 3	Word 2	Word 1	Word 0

CAUTION

Global RAM is divided into bank A and B at the address FEF0 0000 H.

Note that the error information of the CPU1 and CPU2 are retained by each instruction and by each data. The error information of the interconnect are retained on the basis of the address of the data (higher-order bits or lower-order bits). See **Figure 29.5**.

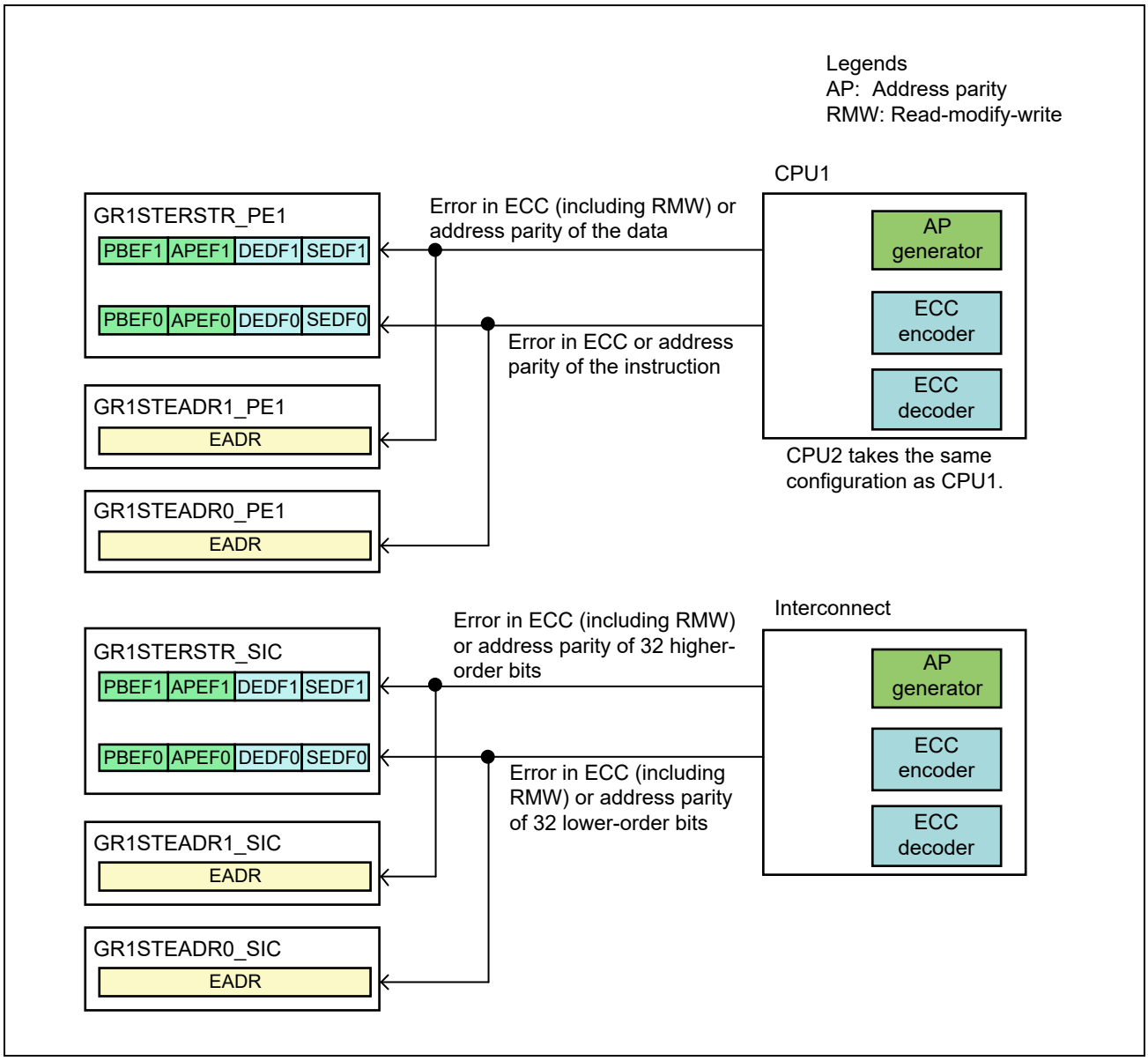


Figure 29.5 Structure for Holding Error Information in the Global RAM

29.2.5.2 List of Registers

Table 29.41 List of Registers

Module Name	Register Name	Symbol*1	Address
ECCGRAM	Global RAM ECC control register (GRAMC)	GRECCCTL_GRAMC	FFC6 4000H
ECCGRAM	Global RAM test control register	GRTSTCTL	FFC6 4180H
ECCGRAM	Global RAM test data read buffer	GRTDATBF	FFC6 4184H
ECCGRAM	Global RAM ECC decoder input data buffer 0	GRDECINBF0	FFC6 4188H
ECCGRAM	Global RAM ECC decoder input data buffer 1	GRDECINBF1	FFC6 418CH
ECCGRAM	Global RAM error information control register (SIC)	GRERRINT_SIC	FFC6 4204H
ECCGRAM	Global RAM status clear register (SIC)	GRSTCLR_SIC	FFC6 4208H
ECCGRAM	Global RAM error count overflow status register (SIC)	GROVFSTR_SIC	FFC6 420CH
ECCGRAM	Global RAM 1st error status register (SIC)	GR1STERSTR_SIC	FFC6 4210H
ECCGRAM	Global RAM 1st error (lower-order 32-bit data) address register (SIC)	GR1STEADR0_SIC	FFC6 4250H
ECCGRAM	Global RAM 1st error (higher-order 32-bit data) address register (SIC)	GR1STEADR1_SIC	FFC6 4254H
ECCGRAM	Global RAM error information control register (PE1)	GRERRINT_PE1	FFC6 4404H
ECCGRAM	Global RAM status clear register (PE1)	GRSTCLR_PE1	FFC6 4408H
ECCGRAM	Global RAM error count overflow status register (PE1)	GROVFSTR_PE1	FFC6 440CH
ECCGRAM	Global RAM 1st error status register (PE1)	GR1STERSTR_PE1	FFC6 4410H
ECCGRAM	Global RAM 1st error (instruction) address register (PE1)	GR1STEADR0_PE1	FFC6 4450H
ECCGRAM	Global RAM 1st error (data) address register (PE1)	GR1STEADR1_PE1	FFC6 4454H
ECCGRAM	Global RAM error information control register (PE2)	GRERRINT_PE2	FFC6 4604H
ECCGRAM	Global RAM status clear register (PE2)	GRSTCLR_PE2	FFC6 4608H
ECCGRAM	Global RAM error count overflow status register (PE2)	GROVFSTR_PE2	FFC6 460CH
ECCGRAM	Global RAM 1st error status register (PE2)	GR1STERSTR_PE2	FFC6 4610H
ECCGRAM	Global RAM 1st error (instruction) address register (PE2)	GR1STEADR0_PE2	FFC6 4650H
ECCGRAM	Global RAM 1st error (data) address register (PE2)	GR1STEADR1_PE2	FFC6 4654H

Note 1. The registers with symbols “_SIC”, “_PE1”, and “_PE2” as suffixes are provided to the ECC controller for each access port: the registers with “_SIC” are provided to the ECC controller for access from the system interconnect to the global RAM, the registers with “_PE1” are provided to the ECC controller for access from CPU1 to the global RAM, and the registers with “_PE2” are provided to the ECC controller for access from CPU2 to the global RAM. The registers with “_GRAMC” are control registers that are common to all access ports.

29.2.5.3 Details of Registers

(1) GRECCCTL_GRAMC — Global RAM ECC control register

GRECCCTL_GRAMC is the ECC and address parity control register shared by global RAMs. It enables or disables address parity check and specifies ECC processing. Set the PROT[1:0] bits to 01_B when writing to this register.

GRECCCTL_GRAMC is initialized by an internal reset or an external reset.

Access: GRECCCTL_GRAMC is readable/writable in 32-bit units.
GRECCCTL_GRAMCL is readable/writable in 16-bit units.

Address: GRECCCTL_GRAMC : FFC6 4000$_H$,
GRECCCTL_GRAMCL : FFC6 4000$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	APARIDIS	SECDIS	ECCDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 29.42 GRECCCTL_GRAMC Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 3	Reserved	The read value is 0. The write value should be 0.
2	APARIDIS	Address Parity Check Disable Enables or disables address parity check. In the initial state, the parity check is enabled. 0: Enables address parity check. 1: Disables address parity check.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. In the initial state, ECC error detection and correction are enabled. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction. **Caution** The encoding function is enabled even though the error detection and correction are disabled.

(2) GRTSTCTL — Global RAM test control register

This register is used for the ECC test (self-diagnosis) and address parity checker test (self-diagnosis). After ECC test mode is enabled by setting TESTEN = 1, desired values can be written to the ECC and address parity bits. The DATSEL bit is used to select the RAM data or the ECC and address parity bits. Also, input and output by the ECC decoder in the global RAM controller can be controlled for testing (self-diagnosis).

By setting address parity test mode (MAPTEST = 1, BAPTEST = 1), the parity to be input to the address parity checker is inverted. Set the PROT[1:0] bits to 01_B when writing to GRTSTCTL.

This register is initialized by an internal reset or an external reset.

Access: GRTSTCTL is readable/writable in 32-bit units.

Address: GRTSTCTL : FFC6 4180_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PROT[1:0]		—	—	—	—	MAPBIT[1:0]		MAPTEST[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAPTEST[7:0]								—	RAMSEL[2:0]			DATSEL[1:0]		DECIN EN	TESTE N
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.43 GRTSTCTL Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
29 to 26	Reserved	The read value is 0. The write value should be 0.
25 to 24	MAPBIT[1:0]	Sets the memory address parity bits. Memory address parity is used for address parity check in read accesses. The setting of these bits are enabled only when TESTEN is 1 while MAPTEST[7:0] is not 00_H. When MAPBIT = 1, the applicable address parity bit is inverted. MAPBIT[1:0] and the corresponding memory address parity bits are; MAPBIT[0]: The lower-order address parity bit MAPBIT[1]: The higher-order address parity bit The setting of these bits apply to all words and banks.
23 to 16	MAPTEST[7:0]	Sets the memory address parity checker to test mode. The setting of MAPBIT[1:0] is enabled when MAPTEST = 1. The setting of these bits is enabled only when TESTEN = 1. MAPTEST [0]: Bank A word 0 MAPTEST [1]: Bank A word 1 MAPTEST [2]: Bank A word 2 MAPTEST [3]: Bank A word 3 MAPTEST [4]: Bank B word 0 MAPTEST [5]: Bank B word 1 MAPTEST [6]: Bank B word 2 MAPTEST [7]: Bank B word 3
15 to 8	BAPTEST[7:0]	Sets the bus address parity checker to test mode. Bus address parity is used for address parity check in write accesses. When BAPTEST = 1, the parity generated through the bus address parity generator is inverted. The setting of these bits is enabled only when TESTEN = 1. BAPTEST [0]: Bank A word 0 BAPTEST [1]: Bank A word 1 BAPTEST [2]: Bank A word 2 BAPTEST [3]: Bank A word 3 BAPTEST [4]: Bank B word 0 BAPTEST [5]: Bank B word 1 BAPTEST [6]: Bank B word 2 BAPTEST [7]: Bank B word 3
7	Reserved	The read value is 0. The write value should be 0.

Table 29.43 GRTSTCTL Register Contents (2/2)

Bit Position	Bit Name	Function
6 to 4	RAMSEL[2:0]	Selects the bank and word of the global RAM from which the data is read into the GRTDATBF register. 000_B: Bank A word 0 001_B: Bank A word 1 010_B: Bank A word 2 011_B: Bank A word 3 100_B: Bank B word 0 101_B: Bank B word 1 110_B: Bank B word 2 111_B: Bank B word 3 The setting of these bits is enabled only when TESTEN = 1.
3, 2	DATSEL[1:0]	These bits select the data to be written to the RAM and the data to be read from GRTDATBF. These bits are valid when TESTEN = 1. 00_B: • GRTDATBF: The RAM data is stored when the RAM is read. • Global RAM: Only the data field in the RAM is updated when the RAM is written. The ECC bit and the address parity bit are not updated. 01_B: • GRTDATBF: The ECC bit and the address parity bit are stored when the RAM is read. • Global RAM: Only the ECC bit and the address parity bit are updated when the RAM is written. The data field is not updated. $1x_B$: Reserved (setting is prohibited)
1	DECINEN	ECC Decoder Error Injection Enable for RMW This bit is valid when TESTEN = 1. This bit enables input of the value in the ECC decoder input buffer to the ECC decoder for use in updating of data in RMW. 0: The value in the ECC decoder input buffer is not input. 1: The value in the ECC decoder input buffer is input.
0	TESTEN	ECC Test When the ECC test mode is set (TESTEN = 1), the values in the ECC bit and the address parity bit can be read directly.

(3) GRTDATBF — Global RAM test data read buffer

Data in the RAM, the ECC bits, and address parity bit can be read from this register. When the RAM is read, the value selected by the DATSEL1 or DATSEL0 bit of the GRTSTCTL register is stored in this buffer.

Access: GRTDATBF is readable in 32-bit units.
GRTDATBFL and GRTDATBFH are readable in 16-bit units.
GRTDATBFLL, GRTDATBFLH, GRTDATBFHL, and GRTDATBFHH are readable in 8-bit units.

Address: GRTDATBF : FFC6 4184H
GRTDATBFL : FFC6 4184H, GRTDATBFH : FFC6 4186H,
GRTDATBFLL : FFC6 4184H, GRTDATBFLH : FFC6 4185H,
GRTDATBFHL : FFC6 4186H, GRTDATBFHH : FF75 4187H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRTDATBF															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRTDATBF															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.44 GRTDATBF Register Contents

Bit Position	Bit Name	Function
31 to 0	GRTDATBF	These bits operate as a buffer for retaining test data set by the GRTSTCTL register. The bank and word of the global RAM at which the data are to be retained are selected by the setting of the GRTSTCTL.RAMSEL bits and the data are selected by the setting of the GRTSTCTL.DATSEL bits. These bits are only valid while TESTEN = 1. When GRTSTCTL.TESEN = 0, the values are not retained in this register. When GRTSTCTL.TESEN = 1, the data corresponding to the setting of GRTSTCTL.DATSEL are retained. When (DATSEL1, DATSEL0) = (0, 0): When the global RAM is read, the RAM data are stored in GRTDATBF [31:0]. When (DATSEL1, DATSEL0) = (0, 1): When the global RAM is read, the ECC bits and the address parity bit are respectively stored in GRTDATBF [6:0] and GRTDATBF [8:7]. 0 is stored in GRTDATBF [31:9]. When (DATSEL1, DATSEL0) = (1, x): The values are not retained in this register.

(4) GRDECINBF0 — Global RAM ECC decoder input data buffer 0

This register holds the 32 bits of data from RAM for input to the ECC decoder to use in updating the data at the time of RMW access. Self-diagnosis of the ECC decoder can be performed by using this register.

Access: GRDECINBF0 is readable/writable in 32-bit units.
GRDECINBF0L and GRDECINBF0H are readable/writable in 16-bit units.
GRDECINBF0LL, GRDECINBF0LH, GRDECINBF0HL, and GRDECINBF0HH are readable/writable in 8-bit units.

Address: GRDECINBF0 : FFC6 4188H,
GRDECINBF0L : FFC6 4188H, GRDECINBF0H : FFC6 418AH,
GRDECINBF0LL : FFC6 4188H, GRDECINBF0LH : FFC6 4189H,
GRDECINBF0HL : FFC6 418AH, GRDECINBF0HH : FFC6 418BH

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRDECINBF0															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRDECINBF0															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.45 GRDECINBF00 Register Contents

Bit Position	Bit Name	Function
31 to 0	GRDECINBF0	These bits are valid when TESTEN = 1 (selecting test mode) in the GRTSTCTL register. When DECINEN = 1, the value of this register is input to the ECC decoder as data for use in updating in response to the execution of an RMW instruction. This register is common to banks A and B, the higher-order bits and the lower-order bits, and words 0 to 3.

(5) GRDECINBF1 — Global RAM ECC decoder input data buffer 1

This register holds the 7 bits of the ECC encode data for input to the ECC decoder to use in updating the data at the time of RMW access. Self-diagnosis of the ECC decoder can be performed by using this register.

Access: GRDECINBF1 is readable/writable in 32-bit units.
GRDECINBF1L is readable/writable in 16-bit units.
GRDECINBF1LL is readable/writable in 8-bit units.

Address: GRDECINBF1 : FFC6 418C$_H$,
GRDECINBF1L : FFC6 418C$_H$,
GRDECINBF1LL : FFC6 418C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	GRDECINBF1						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.46 GRDECINBF1 Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	The read value is 0. The write value should be 0.
6 to 0	GRDECINBF1	These bits are valid when TESTEN = 1 (selecting test mode) in the GRTSTCTL register. When DECINEN = 1, the value of this register is input to the ECC decoder as the 7-bit data for use in updating in response to the execution of an RMW instruction. This register is common to banks A and B, the higher-order bits and the lower-order bits, and words 0 to 3.

(6) GRERRINT_SIC/PE1/PE2 — Global RAM error information control register

GRERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, or an address parity error.

The setting of this register is used for accesses through the respective access port.

GRERRINT is initialized by an internal reset or an external reset.

Access: GRERRINT_SIC, GRERRINT_PE1, and GRERRINT_PE2 are readable/writable in 32-bit units.
GRERRINT_SICL, GRERRINT_PE1L, and GRERRINT_PE2L are readable/writable in 16-bit units.
GRERRINT_SICLL, GRERRINT_PE1LL, and GRERRINT_PE2LL are readable/writable in 8-bit units.

Address: GRERRINT_SIC : FFC6 4204H, GRERRINT_PE1 : FFC6 4404H, GRERRINT_PE2 : FFC6 4604H,
GRERRINT_SICL : FFC6 4204H, GRERRINT_PE1L : FFC6 4404H, GRERRINT_PE2L : FFC6 4604H,
GRERRINT_SICLL : FFC6 4204H, GRERRINT_PE1LL : FFC6 4404H, GRERRINT_PE2LL : FFC6 4604H

Value after reset: 0000 0006H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PBEIE	APEIE	DEDIE	SEDIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.47 GRERRINT Register Contents

Bit Position	Bit Name	Function
31 to 4	Reserved	The read value is 0. The write value should be 0.
3	PBEIE	Parity Bit Error Notification Enable Enables or disables generation of the error notification signal upon detection of a parity bit error when address parity check is enabled. 0: Disables notification of the parity bit error. 1: Enables notification of the parity bit error.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(7) GRSTCLR_SIC/PE1/PE2 — Global RAM status clear register

GRSTCLR clears the error flags in the error status register (GR1STERSTR), the overflow flag in the error count overflow status register (GROVFSTR), and the error address register (GR1STEADR).

GRSTCLR is a write-only register and is always read as 0.

The setting of this register is used for accesses through the respective access port.

Access: GRSTCLR_SIC, GRSTCLR_PE1, and GRSTCLR_PE2 are writable in 32-bit units.
GRSTCLR_SICL, GRSTCLR_PE1L, and GRSTCLR_PE2L are writable in 16-bit units.
GRSTCLR_SICLL, GRSTCLR_PE1LL, and GRSTCLR_PE2LL are writable in 8-bit units.

Address: GRSTCLR_SIC : FFC6 4208$_H$, GRSTCLR_PE1 : FFC6 4408$_H$, GRSTCLR_PE2 : FFC6 4608$_H$,
GRSTCLR_SICL : FFC6 4208$_H$, GRSTCLR_PE1L : FFC6 4408$_H$, GRSTCLR_PE2L : FFC6 4608$_H$,
GRSTCLR_SICLL : FFC6 4208$_H$, GRSTCLR_PE1LL : FFC6 4408$_H$, GRSTCLR_PE2LL : FFC6 4608$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR1	STCLR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 29.48 GRSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	STCLR1	Error Overflow Flag Clear (PE1 and PE2: for data and for RMW access, SIC: for the 32 higher-order bits and for RMW access) Writing 1 to this bit clears the PBEF1, APEF1, DEDF1, and SEDF1 flags in GR1STERSTR, the ERROVF1 flag in GROVFSTR, and GR1STEADR1.
0	STCLR0	Error Overflow Flag Clear (PE1 and PE2: for instruction, SIC: for the 32 lower-order bits and for RMW access) Writing 1 to this bit clears the PBEF0, APEF0, DEDF0, and SEDF0 flags in GR1STERSTR, the ERROVF0 flag in GROVFSTR, and GR1STEADR0.

(8) GROVFSTR_SIC/PE1/PE2 — Global RAM error count overflow status register

GROVFSTR monitors occurrence of error overflow. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR.

The setting of this register is used for accesses through the respective access port.

Access: GROVFSTR_SIC, GROVFSTR_PE1, and GROVFSTR_PE2 are readable in 32-bit units.
GROVFSTR_SICL, GROVFSTR_PE1L, and GROVFSTR_PE2L are readable in 16-bit units.
GROVFSTR_SICLL, GROVFSTR_PE1LL, and GROVFSTR_PE2LL are readable in 8-bit units.

Address: GROVFSTR_SIC : FFC6 420C$_H$, GROVFSTR_PE1 : FFC6 440C$_H$, GROVFSTR_PE2 : FFC6 460C$_H$,
GROVFSTR_SICL : FFC6 420C$_H$, GROVFSTR_PE1L : FFC6 440C$_H$,
GROVFSTR_PE2L : FFC6 460C$_H$,
GROVFSTR_SICLL : FFC6 420C$_H$, GROVFSTR_PE1LL : FFC6 440C$_H$,
GROVFSTR_PE2LL : FFC6 460C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF1	ERROVF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.49 GROVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	ERROVF1	Error Overflow Flag (PE1 and PE2: for data and for RMW access, SIC: for the 32 higher-order bits and for RMW access) This flag is set if the second error occurs while any of the error flags (PBEF1, APEF1, DEDF1, and SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag (PE1 and PE2: for instruction, SIC: for the 32 lower-order bits and for RMW access) This flag is set if the second error occurs while any of the error flags (PBEF0, APEF0, DEDF0, and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(9) GR1STERSTR_SIC/PE1/PE2 — Global RAM 1st error status register

GR1STERSTR monitors occurrence of the first error. The error status is set if an error occurs while the error flag is 0. The applicable error flag is set if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag, the parity bit error flag, or both are set.

If more than one error occurs simultaneously, all the corresponding error flags are set. GR1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR.

The setting of this register is used for accesses through the respective access port.

Access: GR1STERSTR_SIC, GR1STERSTR_PE1, and GR1STERSTR_PE2 are readable in 32-bit units.
GR1STERSTR_SICL, GR1STERSTR_PE1L, and GR1STERSTR_PE2L are readable in 16-bit units.
GR1STERSTR_SICLL, GR1STERSTR_SICLH, GR1STERSTR_PE1LL, GR1STERSTR_PE1LH,
GR1STERSTR_PE2LL, and GR1STERSTR_PE2LH are readable in 8-bit units.

Address: GR1STERSTR_SIC : FFC6 4210$_H$, GR1STERSTR_PE1 : FFC6 4410$_H$,
GR1STERSTR_PE2 : FFC6 4610$_H$,
GR1STERSTR_SICL : FFC6 4210$_H$, GR1STERSTR_PE1L : FFC6 4410$_H$,
GR1STERSTR_PE2L : FFC6 4610$_H$,
GR1STERSTR_SICLL : FFC6 4210$_H$, GR1STERSTR_SICLH : FFC6 4211$_H$,
GR1STERSTR_PE1LL : FFC6 4410$_H$, GR1STERSTR_PE1LH : FFC6 4411$_H$,
GR1STERSTR_PE2LL : FFC6 4610$_H$, GR1STERSTR_PE2LH : FFC6 4611$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PBEF1	APEF1	DEDF1	SEDF1	—	—	—	—	PBEF0	APEF0	DEDF0	SEDF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.50 GR1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 12, 7 to 4	Reserved	The read value is 0. The write value should be 0.
3 + 8n	PBEFn	Parity Bit Error Monitor Flag Clearing condition: Setting 1 to the STCLRn bit in GRSTCLR. Setting condition: A parity bit error has occurred while all the error flags PBEFn, APEFn, DEDFn, and SEDFn are 0.
2 + 8n	APEFn	Address Parity Error Monitor Flag Clearing condition: Setting 1 to the STCLRn bit in GRSTCLR. Setting condition: An address parity error has occurred while the error flags APEFn and DEDFn are 0. **CAUTION:** This flag is set equally for read and write; the error generation source is not considered.
1 + 8n	DEDFn	2-Bit ECC Error Monitor Flag Clearing condition: Setting 1 to the STCLRn bit in GRSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flags APEFn and DEDFn are 0.
0 + 8n	SEDFn	1-Bit ECC Error Monitor Flag Clearing condition: Setting 1 to the STCLRn bit in GRSTCLR. Setting condition: A 1-bit ECC error has occurred while all the error flags PBEFn, APEFn, DEDFn, and SEDFn are 0.

NOTE

SIC: n = 0: Error detection in the 32 lower-order bits and RMW access
n = 1: Error detection in the 32 higher-order bits and RMW access
PE1/PE2: n = 0: Error detection in the instruction
n = 1: Error detection in the data and RMW access

(10) GR1STEADRn_SIC/PE1/PE2 — Global RAM 1st error address register n (n = 0, 1)

GR1STEADRn holds the address at which an error has occurred.

The error address is set if an error occurs while all the error flags are 0 in GR1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag, the parity bit error flag, or both are set. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

EADR[20:2] of this register correspond to bits [20:2] of the real address. The real address can be calculated by appending the higher-order address bits [31:21] as a base address. GR1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR. During accesses to the lower-order 32-bit data, the address is stored in GR1STEADR0. During accesses to the higher-order 32-bit data, the address is stored in GR1STEADR1.

The setting of this register is used for accesses through the respective access port.

Access: GR1STEADRn_(SIC/PE1/PE2) is readable in 32-bit units.
GR1STEADRn_(SIC/PE1/PE2)(L/H) is readable in 16-bit units.
GR1STEADRn_(SIC/PE1/PE2)(LL/LH/HL/HH) is readable in 8-bit units.

Address: GR1STEADR0_SIC : FFC6 4250H, GR1STEADR1_SIC : FFC6 4254H,
GR1STEADR0_PE1 : FFC6 4450H, GR1STEADR1_PE1 : FFC6 4454H,
GR1STEADR0_PE2 : FFC6 4650H, GR1STEADR1_PE2 : FFC6 4654H,
GR1STEADRn_(SIC/PE1/PE2)L : GR1STEADRn_(SIC/PE1/PE2) + 00H
GR1STEADRn_(SIC/PE1/PE2)H : GR1STEADRn_(SIC/PE1/PE2) + 02H
GR1STEADRn_(SIC/PE1/PE2)LL : GR1STEADRn_(SIC/PE1/PE2) + 00H
GR1STEADRn_(SIC/PE1/PE2)LH : GR1STEADRn_(SIC/PE1/PE2) + 01H
GR1STEADRn_(SIC/PE1/PE2)HL : GR1STEADRn_(SIC/PE1/PE2) + 02H
GR1STEADRn_(SIC/PE1/PE2)HH : GR1STEADRn_(SIC/PE1/PE2) + 03H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	EADR[20:16]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR[15:2]														—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.51 GR1STEADRn Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	The read value is 0. The write value should be 0.
20 to 2	EADR[20:2]	1st Error Address These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in GR1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated
1, 0	Reserved	The read value is 0. The write value should be 0.

NOTE

SIC: n = 0: Error detection in the 32 lower-order bits and RMW access
n = 1: Error detection in the 32 higher-order bits and RMW access
PE1/PE2: n = 0: Error detection in the instruction
n = 1: Error detection in the data and RMW access

The second bit is always 0 for the following registers.

GR1STEADR0_PE1

GR1STEADR0_PE2

For example, if an address FEF0 0000_H is stored in the above registers, an error has occurred either at FEF0 0000_H or FEF0 0004_H.

29.2.5.4 Test Function

Through appropriate register setting, desired values can be written as RAM data and to the ECC and address parity bits.

Also, data in the RAM and the ECC and address parity bits and the ECC decoder output data for RMW can all be read.

It is possible to input the desired data in the ECC decoder for RMW.

(1) Writing RAM data

1. Set the TESTEN bit in the GRTSTCTL register to 1 to set test mode.
2. Set the corresponding DATSEL1 to 0 and DATSEL0 to 0 in the GRTSTCTL register to select RAM data for access when writing.
3. When data is written to the global RAM, only the RAM data can be modified without updating the ECC bits.

How to exit this test mode:

4. Set the TESTEN bit in the GRTSTCTL register to 0 to disable test mode (normal mode).

(2) Reading RAM data

1. Set the ECCDIS bit in the GRECCCTL_GRAMC register to 1 to disable ECC error detection and correction.
2. Read the global RAM. Since neither error detection nor correction proceeds when the global RAM is read, the RAM data is read unchanged.

How to exit this test mode:

3. Set the ECCDIS bit in the GRECCCTL_GRAMC register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC and address parity bits

1. Set the TESTEN bit in the GRTSTCTL register to 1 to set test mode.
2. Set the corresponding DATSEL1 bit to 0 and DATSEL0 bit to 1 in the GRTSTCTL register to select the ECC bits and the address parity bit for access when writing.
3. When data is written to the global RAM, only the ECC and address parity bits can be modified without updating the RAM data. At that time, bits [6:0] and bits [8:7] are respectively written to the ECC bits and to the address parity bit.

CAUTION

Writing desired values to the RAM data and the address parity bit should be in order of 1) writing the RAM data and then 2) writing the address parity bit. When writing desired values to the RAM data and ECC bits, you can start either by writing to the RAM data or ECC bits.

How to exit this test mode:

4. Set the TESTEN bit in the GRTSTCTL register to 0 to disable test mode (normal mode).

(4) Reading the ECC and address parity bits

1. Set the TESTEN bit in the GRTSTCTL register to 1 to set test mode.
2. Setting the DATSEL1 and DATSEL0 bits in the GRTSTCTL register to 0 and 1, respectively, to select the ECC and address parity bits for access when reading.
   Set the RAMSEL bit of the GRTSTCTL register depending on the global RAM to read the data from.
3. When the global RAM is read, the ECC bits and the address parity bit are stored in the GRTDATBF register.

How to exit this test mode:

4. Set the TESTEN bit in the GRTSTCTL register to 0 to disable test mode (normal mode).

(5) Self-diagnosis of the ECC check function for the access ports

Desired values can be written to the RAM data and/or ECC bits by following the procedure described in (1) or (3) above. Therefore, a fault can be injected by, for example, inverting appropriate bits of the RAM data and/or ECC bits. After that, self-diagnosis of the ECC decoder is possible by reading the global RAM in normal mode and checking the result of error correction or detection.

(6) Self-diagnosis of the address parity check function

Self-diagnosis of write access and read access is enabled by following either of the two procedures below.

(1) In write access, setting BAPTEST[7:0] in the GRTSTCTL register to 1 inverts the result of address parity generation for the corresponding data area (words 0 to 3) of the corresponding bank (bank A or B). That is, a fault can be injected to the address parity generator. Writing to the corresponding data area of the corresponding bank in the global RAM in this state and checking the result of parity error detection allows self-diagnosis of the address parity check function in writing.

(2) In read access, setting MAPTEST[7:0] in the GRTSTCTL register to 1 inverts the result of address parity generation for the corresponding data area (words 0 to 3) of the corresponding bank (bank A or B). That is, a fault can be injected to the address parity generator. Reading from the corresponding data area of the corresponding bank in the global RAM in this state returns the address parity error according to the setting in the MAPBIT[1:0] bits. Checking the result of parity error detection allows self-diagnosis of the address parity check function in reading. Additionally, self-diagnosis is also possible by following the procedure described in (3) and (4) above, by writing a desired value in the address party bit. A 1-bit or 2-bit failure can be injected to the address parity bit by inverting the value in the bit. Then, read the global RAM in normal mode and check the result of error detection.

(7) Self-diagnosis of the ECC check function for the data read in an RMW operation

Desired values can be written to the RAM data and/or ECC bits by following the procedure described in (1) or (3) above. Therefore, a fault can be injected by, for example, inverting appropriate bits of the RAM data and/or ECC bits. After RMW processing for the global RAM proceeds, self-diagnosis of the ECC decoder is possible by checking the result of error correction or detection.

(8) Self-diagnosis of the ECC decoder for the data being updated at the time of RMW access

Setting DECINEN in the GRTSTCTL register to 1 makes the output data from the ECC decoder for the data being updated at the time of RMW access the target for reading. Though this setting, the input data from the ECC decoder is switched to the data from the ECC decoder input buffer 0 or 1 (GRDECINBF0, 1) in the global RAM from write data sent from the access ports.

As a result, suitable erroneous values can be injected by setting an appropriate value in ECC decoder input buffer 0 or 1.

After RMW processing for the global RAM proceeds, self-diagnosis of the ECC decoder is possible by checking the result of error correction or detection.

29.2.6 Instruction Cache ECC and EDC

29.2.6.1 Overview

The instruction cache ECC is summarized in the table below.

Table 29.52 Overview of the Instruction Cache ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. This function is enabled after a reset, that is, detection of 2-bit errors and detection and correction of 1-bit errors are enabled.
Address parity	None
Error notification	An indicator of the occurrence of an ECC error is conveyed to the ECM. ECC Error: • Enabling or disabling of error notification upon detection of a 2-bit ECC error can be selected. • Enabling or disabling of error notification upon detection of a 1-bit ECC error can be selected. The register values after a reset disable notification of the 2-bit ECC error and disable notification of the 1-bit ECC error. The error notification signal is output, where a 2-bit ECC error is handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the states of 2-bit ECC error detection and 1-bit ECC error detection. If an error occurs while no error status is set, the corresponding status is set. The error status can be cleared using the clear register.
Address capture	If an ECC error occurs while no error status is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error or a 1-bit ECC error is detected. The error status serves as the enable bit of the capture address.
Self-diagnosis	A cache instruction is used to write the desired values as RAM data and to the ECC bits. Similarly, data in the RAM and the ECC bits can be read directly. Since instructions as described above go through the same encoding or decoding path as a normal cache fill or instruction fetch, only such instructions can be used in inserting and confirming errors.

29.2.6.2 List of Registers

Table 29.53 List of Registers

Module Name	Register Name	Symbol	Address
ECCIC1	Instruction cache data RAM ECC control register (PE1)	IDECCCTL_PE1	FFC6 0400$_H$
ECCIC1	Instruction cache data RAM error information control register (PE1)	IDERRINT_PE1	FFC6 0404$_H$
ECCIC1	Instruction cache data RAM error status clear register (PE1)	IDSTCLR_PE1	FFC6 0408$_H$
ECCIC1	Instruction cache data RAM error count overflow status register (PE1)	IDOVFSTR_PE1	FFC6 040C$_H$
ECCIC1	Instruction cache data RAM 1st error status register (PE1)	ID1STERSTR_PE1	FFC6 0410$_H$
ECCIC1	Instruction cache data RAM 1st error address register (PE1)	ID1STEADR0_PE1	FFC6 0450$_H$
ECCIC2	Instruction cache data RAM ECC control register (PE2)	IDECCCTL_PE2	FFC6 0600$_H$
ECCIC2	Instruction cache data RAM error information control register (PE2)	IDERRINT_PE2	FFC6 0604$_H$
ECCIC2	Instruction cache data RAM error status clear register (PE2)	IDSTCLR_PE2	FFC6 0608$_H$
ECCIC2	Instruction cache data RAM error count overflow status register (PE2)	IDOVFSTR_PE2	FFC6 060C$_H$
ECCIC2	Instruction cache data RAM 1st error status register (PE2)	ID1STERSTR_PE2	FFC6 0610$_H$
ECCIC2	Instruction cache data RAM 1st error address register (PE2)	ID1STEADR0_PE2	FFC6 0650$_H$
ECCIC1	Instruction cache tag RAM ECC control register (PE1)	ITECCCTL_PE1	FFC6 1400$_H$
ECCIC1	Instruction cache tag RAM error information control register (PE1)	ITERRINT_PE1	FFC6 1404$_H$
ECCIC1	Instruction cache tag RAM error status clear register (PE1)	ITSTCLR_PE1	FFC6 1408$_H$
ECCIC1	Instruction cache tag RAM error count overflow status register (PE1)	ITOVFSTR_PE1	FFC6 140C$_H$
ECCIC1	Instruction cache tag RAM 1st error status register (PE1)	IT1STERSTR_PE1	FFC6 1410$_H$
ECCIC1	Instruction cache tag RAM 1st error address register (PE1)	IT1STEADR0_PE1	FFC6 1450$_H$
ECCIC2	Instruction cache tag RAM ECC control register (PE2)	ITECCCTL_PE2	FFC6 1600$_H$
ECCIC2	Instruction cache tag RAM error information control register (PE2)	ITERRINT_PE2	FFC6 1604$_H$
ECCIC2	Instruction cache tag RAM error status clear register (PE2)	ITSTCLR_PE2	FFC6 1608$_H$
ECCIC2	Instruction cache tag RAM error count overflow status register (PE2)	ITOVFSTR_PE2	FFC6 160C$_H$
ECCIC2	Instruction cache tag RAM 1st error status register (PE2)	IT1STERSTR_PE2	FFC6 1610$_H$
ECCIC2	Instruction cache tag RAM 1st error address register (PE2)	IT1STEADR0_PE2	FFC6 1650$_H$
ECCIC3	Instruction cache data RAM ECC control register (PE3)	IDECCCTL_PE3	FF75 1000$_H$
ECCIC3	Instruction cache data RAM error information control register (PE3)	IDERRINT_PE3	FF75 1004$_H$
ECCIC3	Instruction cache data RAM error status clear register (PE3)	IDSTCLR_PE3	FF75 1008$_H$
ECCIC3	Instruction cache data RAM error count overflow status register (PE3)	IDOVFSTR_PE3	FF75 100C$_H$
ECCIC3	Instruction cache data RAM 1st error status register (PE3)	ID1STERSTR_PE3	FF75 1010$_H$
ECCIC3	Instruction cache data RAM 1st error address register (PE3)	ID1STEADR0_PE3	FF75 1050$_H$
ECCIC3	Instruction cache tag RAM ECC control register (PE3)	ITECCCTL_PE3	FF75 2000$_H$
ECCIC3	Instruction cache tag RAM error information control register (PE3)	ITERRINT_PE3	FF75 2004$_H$
ECCIC3	Instruction cache tag RAM error status clear register (PE3)	ITSTCLR_PE3	FF75 2008$_H$
ECCIC3	Instruction cache tag RAM error count overflow status register (PE3)	ITOVFSTR_PE3	FF75 200C$_H$
ECCIC3	Instruction cache tag RAM 1st error status register (PE3)	IT1STERSTR_PE3	FF75 2010$_H$
ECCIC3	Instruction cache tag RAM 1st error address register (PE3)	IT1STEADR0_PE3	FF75 2050$_H$

29.2.6.3 Details of Registers

(1) IDECCCTL_PE1/PE2/PE3 — Instruction cache data RAM ECC control register

IDECCCTL enables or disables ECC error detection and correction and 1-bit error correction for cache data RAM. Set the PROT[1:0] bits to 01_B when writing to IDECCCTL.

IDECCCTL is initialized by an internal reset or an external reset.

Access: IDECCCTL_PE1, IDECCCTL_PE2, and IDECCCTL_PE3 are readable/writable in 32-bit units.
IDECCCTL_PE1L, IDECCCTL_PE2L, and IDECCCTL_PE3L are readable/writable in 16-bit units.

Address: IDECCCTL_PE1 : FFC6 0400_H, IDECCCTL_PE2 : FFC6 0600_H, IDECCCTL_PE3 : FF75 1000_H,
IDECCCTL_PE1L : FFC6 0400_H, IDECCCTL_PE2L : FFC6 0600_H, IDECCCTL_PE3L : FF75 1000_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.54 IDECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	Reserved	The read value is 0. The write value should be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(2) IDERRINT_PE1/PE2/PE3 — Instruction cache data RAM error information control register

IDERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error or a 1-bit ECC error in cache data RAM.

IDERRINT is initialized by an internal reset or an external reset.

Access: IDERRINT_PE1, IDERRINT_PE2, and IDERRINT_PE3 are readable/writable in 32-bit units.
IDERRINT_PE1L, IDERRINT_PE2L, and IDERRINT_PE3L are readable/writable in 16-bit units.
IDERRINT_PE1LL, IDERRINT_PE2LL, and IDERRINT_PE3LL are readable/writable in 8-bit units.

Address: IDERRINT_PE1 : FFC6 0404H, IDERRINT_PE2 : FFC6 0604H, IDERRINT_PE3 : FF75 1004H,
IDERRINT_PE1L : FFC6 0404H, IDERRINT_PE2L : FFC6 0604H, IDERRINT_PE3L : FF75 1004H,
IDERRINT_PE1LL : FFC6 0404H, IDERRINT_PE2LL : FFC6 0604H, IDERRINT_PE3LL : FF75 1004H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.55 IDERRINT Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(3) IDSTCLR_PE1/PE2/PE3 — Instruction cache data RAM error status clear register

IDSTCLR clears the error flags in the error status register (ID1STERSTR), the overflow flag in the error count overflow status register (IDOVFSTR), and the error address register (ID1STEADR).

IDSTCLR is a write-only register and is always read as 0.

Access: IDSTCLR_PE1, IDSTCLR_PE2, and IDSTCLR_PE3 are writable in 32-bit units.
IDSTCLR_PE1L, IDSTCLR_PE2L, and IDSTCLR_PE3L are writable in 16-bit units.
IDSTCLR_PE1LL, IDSTCLR_PE2LL, and IDSTCLR_PE3LL are writable in 8-bit units.

Address: IDSTCLR_PE1 : FFC6 0408H, IDSTCLR_PE2 : FFC6 0608H, IDSTCLR_PE3 : FF75 1008H,
IDSTCLR_PE1L : FFC6 0408H, IDSTCLR_PE2L : FFC6 0608H, IDSTCLR_PE3L : FF75 1008H,
IDSTCLR_PE1LL : FFC6 0408H, IDSTCLR_PE2LL : FFC6 0608H, IDSTCLR_PE3LL : FF75 1008H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 29.56 IDSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	The read value is 0. The write value should be 0.
0	STCLR0	Error Overflow Flag Clear Writing 1 to this bit clears the DEDF0 and SEDF0 flags in ID1STERSTR, ERROVF0 flag in IDOVFSTR; and ID1STEADR0.

(4) IDOVFSTR_PE1/PE2/PE3 — Instruction cache data RAM error count overflow status register

IDOVFSTR monitors occurrence of error overflow in cache data RAM. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

Access: IDOVFSTR_PE1, IDOVFSTR_PE2, and IDOVFSTR_PE3 are readable in 32-bit units.
IDOVFSTR_PE1L, IDOVFSTR_PE2L, and IDOVFSTR_PE3L are readable in 16-bit units.
IDOVFSTR_PE1LL, IDOVFSTR_PE2LL, and IDOVFSTR_PE3LL are readable in 8-bit units.

Address: IDOVFSTR_PE1 : FFC6 040C$_H$, IDOVFSTR_PE2 : FFC6 060C$_H$, IDOVFSTR_PE3 : FF75 100C$_H$,
IDOVFSTR_PE1L : FFC6 040C$_H$, IDOVFSTR_PE2L : FFC6 060C$_H$, IDOVFSTR_PE3L : FF75 100C$_H$,
IDOVFSTR_PE1LL : FFC6 040C$_H$, IDOVFSTR_PE2LL : FFC6 060C$_H$, IDOVFSTR_PE3LL : FF75 100C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.57 IDOVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	The read value is 0. The write value should be 0.
0	ERROVF0	Error Overflow Flag This flag is set if the second error occurs while any of the error flags (DEDF0 and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(5) ID1STERSTR_PE1/PE2/PE3 — Instruction cache data RAM 1st error status register

ID1STERSTR monitors occurrence of the first error in cache data RAM. The error status is set if an error occurs while all the error flags for the relevant banks are 0. The applicable error flag is set if a 2-bit ECC error occurs while the 1-bit ECC error flag is set, and the 1-bit ECC error flag is not changed.

On the other hand, the applicable error flag is not set if a 1-bit ECC error occurs while the 2-bit ECC error flag is set.

If more than one error occurs simultaneously in a bank, all the corresponding error flags are set.

ID1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

Access: ID1STERSTR_PE1, ID1STERSTR_PE2, and ID1STERSTR_PE3 are readable in 32-bit units.
ID1STERSTR_PE1L, ID1STERSTR_PE2L, and ID1STERSTR_PE3L are readable in 16-bit units.
ID1STERSTR_PE1LL, ID1STERSTR_PE2LL, and ID1STERSTR_PE3LL are readable in 8-bit units.

Address: ID1STERSTR_PE1 : FFC6 0410$_H$, ID1STERSTR_PE2 : FFC6 0610$_H$,
ID1STERSTR_PE3 : FF75 1010$_H$,
ID1STERSTR_PE1L : FFC6 0410$_H$, ID1STERSTR_PE2L : FFC6 0610$_H$,
ID1STERSTR_PE3L : FF75 1010$_H$,
ID1STERSTR_PE1LL : FFC6 0410$_H$, ID1STERSTR_PE2LL : FFC6 0610$_H$,
ID1STERSTR_PE3LL : FF75 1010$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDF0	SEDF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.58 ID1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	DEDF0	2-Bit ECC Error Monitor Flag Clearing condition: Reset or setting the STCLR0 bit to 1 in IDSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flag DEDF0 is 0.
0	SEDF0	1-Bit ECC Error Monitor Flag Clearing condition: Reset or setting the STCLR0 bit to 1 in IDSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags DEDF0 and SEDF0 are 0.

(6) ID1STEADR0_PE1/PE2/PE3 — Instruction cache data RAM 1st error address register

ID1STEADR holds the address at which an error has occurred in cache data RAM. The error address is set if an error occurs while all the error flags for the relevant banks are 0 in ID1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set for a first error. Once a 2-bit ECC error occurs, the address is not updated.

ID1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

Addresses of data RAM and tag RAM for the cached data are specified from the addresses of the code flash and this register holds the address of the cached data in the code flash. For relationship between the coder flash address and the data RAM and the tag RAM of the instruction cache, see Section **3.2.2 Instruction Cache and Data Buffer**.

ID1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

Access: ID1STEADR0_ (PE1/PE2/PE3) is readable in 32-bit units.
ID1STEADR0_ (PE1/PE2/PE3) (L/H) is readable in 16-bit units.
ID1STEADR0_ (PE1/PE2/PE3) (LL/LH/HL/HH) is readable in 8-bit units.

Address: ID1STEADR0_PE1 : FFC6 0450$_H$, ID1STEADR0_PE2 : FFC6 0650$_H$, ID1STEADR0_PE3 : FF75 1050$_H$,
ID1STEADR0_ (PE1/PE2/PE3)L : ID1STEADR0_ (PE1/PE2/PE3) + 00$_H$
ID1STEADR0_ (PE1/PE2/PE3)H : ID1STEADR0_ (PE1/PE2/PE3) + 02$_H$
ID1STEADR0_ (PE1/PE2/PE3)LL : ID1STEADR0_ (PE1/PE2/PE3) + 00$_H$
ID1STEADR0_ (PE1/PE2/PE3)LH : ID1STEADR0_ (PE1/PE2/PE3) + 01$_H$
ID1STEADR0_ (PE1/PE2/PE3)HL : ID1STEADR0_ (PE1/PE2/PE3) + 02$_H$
ID1STEADR0_ (PE1/PE2/PE3)HH : ID1STEADR0_ (PE1/PE2/PE3) + 03$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	EADR0[24:16]								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR0[15:3]													—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.59 ID1STEADR0 Register Contents

Bit Position	Bit Name	Function
31 to 25	Reserved	The read value is 0. The write value should be 0.
24 to 3	EADR0[24:3]	1st Error Address These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in ID1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error occurs, the address is not updated.
2 to 0	Reserved	The read value is 0. The write value should be 0.

(7) ITECCCTL_PE1/PE2/PE3 — Instruction cache tag RAM ECC control register

ITECCCTL enables or disables ECC error detection in cache tag RAM. Set the PROT[1:0] bits to 01_B when writing to ITECCCTL.

ITECCCTL is initialized by an internal reset or an external reset.

Access: ITECCCTL_PE1, ITECCCTL_PE2, and ITECCCTL_PE3 are readable/writable in 32-bit units.
ITECCCTL_PE1L, ITECCCTL_PE2L, and ITECCCTL_PE3L are readable/writable in 16-bit units.

Address: ITECCCTL_PE1 : FFC6 1400$_H$, ITECCCTL_PE2 : FFC6 1600$_H$, ITECCCTL_PE3 : FF75 2000$_H$,
ITECCCTL_PE1L : FFC6 1400$_H$, ITECCCTL_PE2L : FFC6 1600$_H$, ITECCCTL_PE3L : FF75 2000$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	ECCDIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.60 ECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 1	Reserved	The read value is 0. The write value should be 0.
0	ECCDIS	ECC Disable Enables or disables ECC error detection 0: Enables ECC error detection. 1: Disables ECC error detection.

(8) ITERRINT_PE1/PE2/PE3 — Instruction cache tag RAM error information control register

ITERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error or a 1-bit ECC error in cache tag RAM.

ITERRINT is initialized by an internal reset or an external reset.

Access: ITERRINT_PE1, ITERRINT_PE2, and ITERRINT_PE3 are readable/writable in 32-bit units.
ITERRINT_PE1L, ITERRINT_PE2L, and ITERRINT_PE3L are readable/writable in 16-bit units.
ITERRINT_PE1LL, ITERRINT_PE2LL, and ITERRINT_PE3LL are readable/writable in 8-bit units.

Address: ITERRINT_PE1 : FFC6 1404$_H$, ITERRINT_PE2 : FFC6 1604$_H$, ITERRINT_PE3 : FF75 2004$_H$,
ITERRINT_PE1L : FFC6 1404$_H$, ITERRINT_PE2L : FFC6 1604$_H$, ITERRINT_PE3L : FF75 2004$_H$,
ITERRINT_PE1LL : FFC6 1404$_H$, ITERRINT_PE2LL : FFC6 1604$_H$, ITERRINT_PE3LL : FF75 2004$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.61 ERRINT Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection is enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection is enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(9) ITSTCLR_PE1/PE2/PE3 — Instruction cache tag RAM error status clear register

ITSTCLR clears the error flags in the error status register (IT1STERSTR), the overflow flag in the error count overflow status register (ITOVFSTR), and the error address register (IT1STEADR).

ITSTCLR is a write-only register and is always read as 0.

Access: ITSTCLR_PE1, ITSTCLR_PE2, and ITSTCLR_PE3 are writable in 32-bit units.
ITSTCLR_PE1L, ITSTCLR_PE2L, and ITSTCLR_PE3L are writable in 16-bit units.
ITSTCLR_PE1LL, ITSTCLR_PE2LL, and ITSTCLR_PE3LL are writable in 8-bit units.

Address: ITSTCLR_PE1 : FFC6 1408$_H$, ITSTCLR_PE2 : FFC6 1608$_H$, ITSTCLR_PE3 : FF75 2008$_H$,
ITSTCLR_PE1L : FFC6 1408$_H$, ITSTCLR_PE2L : FFC6 1608$_H$, ITSTCLR_PE3L : FF75 2008$_H$,
ITSTCLR_PE1LL : FFC6 1408$_H$, ITSTCLR_PE2LL : FFC6 1608$_H$, ITSTCLR_PE3LL : FF75 2008$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 29.62 STCLR Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	The read value is 0. The write value should be 0.
0	STCLR0	Error Overflow Flag Clear Writing 1 to this bit clears the DEDF0 and SEDF0 flags in IT1STERSTR, ERROVF0 flag in ITOVFSTR, and IT1STEADR0.

(10) ITOVFSTR_PE1/PE2/PE3 — Instruction cache tag RAM error count overflow status register

ITOVFSTR monitors occurrence of error overflow in cache tag RAM. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

Access: ITOVFSTR_PE1, ITOVFSTR_PE2, and ITOVFSTR_PE3 are readable in 32-bit units.
ITOVFSTR_PE1L, ITOVFSTR_PE2L, and ITOVFSTR_PE3L are readable in 16-bit units.
ITOVFSTR_PE1LL, ITOVFSTR_PE2LL, and ITOVFSTR_PE3LL are readable in 8-bit units.

Address: ITOVFSTR_PE1 : FFC6 140C$_H$, ITOVFSTR_PE2 : FFC6 160C$_H$, ITOVFSTR_PE3 : FF75 200C$_H$,
ITOVFSTR_PE1L : FFC6 140C$_H$, ITOVFSTR_PE2L : FFC6 160C$_H$, ITOVFSTR_PE3L : FF75 200C$_H$,
ITOVFSTR_PE1LL : FFC6 140C$_H$, ITOVFSTR_PE2LL : FFC6 160C$_H$, ITOVFSTR_PE3LL : FF75 200C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.63 ITOVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	The read value is 0. The write value should be 0.
0	ERROVF0	Error Overflow Flag This flag is set if the second error occurs while any of the error flags (DEDF0 and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(11) IT1STERSTR_PE1/PE2/PE3 — Instruction cache tag RAM 1st error status register

IT1STERSTR monitors occurrence of the first error in cache tag RAM. The error status is set if an error occurs while all the error flags of the IT1STERSTR register are 0. The applicable error flag is set if a 2-bit ECC error occurs while the 1-bit ECC error flag is set, and the 1-bit ECC error flag is not changed.

On the other hand, the applicable error flag is not set if a 1-bit ECC error occurs while the 2-bit ECC error flag is set.

This register cannot specify the Way at which an error has occurred.

IT1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

Access: IT1STERSTR_PE1, IT1STERSTR_PE2, and IT1STERSTR_PE3 are readable in 32-bit units.
IT1STERSTR_PE1L, IT1STERSTR_PE2L, and IT1STERSTR_PE3L are readable in 16-bit units.
IT1STERSTR_PE1LL, IT1STERSTR_PE2LL, and IT1STERSTR_PE3LL are readable in 8-bit units.

Address: IT1STERSTR_PE1 : FFC6 1410$_H$, IT1STERSTR_PE2 : FFC6 1610$_H$,
IT1STERSTR_PE3 : FF75 2010$_H$,
IT1STERSTR_PE1L : FFC6 1410$_H$, IT1STERSTR_PE2L : FFC6 1610$_H$,
IT1STERSTR_PE3L : FF75 2010$_H$,
IT1STERSTR_PE1LL : FFC6 1410$_H$, IT1STERSTR_PE2LL : FFC6 1610$_H$,
IT1STERSTR_PE3LL : FF75 2010$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDF0	SEDF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.64 IT1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When read, 0 is always returned. The write value should be 0.
1	DEDF0	2-Bit ECC Error Monitor Flag Clearing condition: Reset or setting the STCLR0 bit to 1 in ITSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flag DEDF0 is 0.
0	SEDF0	1-Bit ECC Error Monitor Flag Clearing condition: Reset or setting the STCLR0 bit to 1 in ITSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags DEDF0 and SEDF0 are 0.

(12) IT1STEADR0_PE1/PE2/PE3 — Instruction cache tag RAM 1st error address register

IT1STEADR holds the address at which an error has occurred in cache tag RAM. The error address is set if an error occurs while all the error flags for the relevant banks are 0 in IT1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error occurs, the address will not be updated. Addresses of data RAM and tag RAM for the cached data are specified from the addresses of the code flash. This register holds the address of the cached data in the code flash.

For relationship between the coder flash address and the data RAM and the tag RAM of the instruction cache, see **Section 3.2.2, Instruction Cache and Data Buffer**.

IT1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

Access: IT1STEADR0_ (PE1/PE2/PE3) is readable in 32-bit units.
IT1STEADR0_ (PE1/PE2/PE3) (L/H) is readable in 16-bit units.
IT1STEADR0_ (PE1/PE2/PE3) (LL/LH/HL/HH) is readable in 8-bit units.

Address: IT1STEADR0_PE1 : FFC6 1450H, IT1STEADR0_PE2 : FFC6 1650H, IT1STEADR0_PE3 : FF75 2050H,
IT1STEADR0_ (PE1/PE2/PE3)L : IT1STEADR0_ (PE1/PE2/PE3) + 00H,
IT1STEADR0_ (PE1/PE2/PE3)H : IT1STEADR0_ (PE1/PE2/PE3) + 02H,
IT1STEADR0_ (PE1/PE2/PE3)LL : IT1STEADR0_ (PE1/PE2/PE3) + 00H,
IT1STEADR0_ (PE1/PE2/PE3)LH : IT1STEADR0_ (PE1/PE2/PE3) + 01H,
IT1STEADR0_ (PE1/PE2/PE3)HL : IT1STEADR0_ (PE1/PE2/PE3) + 02H,
IT1STEADR0_ (PE1/PE2/PE3)HH : IT1STEADR0_ (PE1/PE2/PE3) + 03H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	EADR[24:16]								
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR[15:5]											—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.65 IT1STEADR0 Register Contents

Bit Position	Bit Name	Function
31 to 25	Reserved	The read value is 0. The write value should be 0.
24 to 5	EADR[24:5]	1st Error Address These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in IT1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.
4 to 0	Reserved	The read value is 0. The write value should be 0.

29.2.6.4 Test Function

A cache instruction is used to write the desired values as RAM data and to the ECC bits, and read data in the RAM and the ECC bits directly.

Since instructions as described above go through the same encoding or decoding path as a normal cache fill or instruction fetch, only such instructions can be used in inserting and confirming errors.

For details, refer to *RH850 Family User's Manual: Software*.

29.2.7 DTS RAM ECC

See **Section 7, DMA Controller**.

29.2.8 ECC for Peripheral RAM (32 Bits)

29.2.8.1 Overview

This is an ECC module for the RAM of the following peripheral modules.

RS-CAN, CSIH

Error Detection and Correction

7-bit ECC data is appended to the 32-bit RAM data.

This ECC module provides 2-bit ECC error detection and 1-bit ECC error detection and correction.

CAUTION

This module is not capable of reliably detecting errors in three or more bits.

If errors occur in three or more bits, the module may detect the errors as 1- or 2-bit ECC errors or not detect any errors.

Depending on the settings, this may lead to the correction of a bit that was not actually inverted.

Enabling or disabling ECC error detection and correction

- ECC error detection can be either enabled or disabled.
- 1-bit ECC error correction can be either enabled or disabled.
- If all the bits of RAM output data are stuck at 0 or 1, it is detected as a 2-bit ECC error.

Error notification

- The ECM is notified when 2-bit ECC errors are detected (this can be enabled or disabled).
- The ECM is notified when 1-bit ECC errors are detected (this can be enabled or disabled).
- Once the ECM has been notified of an error, even if another ECC error is detected, the ECM is not notified until the error status bit corresponding to the initial error is cleared.

Error status

- Detection of 2- and 1-bit ECC errors can be monitored.
- Special registers are provided to clear error status.

Address capture

- Only one address at which an ECC error has occurred can be captured.
- The address at which an error has occurred is captured for a 2-bit or 1-bit ECC error that was first detected since the corresponding error monitoring flag was cleared.

Test function (Error injection)

- By setting the test mode, register values can be used as the data to be output to the RAM. When a peripheral module writes to the RAM, the value of the ECEDB[31:0] bits can be written to the RAM data section, and the value of the ECERDB[6:0] bits can be written to the ECC redundant bit section.
- By setting the test mode, the ECC redundant bit section can be latched when RAM data is read, and the value can be confirmed.
- By setting the test mode, the ECC redundant bit (encoding circuit) and syndrome code (decoding circuit), which are generated from the input data, can be confirmed.

29.2.8.2 List of Registers

(1) List of ECC modules

The RAMs of the multiple peripheral functions are provided with the ECC modules. The following table shows the peripheral functions provided with the ECC modules, the corresponding ECC module names, and base addresses of the ECC modules.

Table 29.66 List of ECC Modules

Supported Peripheral Function		ECC Module Names and Register Base Addresses			
		Master Side*1		Checker Side*1	
		Module Name	Base Address <base_addr>	Module Name	Base Address <base_addr>
RS-CANFD	Message buffer RAM (MB RAM)	E7RC1M	FFC71000$_H$	E7RC1C	FFC71200$_H$
	Acceptance filter list RAM (AFL RAM)	E7RC2M	FFC71400$_H$	E7RC2C	FFC71600$_H$
CSIH	CSIH0	E7CS0M	FFC70000$_H$	E7CS0C	FFC70200$_H$
	CSIH1	E7CS1M	FFC70400$_H$	E7CS1C	FFC70600$_H$
	CSIH2	E7CS2M	FFC70800$_H$	E7CS1C	FFC70A00$_H$

Note 1. Two modules (one for master and the other for checker) are provided for duplex configuration.

(2) List of registers

Each ECC module has the registers shown in the following table.

Table 29.67 List of Registers

Module Name	Register Name	Symbol	Address
*1	ECC control register*2	E710CTL	<base_addr>+00_H
*1	ECC test mode control register	E710TMC	<base_addr>+04_H
*1	ECC redundant bit data control test register	E710TRC	<base_addr>+08_H
*1	ECC encoder and decoder data test register	E710TED	<base_addr>+$0C_H$
*1	ECC error address register	E710EAD	<base_addr>+10_H

Note 1. For the names of each ECC module, see **Table 29.66 List of ECC Modules**.

Note 2. The reset value of the LSB in the ECC control register is undefined.

(3) Register map

Table 29.68 Register Map

Abbreviation	31 24	23 16	15 8	7 0	Offset address
E710CTL	— (00_H)	— (00_H)	ECCTL[15:8]	ECCTL[7:0]	00_H
E710TMC	— (00_H)	— (00_H)	ECTMC[15:8]	ECTMC[7:0]	04_H
E710TRC	ECSYND[7:0]	ECHORD[7:0]	ECECRD[7:0]	ECERDB[7:0]	08_H
E710TED	ECEDB[31:24]	ECEDB[23:16]	ECEDB[15:8]	ECEDB[7:0]	$0C_H$
E710EAD	ECEAD[31:24]	ECEAD[23:16]	ECEAD[15:8]	ECEAD[7:0]	10_H

29.2.8.3 Details of Registers

(1) E710CTL — ECC control register

E710CTL controls the status and modes of the ECC module.

E710CTL can be read and written to using the 16-bit or 8-bit manipulation instruction.

However, only the 16-bit manipulation instruction is valid for writing to bit 7.

Access: E710CTL is readable/writable in 16-bit units.
E710CTLL and E710CTLH are readable/writable in 8-bit units.

Address: E710CTL : <base_addr> + 00_H,
E710CTLL : <base_addr> + 00_H,
E710CTLH : <base_addr> + 01_H

Value after reset: $001X_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EMCA[1:0]		—	—	ECOVFF	ECER2C	ECER1C	—	ECTHM	—	EC1ECP	EC2EDIC	EC1EDIC	ECER2F	ECER1F	ECEMF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	—
R/W	W*1	W*1	R	R	R	R/W*1	R/W*1	R	R/W	R	R/W	R/W	R/W	R	R	R

Note 1. Reading this bit always returns 0.

Table 29.69 E710CTL Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	EMCA[1:0]	Access Control 1 and 0 to ECC Mode Select Bit These bits reserve the write trigger to bit 7. The read value is always 0.
13, 12	Reserved	The read value is 0. The write value should be 0.
11	ECOVFF	Error Overflow Detection Flag This flag is set if the second error occurs while any of the error flags (ECER2F and ECER1F) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error. This flag is cleared by an internal reset, an external reset, when through mode is enabled (ECTHM = 1), or setting 1 to the ECER2C and ECER1C bits.
10	ECER2C	2-Bit ECC Error Detection Flag Clear This bit clears the bit 2 (ECER2F) status flag. The read value is always 0, and writing 0 to ECER2C does not change the value. If writing 1 to ECER2C conflicts with the condition for setting bit 2, the former takes priority. Writing 1 to ECER2C while ECER2F is set clears ECER2F.
9	ECER1C	1-Bit ECC Error Detection Flag Clear This bit clears the bit 1 (ECER1F) status flag. The read value is always 0, and writing 0 to ECER1C does not change the internal state. If writing 1 to ECER1C conflicts with the condition for setting bit 1, the former takes priority. Writing 1 to ECER1C while ECER1F is set clears ECER1F.
8	Reserved	The read value is 0. The write value should be 0.

Table 29.69 E710CTL Register Contents (2/2)

Bit Position	Bit Name	Function
7	ECTHM	ECC Function Disable Select Selects the ECC decoding operation. Write access to ECTHM is enabled when the value of bits 15 and 14 is 01_B. Therefore, only the 16-bit manipulation instruction is valid. Setting ECTHM to 1 disables error detection and bit correction. Here, if the data to be output to the peripheral module contains an error, the data is output without bit correction. Setting ECTHM to 1 has no effect on the encoder side. 0: ECC detection and correction enabled. 1: ECC detection and correction disabled.
6	Reserved	The read value is 0. The write value should be 0.
5	EC1ECP	1-Bit ECC Error Correction Enable 0: Enables 1-bit error correction upon error detection. 1: Disables 1-bit error correction upon error detection.
4	EC2EDIC	2-Bit ECC Error Detection Notification Enable 0: When a 2-bit error is detected, the ECM is not notified of the error. 1: When a 2-bit error is detected, the ECM is notified of the error.
3	EC1EDIC	1-Bit ECC Error Detection Notification Enable 0: When a 1-bit error is detected, the ECM is not notified of the error. 1: When a 1-bit error is detected, the ECM is notified of the error.
2	ECER2F	2-Bit ECC Error Detection Flag Indicates that errors have been detected in two bits among bits 0 to 38 of data read from the RAM while error detection is enabled. This bit is read-only. 0: A 2-bit error has not occurred. 1: A 2-bit error has occurred. [Clearing conditions] (1) Reset is applied. (2) 1 is written to ECER2C. (3) ECC detection and correction are disabled (ECTHM = 1).
1	ECER1F	1-Bit ECC Error Detection And Correction Flag Indicates that an error has been detected in one bit among bits 0 to 38 of data read from the RAM while error detection is enabled. This bit is read-only. 0: A 1-bit error has not occurred. 1: A 1-bit error has occurred. [Clearing conditions] (1) Reset is applied. (2) 1 is written to ECER1C. (3) The ECC detection and correction are disabled (ECTHM = 1).
0	ECEMF	ECC Error Message Flag Indicates that the current read data bus contains an error. ECEMF is updated every time RAM data is output. Since the RAM value after a reset is undefined, it is determined to be an error, and thus ECEMF may be set. Therefore, the ECEMF bit value after a reset is undefined. 0: The current read data bus contains no bit errors. 1: The current read data bus contains bit errors. ECEMF remains set as long as RAM data containing a bit error is output with error detection being enabled. [Clearing conditions] (1) ECC detection and correction are disabled (ECTHM = 1). (2) Decoding circuit input data contains no 1-bit errors. ECEMF remains set as long as RAM data containing a bit error is output with error detection being enabled.

(2) E710TMC — ECC test mode control register

E710TMC is a 16-bit register to switch the mode to test mode and controls the mode.

E710TMC can be read and written to using the 16-bit or 8-bit manipulation instruction.

However, only the 16-bit manipulation instruction is valid for writing to bit 7.

Access: E710TMC is readable/writable in 16-bit units.
E710TMCL and E710TMCH are readable/writable in 8-bit units.

Address: E710TMC: <base_addr> + 04_H,
E710TMCL : <base_addr> + 04_H,
E710TMCH : <base_addr> + 05_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ETMA[1:0]		—	—	—	—	—	—	ECTMCE	—	—	ECTRRS	ECREOS	ECENS	ECDCS	ECREIS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*1	W*1	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Note 1. Reading this bit always returns 0.

Table 29.70 E710TMC Register Contents

Bit Position	Bit Name	Function
15, 14	ETMA[1:0]	Access Control 1 and 0 to ECC Test Mode Control Enable Bit These bits reserve the write trigger to bit 7. The read value is always 0.
13 to 8	Reserved	The read value is 0. The write value should be 0.
7	ECTMCE	ECC Test Mode Enable ECTMCE enables or disables access to the test registers and test control bits. Access to ETCMCE is enabled when the value of bits 15 and 14 is 10_B. 0: Disables access to the test registers and test control bits. 1: Enables access to the test registers and test control bits.
6, 5	Reserved	The read value is 0. The write value should be 0.
4	ECTRRS	ECC RAM Read Test Mode Select ECTRRS enables the external read status of RAM to be generated by reading the E710TED register, and also allows the RAM output data to be read when the E710TRC.ECERDB[7:0] bits and the E710TED register are read. Write access to ECTRRS is enabled only when ECTMCE is 1 (can be set simultaneously). ECTRRS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Disables generation of RAM read status for testing when E710TED is read. 1: Enables generation of RAM read status for testing when E710TED is read. When E710TRC.ECERDB[7:0] and E710TED are read, the values of the RAM output data pin are read.
3	ECREOS	ECC Redundant Bit Output Data Select ECREOS selects either the ECC encoder output data or the ECERDB register value to be output as the ECC redundant bit output. Write access to ECREOS is enabled only when ECTMCE is 1 (can be set simultaneously). ECREOS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows encoding result to be output as the ECC redundant bit output. 1: Allows the E710TRC.ECERDB[6:0] value to be output as the ECC redundant bit output.
2	ECENS	ECC Encoder Input Select ECENS selects either the data value from the peripheral module or the internal test register value (E710TED.ECEDB[31:0]) as the input signal to be encoded. Write access to ECENS is enabled only when ECTMCE is 1 (can be set simultaneously). ECENS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the RAM write data from the peripheral module to be input as the ECC encoder input data. 1: Allows the E710TED.ECEDB[31:0] value to be input as the ECC encoder input data.
1	ECDCS	ECC Decoder Input Select ECDCS selects either the lower 32-bit data value from the RAM or the internal test register value (E710TED.ECEDB[31:0]) as the lower 32-bit data of the input signal to be decoded. Write access to ECDCS is enabled only when ECTMCE is 1 (can be set simultaneously). ECDCS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the lower 32-bit RAM output data to be input to the data area (32 lower-order bits) to the decoder. 1: Allows the E710TED.ECEDB[31:0] value to be input to the data area to the decoder.
0	ECREIS	ECC Redundant Bit Input Data Select ECREIS selects either the upper 7-bit data value from the RAM (redundant bit area) or the test register value (E710TRC.ECERDB[6:0]) as the upper 7-bit data of the input signal to be decoded. Write access to ECREIS is enabled only when ECTMCE is 1 (can be set simultaneously). ECREIS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the upper 7-bit RAM output data to be input to the ECC redundant bit area to the decoder. 1: Allows the E710TRC.ECERDB[6:0] value to be input to the ECC redundant bit area to the decoder.

(3) E710TED — ECC encoder and decoder data test register

E710TED is a 32-bit data test register for ECC encoding and decoding.

When ECTMCE = 1, E710TED can be read and written to using the 32-bit manipulation instruction.

When ECTMCE = 0, E710TED is always read as 0.

In test mode, the E710TED value can be used as the input data to the encoding circuit and decoding circuit.

Access: E710TED is readable/writable in 32-bit units.

Address: E710TED: <base_addr> + $0C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECEDB[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECEDB[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. When ECTMCE = 0, the read value is always 0. When writing, write 0.

NOTE

Changing ECTMCE from 1 to 0 resets E710TED synchronously.

When E710TMC:ECENS = 1, the value of this register is input to the encoding circuit and supplied to the RAM.

When E710TMC:ECDCS = 1, the value of this register is input as the bits 31-0 of the input data to the decoding circuit.

When E710TMC:ECTRRS = 1, reading this register returns the RAM output data instead of the data written to this register

(4) E710TRC — ECC redundant bit data control test register

E710TRC is a 32-bit test register consisting of four fields (ECSYND, ECHORD, ECECRD, and ECERDB) corresponding to the ECC redundant bit area. Each field can be accessed as the 8-bit register with the same name. For details of each field, refer to the descriptions of these four registers.

Access: ECSYND, ECHORD, ECECRD, and ECERDB are simultaneously accessible by E710TRC in 32-bit units.

Address: E710TRC: <base_addr> + 08_H

Value after reset: See the descriptions of each register ECSYND, ECHORD, ECECRD, and ECERDB.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECSYND								ECHORD							
Value after reset:	*1	*1	*1	*1	*1	*1	*1	*1	*2	*2	*2	*2	*2	*2	*2	*2
R/W	*1	*1	*1	*1	*1	*1	*1	*1	*2	*2	*2	*2	*2	*2	*2	*2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECECRD								ECERDB							
Value after reset:	*3	*3	*3	*3	*3	*3	*3	*3	*4	*4	*4	*4	*4	*4	*4	*4
R/W	*3	*3	*3	*3	*3	*3	*3	*3	*4	*4	*4	*4	*4	*4	*4	*4

Note 1. See the description of the ECSYND register.

Note 2. See the description of the ECHORD register.

Note 3. See the description of the ECECRD register.

Note 4. See the description of the ECERDB register.

NOTE

Changing ECTMCE from 1 to 0 resets E710TRC synchronously.

(5) ECSYND — ECC decoder syndrome data register

ECSYND is a read-only register to confirm the syndrome code generated by the decoding circuit in test mode (ECTMCE = 1).

Write access to ECSYND is ignored.

Access: E710TRC (.ECSYND) is readable/writable in 32-bit units.
(ECSYND as a single register is readable in 8-bit units.)

Address: E710TRC (.ECSYND): <base_addr> + 08_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24
	—	SYND[6:0]						
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

When read, the ECSYND bits return the value of the syndrome code (synd[6:0]) generated based on the input data to the decoding circuit.

The ECSYND bits are not holding circuits; the register value changes as the input signal changes.

ECSYND is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

Bit 31 is a reserved bit. It is always read as 0 and the write value should be 0.

(6) ECHORD — ECC -bit redundant bit data holding test register

ECHORD holds the 7-bit ECC redundant area (higher-order 7 bits of RAM data), which cannot be confirmed by the peripheral module, when the peripheral module accesses the RAM for reading in test mode (ECTMCE = 1).

Access: E710TRC (.ECHORD) is readable/writable in 32-bit units.
(ECHORD as a single register is readable in 8-bit units.)

Address: E710TRC (.ECHORD): <base_addr> + 08_H

Value after reset: 0000 0000_H

Bit	23	22	21	20	19	18	17	16
	—	HORD[6:0]						
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

The ECHORD bits are loaded with the higher-order 7 bits of RAM output data at the next rising edge of the operating clock signal after the peripheral module accesses the RAM for reading data in test mode (ECTMCE = 1).

The ECHORD bits are also loaded with the higher-order 7 bits of RAM output data at the next rising edge of the operating clock signal when the ECEDB[15:0] bits are read while E710TMC.ECTRRS = 1.

ECHORD is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

Bit 23 is a reserved bit. It is always read as 0 and the write value should be 0.

(7) ECECRD — ECC encode test register

ECECRD is a read-only register to read the 7-bit redundant section generated by the encoding circuit in test mode (ECTMCE = 1).

Access: E710TRC (.ECECRD) is readable/writable in 32-bit units.
(ECECRD as a single register is readable in 8-bit units.)

Address: E710TRC (.ECECRD): <base_addr> + 08_H

Value after reset: 0000 0000_H

Bit	15	14	13	12	11	10	9	8
	—	ECRD[6:0]						
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

ECECRD is used to confirm the redundant bits generated by the input data from the peripheral module.

The read data is the result of encoding (ecc[6:0]). It is not in the output value.

ECECRD is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

Bit 15 is a reserved bit. It is always read as 0 and the write value should be 0.

(8) ECERDB — ECC redundant bit input and output substitution buffer register

ECERDB is a buffer register for the data that substitutes for the input and output data for the 7-bit ECC redundant data area in test mode (ECTMCE = 1).

ECERDB can be read and written to in ECC test mode (ECTMCE = 1).

Access: E710TRC (.ECERDB) readable/writable in 32-bit units.
(ECERDB as a single register is readable/writable in 8-bit units.)

Address: E710TRC (.ECERDB): <base_addr> + 08_H

Value after reset: $0000\ 0000_H$

Bit	7	6	5	4	3	2	1	0
	—	ERDB[6:0]						
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When ECREOS = 1, the ECERDB value, instead of the seven redundant bits generated by the encoding circuit, is output to the pin to be supplied to the RAM.

When ECREIS = 1, the ECERDB value, instead of the higher-order seven data bits to be input to the decoding circuit, is handled by the decoding circuit.

When ECTRRS = 1, reading ECERDB returns the signal value supplied to RAM instead of the data written to ECERDB.

Bit 7 is a reserved bit. It is always read as 0 and the write value should be 0.

(9) E710EAD — ECC error address register

E710EAD is a read-only register to hold the address at which an ECC error has occurred.

If an ECC error is detected while ECC error detection is enabled, the RAM address is latched using the detection signal as a trigger, and the address is stored in E710EAD as the address at which the ECC error has occurred.

The address is stored upon detection of the first ECC error while no error status is set. However, if a 1-bit error is followed by a 2-bit error, the address of the latter is stored.

Only one address can be held in E710EAD.

Access: E710EAD is readable in 32-bit units.

Address: E710EAD: <base_addr> + 10_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECEAD[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECEAD[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

29.2.8.4 Notification to ECM

This module can detect two error signals through programming. The ECM is notified of the detected error.

- 1-bit error notification

  If an error is detected in one bit among bits 0 to 38 of data read from the RAM while 1-bit error notification is enabled (EC1EDIC = 1_B), the 1-bit error is indicated. However, it will not be indicated if ECER1F or ECER2F is already set.

- 2-bit error notification

  If errors are detected in two bits among bits 0 to 38 of data read from the RAM while 2-bit error notification control is enabled (EC2EDIC = 1_B), the 2-bit error is indicated. However, it will not be indicated if ECER2F is already set.

29.2.8.5 Test Function

(1) Writing to RAM data

Write data to the peripheral RAM. However, the ECC corresponding to the written data will be written to the ECC bits simultaneously. In order to write a specified value to the ECC bits, use the ECC test mode described in (3) below.

(2) Reading RAM data

1. Set the ECTHM bit in the ECC control register to 1 to disable ECC error detection and correction.
2. Read the peripheral RAM. Since neither error detection nor correction proceeds when the peripheral RAM is read, the RAM data is read unchanged.

How to exit this test mode:

3. Set the ECTHM bit in the ECC control register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC bits

1. Set the ECTMCE bit in the ECC test mode control register to 1 to set ECC test mode.
2. Write the value for writing to the ECC bits to the E710TRC.ECERDB[6:0] bits.
3. Set the ECREOS bit in the ECC test mode control register to 1 to select writing of the value of the E710TRC.ECERDB[6:0] bits to the ECC bits.
4. When data are written to the peripheral RAM, the value in the E710TRC.ECERDB[6:0] bits will be written to the ECC bits.

How to exit this test mode:

5. Set the ECTMCE bit in the ECC test mode control register to 0 to set normal mode.

(4) Reading the ECC bits

1. Set the ECTMCE bit in the ECC test mode control register to 1 to set ECC test mode.
2. When data in the peripheral RAM is read, the ECC bits are stored in the E710TRC.ECHORD[6:0] bits.

How to exit this test mode:

3. Set the ECTMCE bit in the ECC test mode control register to 0 to set normal mode.

29.2.9 Data Parity for Data Transfer Path

The following shows the transfer paths to which data parity is applied. Data parity errors can be detected during data transfer from the access sources to destinations shown below. If a parity error is detected in any of the paths, the error is notified to the ECM.

Table 29.71 Transfer Path to which Data Parity is Applied

Access Source (Master)	Access Destination (Slave)
CPU1, CPU2, SubCPU, DMAC, DTS	INTC2, INTIF, EINT, EMUEINT, DMAC, DTS, CSIH, ADCC, RDC3A, Port group, DNF, RSENT
DMAC, DTS	Local RAM and global RAM of CPU1, CPU2, and SubCPU
CPU1, CPU2, SubCPU	Local RAM of other cores
SubCPU	Global RAM

When a data parity error occurs during read operation, error data is returned to each master. When a data parity error occurs during write operation, the write operation is cancelled if the slave is INTC2, INTIF, EINT, EMUEINT, ADC, RDC3A, DNF, or RSENT.

Some of the parity-applied modules have the control registers in the parity controllers (encoder and decoder), which can hold status upon detection of an error. If a parity error is detected in these modules, it is possible to identify the access that caused the error.

On the other hand, parity controllers which do not have control registers always perform parity detection. Though the status upon detection of an error is not retained in these controllers, whether an error has been detected or not is retained in the ECM.

29.2.9.1 List of Registers

Table 29.72 List of Registers

Module Name	Register Name	Symbol	Address
APDP	P-Bus data parity status register xx	APDPERRST_xx	<base_addr> + 0_H
APDP	P-Bus data parity status clear register xx	APDPERRSTC_xx	<base_addr> + 4_H
APDP	P-Bus data parity test mode control register xx	APDPTMC_xx	<base_addr> + 8_H
APDP	P-Bus data parity error address register xx	APDPERRADR_xx	<base_addr> + C_H

The suffix after each symbol in the above table indicates a parity-applied module for which the control register is provided. The table below shows the parity-applied modules and base addresses <base_addr>.

Table 29.73 List of Data Parity Control Modules

xx	Module to which Parity is Applied	Base Address <base_addr>
INTC2	INTC2	FFC6 8800$_H$
EINT	EINT	FFC8 6000$_H$
EMUEINT	EMUEINT	FFC8 6020$_H$
INTIF	INTIF	FFF9 9000$_H$
PDMA	DMA_DTS	FFC6 8900$_H$
SNT0	RSENT0	FFDC_2100$_H$
SNT1	RSENT1	FF7C_2120$_H$
SNT2	RSENT2	FFDC_2140$_H$
SNT3	RSENT3	FF7C_2160$_H$
CS0A	CSIH0 (group A)*1	FFF9 6000$_H$
CS0B	CSIH0 (group B)*1	FFF9 6020$_H$
CS1A	CSIH1 (group A)*1	FFF9 6040$_H$
CS1B	CSIH1 (group B)*1	FFF9 6060$_H$
CS2A	CSIH2 (group A)*1	FFF9 6080$_H$
CS2B	CSIH2 (group B)*1	FFF9 60A0$_H$
ADC0	ADCC0	FFC8 8000$_H$
ADC1	ADCC1	FF68 8020$_H$
ADC2	ADCC2	FFC8 8040$_H$
PT	Port group	FFC8 5000$_H$
DNF0	DNF0	FFC8 50C0$_H$
DNF1	DNF1	FFC8 50E0$_H$
DNF2	DNF2	FFC8 5100$_H$
DNF3	DNF3	FFC8 5120$_H$
DNF4	DNF4	FFC8 5140$_H$
DNF5	DNF5	FFC8 5160$_H$
DNF6	DNF6	FFC8 5180$_H$
DNF7	DNF7	FFC8 51A0$_H$
DNF8	DNF8	FFC8 51C0$_H$
RDC3A0	RDC3A0	FFFA 2000$_H$
RDC3A1	RDC3A1	FFFA 2020$_H$

Note 1. Each CSIHx register is divided into two groups and they are controlled separately.
Group A: CSIHnCTL0 to 2, CSIHnSTR0, and CSIHnSTCR0
Group B: Other than above.

29.2.9.2 Details of Registers

(1) APDPERRST_xx — P-bus data parity status register

For the suffix “xx”, see **Table 29.73 List of Data Parity Control Modules**.

Access: APDPERRST_xx is readable in 32-bit units.
APDPERRST_xxL is readable in 16-bit units.
APDPERRST_xxLL is readable in 8-bit units.

Address: <base_addr> + 00_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	APDPERR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.74 APDPERRST Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	The read value is 0. The write value should be 0.
0	APDPERR	Data Parity Error Monitor Flag Indicates occurrence of the parity error. APDPERR is cleared by setting the data parity error monitor flag clear bit (APDPERRC) to 1. 0: Indicates that a parity error has not occurred. 1: Indicates that a parity error has occurred.

(2) APDPERRSTC_xx — P-bus data parity status clear register

For the suffix “xx”, see **Table 29.73 List of Data Parity Control Modules**.

Access: APDPERRSTC_xx is writable in 32-bit units.
APDPERRSTC_xxL is writable in 16-bit units.
APDPERRSTC_xxLL is writable in 8-bit units.

Address: <base_addr> + 04_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	APDPERRC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 29.75 APDPERRSTC Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	The read value is 0. The write value should be 0.
0	APDPERRC	Clears the data parity error monitor flag (APDPERR). Writing 1 to this bit with the APDPERR bit set clears APDPERR. This bit is always read as 0.

(3) APDPTMC_xx — P-bus data parity test mode control register

Set the APDPTMC[1:0] bits to 01_B when writing to this register.

For the suffix "xx", see **Table 29.73 List of Data Parity Control Modules**.

Access: APDPTMC_xx is readable/writable in 32-bit units.
APDPTMC_xxL is readable/writable in 16-bit units.

Address: <base_addr> + 08_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APDPTMC[1:0]		—	—	—	—	—	—	—	—	—	—	APDPEIC3	APDPEIC2	APDPEIC1	APDPEIC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.76 APDPTMC Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15, 14	APDPTMC[1:0]	Data Parity Test Mode Control Setting APDPTMC[1:0] to 01_B enables write accesses to the error insertion control bits 3 to 0 (APDPEIC3 to APDPEIC0). When APDPTMC[1:0] are not 01_B, APDPEIC3 to APDPEIC0 are not written to even though a write access is attempted. These bits are always read as 0.
13 to 4	Reserved	The read value is 0. The write value should be 0.
3	APDPEIC3	Byte-Lane 3 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 3 (bits 31 to 24) to odd parity. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
2	APDPEIC2	Byte-Lane 2 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 2 (bits 23 to 16) to odd parity. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
1	APDPEIC1	Byte-Lane 1 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 1 (bits 15 to 8) to odd parity. APDPEIC1 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
0	APDPEIC0	Byte-Lane 0 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 0 (bits 7 to 0) to odd parity. APDPEIC0 can be written to when APDPTMC[1:0] = 01_B 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)

(4) APDPERRADR_xx — P-bus data parity error address register

For the suffix “xx”, see **Table 29.73 List of Data Parity Control Modules**.

Access: APDPERRADR_xx is readable in 32-bit units.

Address: <base_addr> + $0C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	APDPERRADR[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APDPERRADR[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.77 APDPERRADR Register Contents

Bit Position	Bit Name	Function
31 to 0	APDPERRADR [31:0]	Data Parity Error Address Holds the address at which the first parity error has occurred while the data parity error monitor flag (APDPAERR) is not set. The retained information is not updated even though the next parity error occurs while DPAERR is set.

29.3 Lockstep

This product incorporates CPU1 with the lockstep function to quickly detect CPU failures without special software. CPU1 executes the program using two different cores, that is, master core and checker core, and constantly compares the execution results of the two cores. When the results do not agree, the CPU1 determines that an error has occurred in one of the cores, and notifies the lock step error to the ECM.

Bus outputs to be compared are outputs to the local RAM for CPU1, global RAM, a CPU peripheral, interconnect, the P-bus, code flash memory, and the tag RAM and data RAM of the instruction cache.

The lockstep function of the CPU1 features failure insertion, with which errors can be intentionally caused and thus self-diagnosis of the lockstep operation is possible. Also, when a lockstep compare error occurred, the address and the data of the error are captured by a specified register.

29.3.1 List of Registers

Table 29.78 List of Registers

Module Name	Register Name	Symbol	Address
TESTCOMP	Comparator test register 0	TESTCOMPREG0	FFFE ED00$_H$
TESTCOMP	Comparator test register 1	TESTCOMPREG1	FFFE ED04$_H$
TESTCOMP	Lockstep error control register	LS_ERR_CNT	FFFE EE00$_H$
TESTCOMP	Lockstep error status register	LS_ERR_ST	FFFE EE04$_H$
TESTCOMP	Lockstep error address register	LS_ERR_ADDRESS	FFFE EE10$_H$
TESTCOMP	Lockstep error data register	LS_ERR_DATA	FFFE EE14$_H$

These registers are placed in the CPU Peripheral of CPU1. These registers can only be accessed by CPU1.

29.3.2 Details of Registers

29.3.2.1 TESTCOMPREG0 — Comparator test register 0

TESTCOMPREG0 is a test register 0 used for the lockstep function of CPU1.

In combination with TESTCOMPREG1, TESTCOMPREG0 enables self-diagnosis of the lockstep function. The following gives an example of self-diagnosis procedure.

(1) Write arbitrary value to TESTCOMPREG0.

(2) Write a different value to TESTCOMPREG1.

(3) Read TESTCOMPREG0. The different values are read and sent to the master core and checker core.

(4) Using the values read, run the comparator to be diagnosed.

Access: TESTCOMPREG0 is readable/writable in 32-bit units.
TESTCOMPREG0L and TESTCOMPREG0H are readable/writable in 16-bit units.
TESTCOMPREG0LL, TESTCOMPREG0LH, TESTCOMPREG0HL, and TESTCOMPREG0HH are readable/writable in 8-bit units.

Address: TESTCOMPREG0 : FFFE ED00$_H$,
TESTCOMPREG0L: FFFE ED00$_H$, TESTCOMPREG0H: FFFE ED02$_H$,
TESTCOMPREG0LL: FFFE ED00$_H$, TESTCOMPREG0LH: FFFE ED01$_H$,
TESTCOMPREG0HL: FFFE ED02$_H$, TESTCOMPREG0HH: FFFE ED03$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TESTCOMPREG0[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TESTCOMPREG0[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.79 TESTCOMPREG0 Register Contents

Bit Position	Bit Name	Function
31 to 0	TESTCOMPREG0[31:0]	Write: Data is written to each byte. Read: PE1: TESTCOMPREG0[31:0] value is read. PE1C: TESTCOMPREG1[31:0] value is read.

29.3.2.2 TESTCOMPREG1 — Comparator test register 1

TESTCOMPREG1 is a test register 1 used for the lockstep function of CPU1.

In combination with TESTCOMPREG0, TESTCOMPREG1 enables self-diagnosis of the lockstep function.

Access: TESTCOMPREG1 is readable/writable in 32-bit units.
TESTCOMPREG1L and TESTCOMPREG1H are readable/writable in 16-bit units.
TESTCOMPREG1LL, TESTCOMPREG1LH, TESTCOMPREG1HL, and TESTCOMPREG1HH are readable/writable in 8-bit units.

Address: TESTCOMPREG1 : FFFE ED04H,
TESTCOMPREG1L: FFFE ED04H, TESTCOMPREG1H: FFFE ED06H,
TESTCOMPREG1LL: FFFE ED04H, TESTCOMPREG1LH: FFFE ED05H,
TESTCOMPREG1HL: FFFE ED06H, TESTCOMPREG1HH: FFFE ED07H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TESTCOMPREG1[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TESTCOMPREG1[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.80 TESTCOMPREG1 Register Contents

Bit Position	Bit Name	Function
31 to 0	TESTCOMPREG1 [31:0]	Write: Data is written to each byte. Read: PE1: TESTCOMPREG1[31:0] value is read. PE1C: TESTCOMPREG0[31:0] value is read

29.3.2.3 LS_ERR_CNT — Lockstep error control register

LS_ERR_CNT enables or disables capturing of the address at which a lockstep error has occurred and the data which contains the error. Depending on the destination of access by the CPU, the error information is captured in the corresponding bit of this register.

Access: LS_ERR_CNT is readable/writable in 32-bit units.

Address: LS_ERR_CNT : FFFE EE00$_H$

Value after reset: 0000 000F$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	L1RAM	L2RAM	GAPB	LAPB
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.81 LS_ERR_CNT Register Contents

Bit Position	Bit Name	Function
31 to 4	Reserved	The read value is 0. The write value should be 0.
3	L1RAM	Enables or disables capturing of the address in the local RAM at which a lockstep error has occurred and the data which contains the error. 0: Capturing is disabled. 1: Capturing is enabled.
2	L2RAM	Enables or disables capturing of the address in the global RAM at which a lockstep error has occurred and the data which contains the error. 0: Capturing is disabled. 1: Capturing is enabled.
1	GAPB	Enables or disables capturing of the address at which a lockstep error has occurred and the data which contains the error in an access through P-bus*1. 0: Capturing is disabled. 1: Capturing is enabled.
0	LAPB	Enables or disables capturing of the address at which a lockstep error has occurred and the data which contains the error in an access to IPIR or MEV*1 0: Capturing is disabled. 1: Capturing is enabled.

Note 1. For access configuration of P-bus, IPIR, and MEV, see **Section 3, CPU System**.

29.3.2.4 LS_ERR_ST — Lockstep error status register

LS_ERR_ST monitors the first lockstep error. As long as the error flag for the first error remains uncleared, the error flags for the subsequent errors will not be set to 1. If two or more error sources are generated at the same time, the error flag will be set in the flags L1RAMW0 to 3, L2RAMP0 to 1, GAPB, and LAPB, in this order. These flags are cleared by an internal reset, an external reset, or writing 1 to this register.

Access: LS_ERR_ST is readable/writable in 32-bit units.

Address: LS_ERR_ST : FFFE EE04$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	L1RAMW0	L1RAMW1	L1RAMW2	L1RAMW3	L2RAMP0	L2RAMP1	GAPB	LAPB
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.82 LS_ERR_ST Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	The read value is 0. The write value should be 0.
7	L1RAMW0	Indicates whether or not the information (address and data) in word 0 of the local RAM has been captured as the source of the lockstep error. This flag is cleared by writing 1 to this bit. 0: No information has been captured. 1: Information has been captured.
6	L1RAMW1	Indicates whether or not the information (address and data) in word 1 of the local RAM has been captured as the source of the lockstep error. This flag is cleared by writing 1 to this bit. 0: No information has been captured. 1: Information has been captured.
5	L1RAMW2	Indicates whether or not the information (address and data) in word 2 of the local RAM has been captured as the source of the lockstep error. This flag is cleared by writing 1 to this bit. 0: No information has been captured. 1: Information has been captured.
4	L1RAMW3	Indicates whether or not the information (address and data) in word 3 of the local RAM has been captured as the source of the lockstep error. This flag is cleared by writing 1 to this bit. 0: No information has been captured. 1: Information has been captured.
3	L2RAMP0	Indicates whether or not the information (address and data) in the lower-order bits of the global RAM has been captured as the source of the lockstep error. This flag is cleared by writing 1 to this bit. 0: No information has been captured. 1: Information has been captured.
2	L2RAMP1	Indicates whether or not the information (address and data) in the higher-order bits of the global RAM has been captured as the source of a lockstep error. This flag is cleared by writing 1 to this bit. 0: No information has been captured. 1: Information has been captured.
1	GAPB	Indicates whether or not the information (address and data) has been captured as the source of a lockstep error in an access through P-bus*1. This flag is cleared by writing 1 to this bit. 0: No information has been captured. 1: Information has been captured.
0	LAPB	Indicates whether or not the information (address and data) has been captured as the source of a lockstep error in an access to IPIR or MEV*1. This flag is cleared by writing 1 to this bit. 0: No information has been captured. 1: Information has been captured.

Note 1. For access configuration of P-bus, IPIR, and MEV, see **Section 3, CPU System**.

29.3.2.5 LS_ERR_ADDRESS — Lockstep error address register

LS_ERR_ADDRESS holds the address at which an error occurred. The address of the error source corresponding to the error flag of LS_ERR_ST which has been set is held in this register.

Access: LS_ERR_ADDRESS is readable in 32-bit units.

Address: LS_ERR_ADDRESS : FFFE EE10H

Value after reset: — (retained)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Address[31:16]															
Value after reset:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Address[15:0]															
Value after reset:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.83 LS_ERR_ADDRESS Register Contents

Bit Position	Bit Name	Function
31 to 0	Address[31:0]	Monitors the address at which a lockstep error has occurred. The following shows the valid bits of each flag when the corresponding address is captured and how to change the address of each to real address. 0 is stored in invalid bits. L1RAMW0: Local RAM base_address + (Address [14:0]<<4) L1RAMW1: Local RAM base_address + (Address [14:0]<<4) + 0x4 L1RAMW2: Local RAM base_address + (Address [14:0]<<4) + 0x8 L1RAMW3: Local RAM base_address + (Address [14:0]<<4) + 0xC L2RAMP0: Global RAM_base_address + Address [20:0] L2RAMP1: Global RAM_base_address + Address [20:0] + 0x4 GAPB: Address [31:0] LAPB: 0xFF000000 + Address [23:0]

29.3.2.6 LS_ERR_DATA — Lockstep error data register

LS_ERR_DATA_L holds the data which contains an error. The data which contain the error from the error source corresponding to the error flag of LS_ERR_ST which has been set is held in this register.

Access: LS_ERR_DATA is readable in 32-bit units.

Address: LS_ERR_DATA : FFFE EE14H

Value after reset: — (retained)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA [31:16]															
Value after reset:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA [15:0]															
Value after reset:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.84 LS_ERR_DATA Register Contents

Bit Position	Bit Name	Function
31 to 0	DATA [31:0]	Monitors the data bits [31:0] which contains a lockstep error.

29.4 Memory Protection

29.4.1 Overview

This product incorporates the memory protection function to prevent erroneous accesses to data in memories and control registers of the peripheral circuits.

- MPU :

  CPU1, CPU2, and SubCPU protect memories against illegal accesses from themselves. Accesses to addresses that are prohibited by the MPU are never issued by the CPU1, CPU2, or SubCPU. For details, refer to the *H850 Family User's Manual: Software*.

- Slave Guard: A specific memory is protected against illegal accesses from any bus master. Slave guard includes the following guard types. The details of each type are given in the following sections.
  - PEG:

    The local RAM is protected against illegal accesses. However, accesses from the CPU incorporating the local RAM itself are excluded. For example, accesses from the CPU1 to local RAM in the CPU1 are not rejected by the PEG. For details, see **Section 3, CPU System**.
  - IPG (Internal Peripheral device Guard): The CPU Peripheral is protected against illegal accesses. For details, see **Section 3, CPU System**.
  - GRG : The global RAM is protected against illegal accesses.
  - PBG: The control registers in the peripheral circuits and memories are protected against illegal accesses. For details, see **Section 29.4.3, PBG**.

29.4.1.1 Identifiers for Slave Guard

For the slave guard function, the type of illegal accesses to be rejected can be designated using the following identifiers.

Table 29.85 Identifiers for Slave Guard

Identifier	Function
UM	When the CPU makes an access, the operating mode of the CPU is indicated. 0: Supervisor mode 1: User mode When the DMAC or DTS makes an access, the value of this identifier is the value in the channel master setting register. When the other master makes an access, the value of this identifier is always 0.
SPID	When the CPU makes an access, the system protection identifier SPID that is assigned to the CPU is indicated. When the DMAC or DTS makes an access, the value of this identifier is the value in the channel master setting register.When the other master makes an access, the value of this identifier is always 00_B.
PEID	The access source bus master is indicated. 000_B: Reserved 001_B: CPU1 010_B: CPU2 011_B: SubCPU 100_B: Reserved 101_B: Reserved 110_B: Reserved 111_B: DMAC/DTS

29.4.2 GRG (Global RAM Guard)

This product is provided with 8-channel GRG, which is described in detail in the following sections.

29.4.2.1 List of Registers

Table 29.86 List of Registers (1/2)

Module Name	Register Name	Symbol	Address
MGDGR	GRG control register (SIC)	MGDGRSCTL_SIC	FFC4 9100$_H$
MGDGR	GRG error status register (SIC)	MGDGRSSTAT_SIC	FFC4 9104$_H$
MGDGR	GRG error address register (SIC)	MGDGRSAD_SIC	FFC4 9108$_H$
MGDGR	GRG error access type register (SIC)	MGDGRSTYPE_SIC	FFC4 910C$_H$
MGDGR	GRG control register (PE1)	MGDGRSCTL_PE1	FFC4 9200$_H$
MGDGR	GRG error status register (PE1)	MGDGRSSTAT_PE1	FFC4 9204$_H$
MGDGR	GRG error address register (PE1)	MGDGRSAD_PE1	FFC4 9208$_H$
MGDGR	GRG error access type register (PE1)	MGDGRSTYPE_PE1	FFC4 920C$_H$
MGDGR	GRG control register (PE2)	MGDGRSCTL_PE2	FFC4 9300$_H$
MGDGR	GRG error status register (PE2)	MGDGRSSTAT_PE2	FFC4 9304$_H$
MGDGR	GRG error address register (PE2)	MGDGRSAD_PE2	FFC4 9308$_H$
MGDGR	GRG error access type register (PE2)	MGDGRSTYPE_PE2	FFC4 930C$_H$
MGDGR	GRG protection setting register 0	MGDGRPROT0	FFC4 9400$_H$
MGDGR	GRG protection SPID setting register 0	MGDGRSPID0	FFC4 9404$_H$
MGDGR	GRG compare base address register 0	MGDGRBAD0	FFC4 9408$_H$
MGDGR	GRG valid compare address register 0	MGDGRADV0	FFC4 940C$_H$
MGDGR	GRG protection setting register 1	MGDGRPROT1	FFC4 9410$_H$
MGDGR	GRG protection SPID setting register 1	MGDGRSPID1	FFC4 9414$_H$
MGDGR	GRG compare base address register 1	MGDGRBAD1	FFC4 9418$_H$
MGDGR	GRG valid compare address register 1	MGDGRADV1	FFC4 941C$_H$

Table 29.86 List of Registers (2/2)

Module Name	Register Name	Symbol	Address
MGDGR	GRG protection setting register 2	MGDGRPROT2	FFC4 9420$_H$
MGDGR	GRG protection SPID setting register 2	MGDGRSPID2	FFC4 9424$_H$
MGDGR	GRG compare base address register 2	MGDGRBAD2	FFC4 9428$_H$
MGDGR	GRG valid compare address register 2	MGDGRADV2	FFC4 942C$_H$
MGDGR	GRG protection setting register 3	MGDGRPROT3	FFC4 9430$_H$
MGDGR	GRG protection SPID setting register 3	MGDGRSPID3	FFC4 9434$_H$
MGDGR	GRG compare base address register 3	MGDGRBAD3	FFC4 9438$_H$
MGDGR	GRG valid compare address register 3	MGDGRADV3	FFC4 943C$_H$
MGDGR	GRG protection setting register 4	MGDGRPROT4	FFC4 9440$_H$
MGDGR	GRG protection SPID setting register 4	MGDGRSPID4	FFC4 9444$_H$
MGDGR	GRG compare base address register 4	MGDGRBAD4	FFC4 9448$_H$
MGDGR	GRG valid compare address register 4	MGDGRADV4	FFC4 944C$_H$
MGDGR	GRG protection setting register 5	MGDGRPROT5	FFC4 9450$_H$
MGDGR	GRG protection SPID setting register 5	MGDGRSPID5	FFC4 9454$_H$
MGDGR	GRG compare base address register 5	MGDGRBAD5	FFC4 9458$_H$
MGDGR	GRG valid compare address register 5	MGDGRADV5	FFC4 945C$_H$
MGDGR	GRG protection setting register 6	MGDGRPROT6	FFC4 9460$_H$
MGDGR	GRG protection SPID setting register 6	MGDGRSPID6	FFC4 9464$_H$
MGDGR	GRG compare base address register 6	MGDGRBAD6	FFC4 9468$_H$
MGDGR	GRG valid compare address register 6	MGDGRADV6	FFC4 946C$_H$
MGDGR	GRG protection setting register 7	MGDGRPROT7	FFC4 9470$_H$
MGDGR	GRG protection SPID setting register 7	MGDGRSPID7	FFC4 9474$_H$
MGDGR	GRG compare base address register 7	MGDGRBAD7	FFC4 9478$_H$
MGDGR	GRG valid compare address register 7	MGDGRADV7	FFC4 947C$_H$

- MGDGRPROTn, MGDGRSPIDn , MGDGRBADn, and MGDGRADVn set the protection specifications for each channel (n = 0 to 7).
- MGDGRSCTL_*, MGDGRSSTAT_*, MGDGRSAD_*, and MGDGRSTYPE_* indicate error information on each access port. "_SIC" represents access from the system interconnect to the global RAM, "_PE1" represents access from CPU1 to the global RAM, and "_PE2" represents access from CPU2 to the global RAM. Access from the SubCPU to the global RAM is made via the system interconnect.
- An access to the reserved areas shown below is notified as an error and the ERR bit of MGDGRSSTAT_* is set to 1. The concerned address is stored in the GRIFA bits of MGDGRSAD_*.

  C1M-A2: FEE0_0000$_H$ to FEEE_FFFF$_H$ and FEF1_0000$_H$ to FEFF_FFFF$_H$

  C1M-A1: FEE0_0000$_H$ to FEEE_FFFF$_H$ and FEF0_0000$_H$ to FEFF_FFFF$_H$

29.4.2.2 Details of Registers

(1) MGDGRSCTL_SIC/PE1/PE2 — GRG control register

Access: MGDGRSCTL_SIC, MGDGRSCTL_PE1, and MGDGRSCTL_PE2 are readable/writable in 32-bit units.
MGDGRSCTL_SICL, MGDGRSCTL_PE1L, and MGDGRSCTL_PE2L are readable/writable in 16-bit units.
MGDGRSCTL_SICLL, MGDGRSCTL_PE1LL, and MGDGRSCTL_PE2LL are readable/writable in 8-bit units.

Address: MGDGRSCTL_SIC : FFC4 9100$_H$, MGDGRSCTL_PE1 : FFC4 9200$_H$,
MGDGRSCTL_PE2 : FFC4 9300$_H$,
MGDGRSCTL_SICL : FFC4 9100$_H$, MGDGRSCTL_PE1L : FFC4 9200$_H$,
MGDGRSCTL_PE2L : FFC4 9300$_H$,
MGDGRSCTL_SICLL : FFC4 9100$_H$, MGDGRSCTL_PE1LL : FFC4 9200$_H$,
MGDGRSCTL_PE2LL : FFC4 9300$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERRCLO	ERRCLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.87 MGDGRSCTL_SIC/PE1/PE2

Bit Position	Bit Name	Function
31 to 2	Reserved	These bits are always read as 0. The write value should be 0.
1	ERRCLO	Error Entry Overflow Flag Clear 0: Not used. 1: Clears the overflow flag.
0	ERRCLE	Error Detection Flag Clear 0: Not used. 1: Clears the error detection flag. Set this bit simultaneously with ERRCLO as shown below.

Table 29.88 Combinations of Values in ERRCLO and ERRCLE

ERRCLO	ERRCLE	Function
0	0	No bit will be cleared.
0	1	No bit will be cleared. (This setting is ignored.)
1	0	The OVF bit is cleared.
1	1	Both OVF and ERR bits are cleared.

(2) MGDGRSSTAT_SIC/PE1/PE2 — GRG error status register

Access: MGDGRSSTAT_SIC, MGDGRSSTAT_PE1, and MGDGRSSTAT_PE2 are readable in 32-bit units.
MGDGRSSTAT_SICL, MGDGRSSTAT_PE1L, and MGDGRSSTAT_PE2L are readable in 16-bit units.
MGDGRSSTAT_SICLL, MGDGRSSTAT_PE1LL, and MGDGRSSTAT_PE2LL are readable in 8-bit units.

Address: MGDGRSSTAT_SIC : FFC4 9104H, MGDGRSSTAT_PE1 : FFC4 9204H,
MGDGRSSTAT_PE2 : FFC4 9304H,
MGDGRSSTAT_SICL : FFC4 9104H, MGDGRSSTAT_PE1L : FFC4 9204H,
MGDGRSSTAT_PE2L : FFC4 9304H,
MGDGRSSTAT_SICLL : FFC4 9104H, MGDGRSSTAT_PE1LL : FFC4 9204H,
MGDGRSSTAT_PE2LL : FFC4 9304H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OVF	ERR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.89 MGDGRSSTAT_SIC/PE1/PE2 Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	These bits are always read as 0. The write value should be 0.
1	OVF	Error Entry Overflow Flag 0: No overflow 1: Overflow occurred Since the error entry depth of GRG is 1, when another guard violation occurs after the error detection flag was set due to the earlier guard violation, the error entry overflow occurs and this flag is set. Though occurrence of overflow is reported to ECM, guard violation error information is not captured in case of an overflow.
0	ERR	Error Detection Flag 0: No error 1: An error occurred

CAUTION

- An access to the reserved areas shown below is detected as an error and the ERR bit is set to 1.
  C1M-A2: FEE0_0000H to FEEE_FFFFH and FEF1_0000H to FEFF_FFFFH
  C1M-A1: FEE0_0000H to FEEE_FFFFH and FEF0_0000H to FEFF_FFFFH

(3) MGDGRSAD_SIC/PE1/PE2 — GRG error address register

Access: MGDGRSAD_SIC, MGDGRSAD_PE1, and MGDGRSAD_PE2 are readable in 32-bit units.
MGDGRSAD_SICL, MGDGRSAD_PE1L, and MGDGRSAD_PE2L are readable in 16-bit units.
MGDGRSAD_SICLL, MGDGRSAD_PE1LL, and MGDGRSAD_PE2LL are readable in 8-bit units.

Address: MGDGRSAD_SIC : FFC4 9108H, MGDGRSAD_PE1 : FFC4 9208H,
MGDGRSAD_PE2 : FFC4 9308H,
MGDGRSAD_SICL : FFC4 9108H, MGDGRSAD_PE1L : FFC4 9208H,
MGDGRSAD_PE2L : FFC4 9308H,
MGDGRSAD_SICLL : FFC4 9108H, MGDGRSAD_PE1LL : FFC4 9208H,
MGDGRSAD_PE2LL : FFC4 9308H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	GRIFA[20:16]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRIFA[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.90 MGDGRSAD_SIC/PE1/PE2 Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	These bits are always read as 0. The write value should be 0.
20 to 0	GRIFA[20:0]	These bits hold the address at which an error has occurred.

CAUTION

- An access to the reserved areas shown below is detected as an error and the concerned address is stored in GRIFA.
  C1M-A2: FEE0_0000H to FEEE_FFFFH and FEF1_0000H to FEFF_FFFFH
  C1M-A1: FEE0_0000H to FEEE_FFFFH and FEF0_0000H to FEFF_FFFFH

(4) MGDGRSTYPE_SIC/PE1/PE2 — GRG error access type register

Access: MGDGRSTYPE_SIC, MGDGRSTYPE_PE1, and MGDGRSTYPE_PE2 are readable in 32-bit units.
MGDGRSTYPE_SICL, MGDGRSTYPE_PE1L, and MGDGRSTYPE_PE2L are readable in 16-bit units.
MGDGRSTYPE_SICLL, MGDGRSTYPE_SICLH, MGDGRSTYPE_PE1LL, MGDGRSTYPE_PE1LH , MGDGRSTYPE_PE2LL, and MGDGRSTYPE_PE2LH are readable in 8-bit units.

Address: MGDGRSTYPE_SIC : FFC4 910C$_H$, MGDGRSTYPE_PE1 : FFC4 920C$_H$,
MGDGRSTYPE_PE2 : FFC4 930C$_H$,
MGDGRSTYPE_SICL : FFC4 910C$_H$, MGDGRSTYPE_PE1L : FFC4 920C$_H$,
MGDGRSTYPE_PE2L : FFC4 930C$_H$,
MGDGRSTYPE_SICLL : FFC4 910C$_H$, MGDGRSTYPE_SICLH : FFC4 910D$_H$,
MGDGRSTYPE_PE1LL : FFC4 920C$_H$, MGDGRSTYPE_PE1LH : FFC4 920D$_H$,
MGDGRSTYPE_PE2LL : FFC4 930C$_H$, MGDGRSTYPE_PE2LH : FFC4 930D$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEID[2:0]			—			SPID[1:0]		—	UM	—	TYPE[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.91 MGDGRSTYPE_SIC/PE1/PE2 Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	These bits are always read as 0. The write value should be 0.
15 to 13	PEID[2:0]	PEID of the error source.
12 to 10	Reserved	The read value is 0. The write value should be 0.
9, 8	SPID[1:0]	SPID of the error source.
7	Reserved	The read value is 0. The write value should be 0.
6	UM	UM of the error source.
5	Reserved	These bits are always read as 0. The write value should be 0.
4 to 0	TYPE[4:0]	(Reference Information) Transfer type at the time of error occurrence These bits hold a signal used for an internal bus or interconnect.

(5) MGDGRPROTn — GRG protection setting register n (n = 0 to 7)

Access: MGDGRPROT0, MGDGRPROT1, MGDGRPROT2, MGDGRPROT3, MGDGRPROT4, MGDGRPROT5, MGDGRPROT6, and MGDGRPROT7 are readable/writable in 32-bit units.
MGDGRPROT0H, MGDGRPROT1H, MGDGRPROT2H, MGDGRPROT3H,
MGDGRPROT4H, MGDGRPROT5H, MGDGRPROT6H, and MGDGRPROT7H are readable/writable in 16-bit units.
MGDGRPROT0HL, MGDGRPROT0HH, MGDGRPROT1HL, MGDGRPROT1HH,
MGDGRPROT2HL, MGDGRPROT2HH, MGDGRPROT3HL, MGDGRPROT3HH,
MGDGRPROT4HL, MGDGRPROT4HH, MGDGRPROT5HL, MGDGRPROT5HH,
MGDGRPROT6HL, MGDGRPROT6HH, MGDGRPROT7HL, and MGDGRPROT7HH are readable/writable in 8-bit units.

Address: MGDGRPROT0 : FFC4 9400H, MGDGRPROT1 : FFC4 9410H, MGDGRPROT2 : FFC4 9420H,
MGDGRPROT3 : FFC4 9430H, MGDGRPROT4 : FFC4 9440H, MGDGRPROT5 : FFC4 9450H,
MGDGRPROT6 : FFC4 9460H, MGDGRPROT7 : FFC4 9470H,
MGDGRPROT0H : FFC4 9402H, MGDGRPROT1H : FFC4 9412H, MGDGRPROT2H : FFC4 9422H,
MGDGRPROT3H : FFC4 9432H, MGDGRPROT4H : FFC4 9442H, MGDGRPROT5H : FFC4 9452H,
MGDGRPROT6H : FFC4 9462H, MGDGRPROT7H : FFC4 9472H,
MGDGRPROT0HL : FFC4 9402H, MGDGRPROT0HH : FFC4 9403H,
MGDGRPROT1HL : FFC4 9412H, MGDGRPROT1HH : FFC4 9413H,
MGDGRPROT2HL : FFC4 9422H, MGDGRPROT2HH : FFC4 9423H,
MGDGRPROT3HL : FFC4 9432H, MGDGRPROT3HH : FFC4 9433H,
MGDGRPROT4HL : FFC4 9442H, MGDGRPROT4HH : FFC4 9443H,
MGDGRPROT5HL : FFC4 9452H, MGDGRPROT5HH : FFC4 9453H,
MGDGRPROT6HL : FFC4 9462H, MGDGRPROT6HH : FFC4 9463H,
MGDGRPROT7HL : FFC4 9472H, MGDGRPROT7HH : FFC4 9473H

Value after reset: 07FF 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	EN	—			—	UM	DEB	PEID[7:0]							
Value after reset:	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.92 MGDGRPROTn Register Contents

Bit Position	Bit Name	Function
31	Reserved	These bits are always read as 0. The write value should be 0.
30	EN	Protection Enable 0: Disables protection function. 1: Enables protection function. The setting of this bit covers only the types of access allowed by this register.
29 to 27	Reserved	These bits are always read as 0. The write value should be 0.
26	Reserved	This bit is always read as 1. The write value should be 1.
25	UM	User Mode Access 0: Enables access in supervisor mode. 1: Enables access in user mode and supervisor mode.
24	DEB	Debug Access 0: Disables access from the debug master. 1: Enables access from the debug master. Be sure to set this bit to 1. If the setting is 0, the debugger or RAM monitor tool may not operate correctly.
23 to 16	PEID[7:0]	PEID Access The PEID field is a bit list, in which one bit corresponds to one PEID value. Setting multiple bits enables multiple ID values simultaneously. For example, setting the PEID field to 0101_B enables access with PEID = 0 and PEID = 2. 0: Disables access with PEID = n. 1: Enables access with PEID = n.
15 to 0	Reserved	These bits are always read as 0. The write value should be 0.

(6) MGDGRSPIDn — GRG protection SPID setting register n (n = 0 to 7)

Access: MGDGRSPIDn is readable/writable in 32-bit units.
MGDGRSPIDnL is readable/writable in 16-bit units.
MGDGRSPIDnLL is readable/writable in 8-bit units.

Address: MGDGRSPID0 : FFC4 9404$_H$, MGDGRSPID1 : FFC4 9414$_H$, MGDGRSPID2 : FFC4 9424$_H$,
MGDGRSPID3 : FFC4 9434$_H$, MGDGRSPID4 : FFC4 9444$_H$, MGDGRSPID5 : FFC4 9454$_H$,
MGDGRSPID6 : FFC4 9464$_H$, MGDGRSPID7 : FFC4 9474$_H$,
MGDGRSPID0L : FFC4 9404$_H$, MGDGRSPID1L : FFC4 9414$_H$, MGDGRSPID2L : FFC4 9424$_H$,
MGDGRSPID3L : FFC4 9434$_H$, MGDGRSPID4L : FFC4 9444$_H$, MGDGRSPID5L : FFC4 9454$_H$,
MGDGRSPID6L : FFC4 9464$_H$, MGDGRSPID7L : FFC4 9474$_H$,
MGDGRSPID0LL : FFC4 9404$_H$, MGDGRSPID1LL : FFC4 9414$_H$, MGDGRSPID2LL : FFC4 9424$_H$,
MGDGRSPID3LL : FFC4 9434$_H$, MGDGRSPID4LL : FFC4 9444$_H$, MGDGRSPID5LL : FFC4 9454$_H$,
MGDGRSPID6LL : FFC4 9464$_H$, MGDGRSPID7LL : FFC4 9474$_H$

Value after reset: FFFF FFFF$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	SPID [3:0]			
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.93 MGDGRSPIDn Register Contents

Bit Position	Bit Name	Function
31 to 4	Reserved	These bits are read as 1. The write value should be 1.
3 to 0	SPID[3:0]	SPID Access The SPID field is a bit list, in which one bit corresponds to one SPID value. Setting multiple bits enables the multiple SPID values simultaneously. For example, setting the SPID field to 0101B enables access with SPID = 0 and SPID = 2. 0: Disables access with SPID = n. 1: Enables access with SPID = n.

(7) MGDGRBADn — GRG compare base address register n (n = 0 to 7)

Access: MGDGRBADn is readable/writable in 32-bit units.
MGDGRBADn (L/H) is readable/writable in 16-bit units.
MGDGRBADn (LH/HL) is readable/writable in 8-bit units.

Address: MGDGRBAD0 : FFC4 9408H, MGDGRBAD1 : FFC4 9418H, MGDGRBAD2 : FFC4 9428H,
MGDGRBAD3 : FFC4 9438H, MGDGRBAD4 : FFC4 9448H, MGDGRBAD5 : FFC4 9458H,
MGDGRBAD6 : FFC4 9468H, MGDGRBAD7 : FFC4 9478H,
MGDGRBADnL : MGDGRBADn + 00H,
MGDGRBADnH : MGDGRBADn + 02H,
MGDGRBADnLH : MGDGRBADn + 01H,
MGDGRBADnHL : MGDGRBADn + 02H,

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	AD[20:16]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD[15:9]							—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R

Table 29.94 MGDGRBAD n Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	These bits are always read as 0. The write value should be 0.
20 to 9	AD[20:9]	Compare base address
8 to 0	Reserved	These bits are always read as 0. The write value should be 0.

(8) MGDGRADVn — GRG valid compare address register n (n = 0 to 7)

Access: MGDGRADVn is readable/writable in 32-bit units.
MGDGRADVn (L/H) is readable/writable in 16-bit units.
MGDGRADVn (LH/HL) is readable/writable in 8-bit units.

Address: MGDGRADV0 : FFC4 940C$_H$, MGDGRADV1 : FFC4 941C$_H$, MGDGRADV2 : FFC4 942C$_H$,
MGDGRADV3 : FFC4 943C$_H$, MGDGRADV4 : FFC4 944C$_H$, MGDGRADV5 : FFC4 945C$_H$,
MGDGRADV6 : FFC4 946C$_H$, MGDGRADV7 : FFC4 947C$_H$,
MGDGRADVnL : MGDGRADVn + 00$_H$,
MGDGRADVnH : MGDGRADVn + 02$_H$,
MGDGRADVnLH : MGDGRADVn + 01$_H$,
MGDGRADVnHL : MGDGRADVn + 02$_H$,

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	ADV[20:16]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADV[15:9]							—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R

Table 29.95 MGDGRADV n Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	These bits are always read as 0. The write value should be 0.
20 to 9	ADV[20:9]	Bits of addresses corresponding to bits from among MGDGRADVn[20:9] which have the value 1 are compared. When MGDGRADVn[20:9] are all 1, the 512 bytes from the address specified by MGDGRBADn are protected. When MGDGRADVn[20:9] are all 0, the entire global RAM is subject to protection.
8 to 0	Reserved	These bits are always read as 0. The write value should be 0.

Setting example: When MGDGRBADn[20:9] = 800$_H$ and MGDGRADVn[20:9] = FF7$_H$, the global RAM guard protection areas n are FEF0 0000$_H$ to FEF0 01FF$_H$ and FEF0 1000$_H$ to FEF0 11FF$_H$.

Concept: When MGDGRBADn[20:9] = 800$_H$, the base address is FEF0 0000$_H$ and the base address of the range to which the setting applies is indicated within [ ] below.

1111	1110	111 [1	0000	0000	000] 0	0000	0000
F	E	F	0	0	0	0	0

When MGDGRADVn[20:9] = $FF7_H$, the setting of the lower bits 9-0 is 0 so the corresponding bit of addresses is not compared, and the 512 bytes represented by

1111 1110 111 [1 0000 000x 000] x xxxx xxxx,

i.e.,

F E F 0 0 0 0 0 to

F E F 0 0 1 F F,

and

F E F 0 1 0 0 0 to

F E F 0 1 1 F F

(1 Kbyte in total) are protected.

29.4.3 PBG

The PBG module is divided into several PBG groups, each of which is provided with a maximum of 16 protection channels. A single PBG channel can designate the access against which a single peripheral circuit should be protected. Each PBG group can hold the information of the access that has been rejected.

The following table lists the peripheral circuits to be protected, the corresponding PBG group names, and the PBG channel numbers.

Table 29.96 Peripheral Modules to be Protected and the Corresponding PBG Channel Numbers (1/4)

PBG Group	PBG Channel Number	Module to be Protected	Remark
PBG0	0	INTC2	
	1	DMA_DTS	
PBG1	4	GRG (control register)	
	5	GRG status (SIC)	
	6	GRG status (PE1)	
	7	GRG status (PE2)	
PBG2	0	PBG2 itself	
	1	SCI30	
	2	SCI32	
	3	OSTM0	
	4	OSTM2	
	5	WDTA0	
	6	SENT0	
	7	Data parity (SENT0)	
	8	SENT2	
	9	Data parity (SENT2)	
PBG3	0	PBG3 itself	
	1	SCI31	
	2	OSTM1	
	3	OSTM3	
	4	WDTA1	
	5	SENT1	
	6	Data parity (SENT1)	
	7	SENT3	
	8	Data parity (SENT3)	

Table 29.96 Peripheral Modules to be Protected and the Corresponding PBG Channel Numbers (2/4)

PBG Group	PBG Channel Number	Module to be Protected	Remark
PBG4	0	PBG4 itself	
	1	PIC1B0	
	2	PIC2D	
	3	TAUD0	
	4	TAUD2	
	5	TAUD3	
	6	TAUJ0	
	7	TSG30	
	8	TSG32	
	9	ENCA0	
	10	TAPA0	
	11	TAPA2	
	12	TAPA3	
	13	TAPA5	
	14	TPBA0	
PBG5	0	PBG5 itself	
	1	PIC1B1	
	2	TAUD1	
	3	TAUJ1	
	4	TSG31	
	5	ENCA1	
	6	TAPA1	
	7	TAPA4	
	8	TPBA1	
PBG6	0	PBG6 itself	
	1	RS-CANFD	
	2	ECC (RS-CANFD MB)	
	3	ECC (RS-CANFD AFL)	
	4	DTSTRGSEL	
	5	DMATRGSEL	
	6	INTIF	
	7	Data parity (INTIF)	
	8	RDC3A0	
	9	Data parity (RDC3A0)	
	10	RDC3A1	
	11	DataParity (RDC3A1)	
PBG7	0	PBG7 itself	
	1	CSIH0 (group A)*1	
	2	CSIH0 (group B)*1	
	3	DataParity (CSIH0 group A)	
	4	DataParity (CSIH0 group B)	
	5	ECC (CSIH0)	
	6	CSIH1 (group A)*1	

Table 29.96 Peripheral Modules to be Protected and the Corresponding PBG Channel Numbers (3/4)

PBG Group	PBG Channel Number	Module to be Protected	Remark
PBG7	7	CSIH1 (group B)*1	
	8	Data parity (CSIH1 group A)	
	9	Data parity (CSIH1 group B)	
	10	ECC (CSIH1)	
	11	CSIH2 (group A)*1	
	12	CSIH2 (group B)*1	
	13	Data parity (CSIH2 group A)	
	14	Data parity (CSIH2 group B)	
	15	ECC (CSIH2)	
PBG8	0	PBG8 itself	
	1	EMU3 ch0*2	
	2	EMU3 ch1*2	
	3	EMU3 common register*2	
	4	MISR (SubCPU)	
	5	ECC (EMU3 instruction cache data)	
	6	ECC (EMU3 instruction cache tag)	
	7	ECC (EMU3 code flash)	
	8	ECC (EMU3 local RAM)	
PBG9	0	PBG9 itself	
	1	RLIN30	
	2	RLIN32	
	3	ADCC0	
	4	Data parity (ADCC0)	
	5	ADCC2	
	6	Data parity (ADCC2)	
	7	ADPA	
	8	DCRA0	
	9	ECM (master register)	
	10	ECM (checker register)	
	11	ECM (common register)	
	12	EINT	
	13	Data parity (EINT)	
	14	EMUEINT	
	15	Data parity (EMUEINT)	
PBG10	0	PBG10 itself	
	2	FLSCI	
	4	Port group	
	5	DataParity (port group)	
	6	DNF0	

Table 29.96 Peripheral Modules to be Protected and the Corresponding PBG Channel Numbers (4/4)

PBG Group	PBG Channel Number	Module to be Protected	Remark
PBG10	7	Data parity (DNF0)	
	8	DNF1	
	9	Data parity (DNF1)	
	10	DNF2	
	11	Data parity (DNF2)	
	12	DNF3	
	13	Data parity (DNF3)	
	14	DNF4	
	15	Data parity (DNF4)	
PBG11	0	PBG11 itself	
	1	DNF5	
	2	Data parity (DNF5)	
	3	DNF6	
	4	Data parity (DNF6)	
	5	DNF7	
	6	Data parity (DNF7)	
	7	DNF8	
	8	Data parity (DNF8)	
PBG12	0	PBG12 itself	
	1	RLIN31	
	2	ADCC1	
	3	Data parity (ADCC1)	
	4	DCRA1	
PBG13	0	PBG13 itself	
	1	System control (group A)*3	
	2	System control (group B)*3	

Note 1. The CSIHx registers are divided into the following two groups and controlled separately.
Group A: CSIHnCTL0 to CSIHnCTL2, CSIHnSTR0, and CSIHnSTCR0
Group B: Other than above

Note 2. The registers of the EMU3 are not protected against accesses made by the SubCPU because they are through the dedicated bus. For protection, use the MPU. For details on the MPU, see **Section 3.2.1, Core Functions**.

Note 3. The system control registers are divided into the following two groups and controlled separately.
Group A: RESF, RESFC, PLL0CLKS, PLL0CLKC1, CKSC0CTL, CKSC0ACT, CLKD0DIV, CLKD0STAT, CKSC1CTL, CKSC1ACT, CLMAnCTL0 (n=0 to 3), CLMAnCMPL (n=0 to 3), CLMAnCMPH (n=0 to 3), CLMAnPCMD (n=0 to 3), CLMAnPS (n=0 to 3), CLMATEST, CLMATESTS, FHVE3, and FHVE15
Group B: SWRESA, PROT1PHCMD, and PROT1PS

29.4.3.1 List of Registers

The following table lists the register provided for each PBG channel.

Table 29.97 List of Register Provided for Each PBG Channel

Module Name	Register Name	Symbol	Address
PBG	PBGxx protection register n	FSGDxxDPROTn	<base_addr0> + 4*n

The following table lists the registers provided for each PBG group.

Table 29.98 List of Registers Provided for Each PBG Group

Module Name	Register Name	Symbol	Address
PBG	PBGxx error control register	ERRSLVxxCTL	<base_addr1> + 0_H
PBG	PBGxx error status register	ERRSLVxxSTAT	<base_addr1> + 4_H
PBG	PBGxx error address register	ERRSLVxxADDR	<base_addr1> + 8_H
PBG	PBGxx error type register	ERRSLVxxTYPE	<base_addr1> + C_H

In the above tables, “xx” and “n” in the register names and symbols represents the PBG group numbers and PBG channel numbers, respectively. The table below shows the base address values <base_addr0> and <base_addr1>, which correspond to each of the PBG group numbers and PBG channel numbers.

Table 29.99 Base Addresses

PBG Group	PBG Group xx	PBG Channel Number n	<base_addr0>	<base_addr1>
PBG0	0	0, 1	FFC4 $C000_H$	FFC4 $C800_H$
PBG1	1	4 to 7	FFC4 $C100_H$	FFC4 $C900_H$
PBG2	2	0 to 9	FFDC 0000_H	FFDC 0200_H
PBG3	3	0 to 8	FF7C 0800_H	FF7C $0A00_H$
PBG4	4	0 to 14	FFDD $D000_H$	FFDD $D200_H$
PBG5	5	0 to 8	FF7D $D800_H$	FF7D $DA00_H$
PBG6	6	0 to 11	FFF9 4000_H	FFF9 4200_H
PBG7	7	0 to 15	FFF9 4400_H	FFF9 4600_H
PBG8	8	0 to 8	FFF9 4800_H	FFF9 $4A00_H$
PBG9	9	0 to 15	FFC4 0000_H	FFC4 0200_H
PBG10	10	0, 2, 4 to 15	FFC4 0400_H	FFC4 0600_H
PBG11	11	0 to 8	FFC4 0800_H	FFC4 $0A00_H$
PBG12	12	0 to 4	FF67 8000_H	FF67 8200_H
PBG13	13	0 to 2	FFF9 0000_H	FFF9 0200_H

29.4.3.2 Details of Registers

(1) FSGDxxDPROTn — PBGxx protection register n

FSGDxxDPROTn designates the access to be rejected against which the peripheral circuit control registers and RAM should be protected. Any access that is disabled using any of the identifiers is rejected as an illegal access.

Access: FSGDxxDPROTn is readable/writable in 32-bit units.
FSGDxxDPROTn (L/H) is readable/writable in 16-bit units.
FSGDxxDPROTn (LL/LH/HL/HH) is readable/writable in 8-bit units.

Address: FSGDxxDPROTn : <base_addr0> + 4 × n
FSGDxxDPROTnL : FSGDxxDPROTn + 00_H
FSGDxxDPROTnH : FSGDxxDPROTn + 02_H
FSGDxxDPROTnLL : FSGDxxDPROTn + 00_H
FSGDxxDPROTnLH : FSGDxxDPROTn + 01_H
FSGDxxDPROTnHL : FSGDxxDPROTn + 02_H
FSGDxxDPROTnHH : FSGDxxDPROTn + 03_H

Value after reset: $07FF\ FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PROTUM	PROTPEID[7:0]								—
Value after reset:	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PROTSPID[3:0]				PROTDEB	PROTRDPDEF	PROTWRPDEF	PROTRD	PROTWR
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.100 FSGDxxDPROTn Register Contents

Bit Position	Bit Name	Function
31 to 27	Reserved	The read value is 0. The write value should be 0.
26	Reserved	This bit is read as 1. The write value should be 1.
25	PROTUM	User Mode Access 0: Enables access in supervisor mode. 1: Enables access in user mode and supervisor mode.
24 to 17	PROTPEID[7:0]	PEID Access The PROTnPEID field is a bit list, in which one bit corresponds to one PEID value. Setting multiple bits enables multiple ID values simultaneously. For example, setting the PROTnPEID field to 0101_B enables access with PEID = 0 and PEID = 2. 0: Disables access with PEID = n. 1: Enables access with PEID = n.
16 to 9	Reserved	These bits are read as 1. The write value should be 1.
8 to 5	PROTSPID[3:0]	SPID Access The PROTnSPID field is a bit list, in which one bit corresponds to one SPID value. Setting multiple bits enables the multiple SPID values simultaneously. For example, setting the PROTnSPID field to 0101_B enables access with SPID = 0 and SPID = 2. 0: Disables access with SPID = n. 1: Enables access with SPID = n.
4	PROTDEB	Debug Access 0: Disables access from the debug master. 1: Enables access from the debug master. Always set 1 to this bit. Setting 0 may lead to incorrect operations of the debugger or the RAM monitoring tool.
3	PROTRDPDEF	Default Read Protection 0: Enables read access from any master. 1: Only enables reading by the access-permitted master.
2	PROTWRPDEF	Default Write Protection 0: Enables write access from any master. 1: Only enables write access from the access-permitted master.
1	PROTRD	Read Permission 0: Reading by any master is prohibited. 1: Reading by the access-permitted master only is permitted. Note, however, if both PROTRDPDEF and PROTRD are set to 0, the setting of PROTRDPDEF takes priority, that is, read access from any master is enabled.
0	PROTWR	Default Write Protection 0: Enables write access from any master. 1: Only enables write access from the access-permitted master. Note however, if both PROTWRPDEF and PROTWR are set to 0, the setting of PROTWRPDEF takes priority, that is, write access from any master is enabled.

(2) ERRSLVxxCTL — PBGxx error control register

ERRSLVxxCTL clears the status in the error status registers of PBGxx.

Access: ERRSLVxxCTL is writable in 32-bit units.
ERRSLVxxCTL (L/H) is writable in 16-bit units.
ERRSLVxxCTL (LL/HH/HL/HH) is writable in 8-bit units.

Address: ERRSLVxxCTL : <base_addr1>,
ERRSLVxxCTLL : ERRSLVxxCTL + 00_H,
ERRSLVxxCTLH : ERRSLVxxCTL + 02_H,
ERRSLVxxCTLLL : ERRSLVxxCTL + 00_H,
ERRSLVxxCTLLH : ERRSLVxxCTL + 01_H,
ERRSLVxxCTLHL : ERRSLVxxCTL + 02_H,
ERRSLVxxCTLHH : ERRSLVxxCTL + 03_H,

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLRO	CLRE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 29.101 ERRSLVxxCTL Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	CLRO	Error Entry Overflow Flag Clear 0: No operation. 1: Clears the overflow flag.
0	CLRE	Error Detection Flag Clear 0: No operation. 1: Clears the error detection flag.

(3) ERRSLVxxSTAT — PBGxx error status register

ERRSLVxxSTAT holds the status of the illegal access detected by PBGxx.

Access: ERRSLVxxSTAT is readable in 32-bit units.
ERRSLVxxSTAT (L/H) is readable in 16-bit units.
ERRSLVxxSTAT (LL/LH/HL/HH) is readable in 8-bit units.

Address: ERRSLVxxSTAT : <base_addr1> + 04_H,
ERRSLVxxSTATL : ERRSLVxxSTAT + 00_H,
ERRSLVxxSTATH : ERRSLVxxSTAT + 02_H,
ERRSLVxxSTATLL : ERRSLVxxSTAT + 00_H,
ERRSLVxxSTATLH : ERRSLVxxSTAT + 01_H,
ERRSLVxxSTATHL : ERRSLVxxSTAT + 02_H,
ERRSLVxxSTATHH : ERRSLVxxSTAT + 03_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OVF	ERR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.102 ERRSLVxxSTAT Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	The read value is 0. The write value should be 0.
1	OVF	Error Entry Overflow Flag 0: No overflow 1: Overflow occurred Since the error entry depth of PBG is 1, when another guard violation occurs after the error detection flag was set due to the earlier guard violation, the error entry overflow occurs and this flag is set. Occurrence of overflow is not reported to ECM. Guard violation error information is not captured in case of an overflow.
0	ERR	Error Detection Flag 0: No error 1: An error occurred

(4) ERRSLVxxADDR — PBGxx error address register

ERRSLVxxADDR holds the address of the illegal access detected by PBGxx.

Access: ERRSLVxxADDR is readable in 32-bit units.

Address: ERRSLVxxADDR: <base_addr1> + 08_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADDR[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADDR[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.103 ERRSLVxxADDR Register Contents

Bit Position	Bit Name	Function
31 to 0	ADDR[31:0]	Address at which an error has occurred. ADDR[1:0] are fixed to 0.

(5) ERRSLVxxTYPE — PBGxx error type register

ERRSLVxxTYPE holds the type of illegal access detected by PBGxx.

Access: ERRSLVxxTYPE is readable in 32-bit units.
ERRSLVxxTYPE (L/H) is readable in 16-bit units.

Address: ERRSLVxxTYPE : <base_addr1> + $0C_H$,
ERRSLVxxTYPEL : ERRSLVxxTYPE + 00_H ,
ERRSLVxxTYPEH : ERRSLVxxTYPE + 02_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEID[2:0]			—	—	—	SPID[1:0]		—	UM	—	STRB[3:0]				WRITE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.104 ERRSLVxxTYPE Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	The read value is 0. The write value should be 0.
15 to 13	PEID[2:0]	PEID of the error source.
12 to 10	Reserved	The read value is 0. The write value should be 0.
9, 8	SPID[1:0]	SPID of the error source.
7	Reserved	The read value is 0. The write value should be 0.
6	UM	UM of the error source.
5	Reserved	The read value is 0. The write value should be 0.
4 to 1	STRB[3:0]	(Reference Information) The strobe signal at the time of error occurrence. These bits hold a signal used for an internal bus or interconnect.
0	WRITE	(Reference Information) The write signal at the time of error occurrence. This bit holds a signal used for an internal bus or interconnect.

29.5 Multi-Input Signature Generator (MISG)

29.5.1 Overview

This LSI incorporates multi-input signature generators (MISG) for self-diagnosis by the CPUs. The table below shows the overview of the MISG specifications.

Table 29.105 MISG Overview

Item	Description
Generating polynomials	The polynomials are available for use in signature generation. • MISR1: G (x) = x32 +x26 +x23 +x22 +x16 +x12 +x11 +x10 +x8 +x7 +x5 +x4 +x2 +x +1 • MISR2: G (x) = x32 +x22 +x2 +x +1 To support signature generation for 64-bit data, each CPU is connected with an MISG that has two MISRs (MISR1 and MISR2).
Signature generation	Signature generation can be enabled or disabled. • Signature generation in MISR1 is enabled or disabled. • Signature generation in MISR2 is enabled or disabled. The following two conditions can be selected as conditions for signature generation • Writing to the register A signature is generated by writing to the MISR1 calculation register (MISRCDR_*) • Monitoring write access A signature is generated if writing to the address area specified for monitoring occurs when writing by the CPU is being monitored. The address area is specified by the monitoring area base address register and the monitoring area address mask register.
Automatic signature comparison	Two signature generation units are selected for comparison of signatures. Each signature generation unit has a data counter and comparison proceeds when the values of the data counters in the MISGs selected as the target for comparison match. The data counter counts the number of write accesses to the MISRCDR_* register or the address area being monitored.
Error notification	When signatures are compared and do not match, the ECM is notified of an error. Enabling or disabling of error notification to the ECM can be selected. An interrupt request for the INTC is not made directly.

Throughout this section, an asterisk (*) in MISRCDR_*, MISR1_*, and MISR2_* respectively refers to PE1, PE2, and PE3.

29.5.2 Block Diagram

29.5.2.1 MISG

The figure below is a block diagram of the MISG. The MISG consists of three signature generation units and a signature comparison unit.

Write monitoring mode for signature generation conditions (refer to **Section 29.5.3.1(2)**) can only be executed between the corresponding signature generation unit and the CPU. Therefore, each signature generation unit is given a name corresponding to the CPU number (MISG_PE1, MISG_PE2, or MISG_PE3). MISG_PE1 can monitor CPU1's write accesses, MISG_PE2 can monitor CPU2's write accesses, and the MISG_PE3 can monitor the SubCPU's write accesses.

When register write mode is the signature generation condition, there is no correspondence between a CPU and the signature generation unit. Any CPU can generate a signature in any signature generation unit.

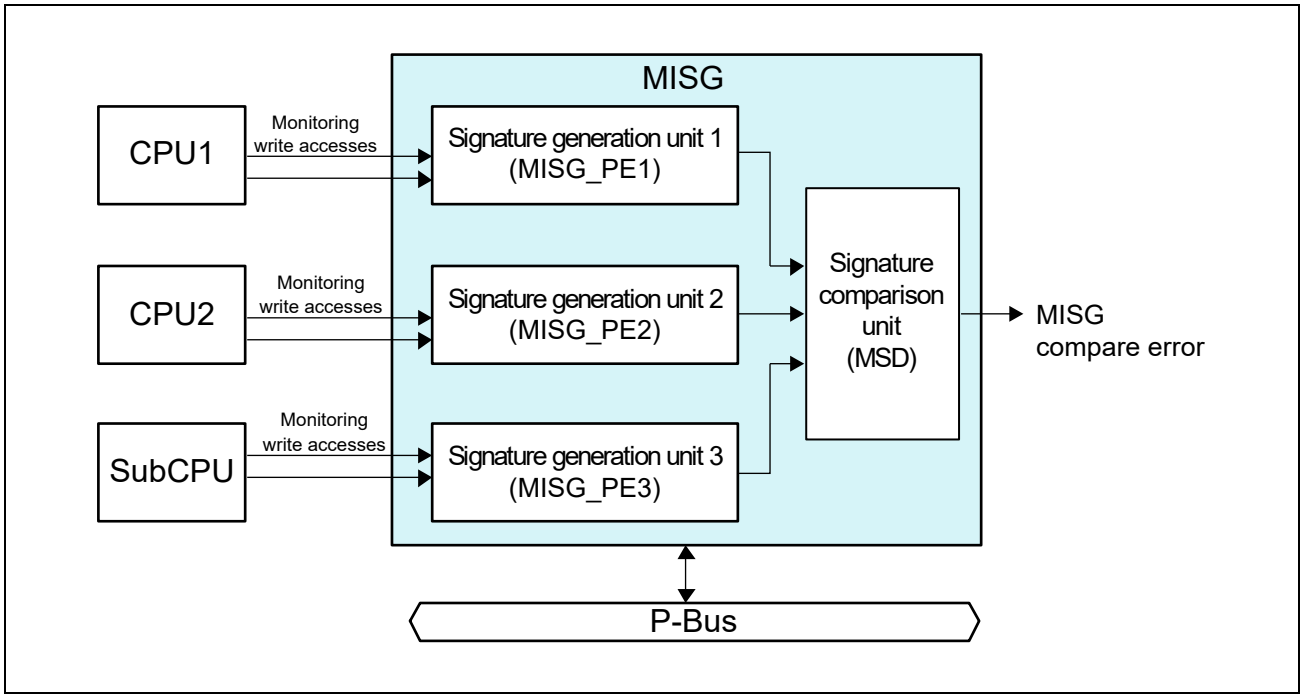


Figure 29.6 MISG Block Diagram (for Multi Core)

29.5.2.2 Signature Generation

The figure below shows the flow of data in signature generation. The signature generation unit consists of two 32-bit signature generators (MISR1_* and MISR2_*).

MISR1_* and MISR2_* can generate a signature from the 64 bits of write data of the CPU to be monitored or the data written to MISRCDR_*.

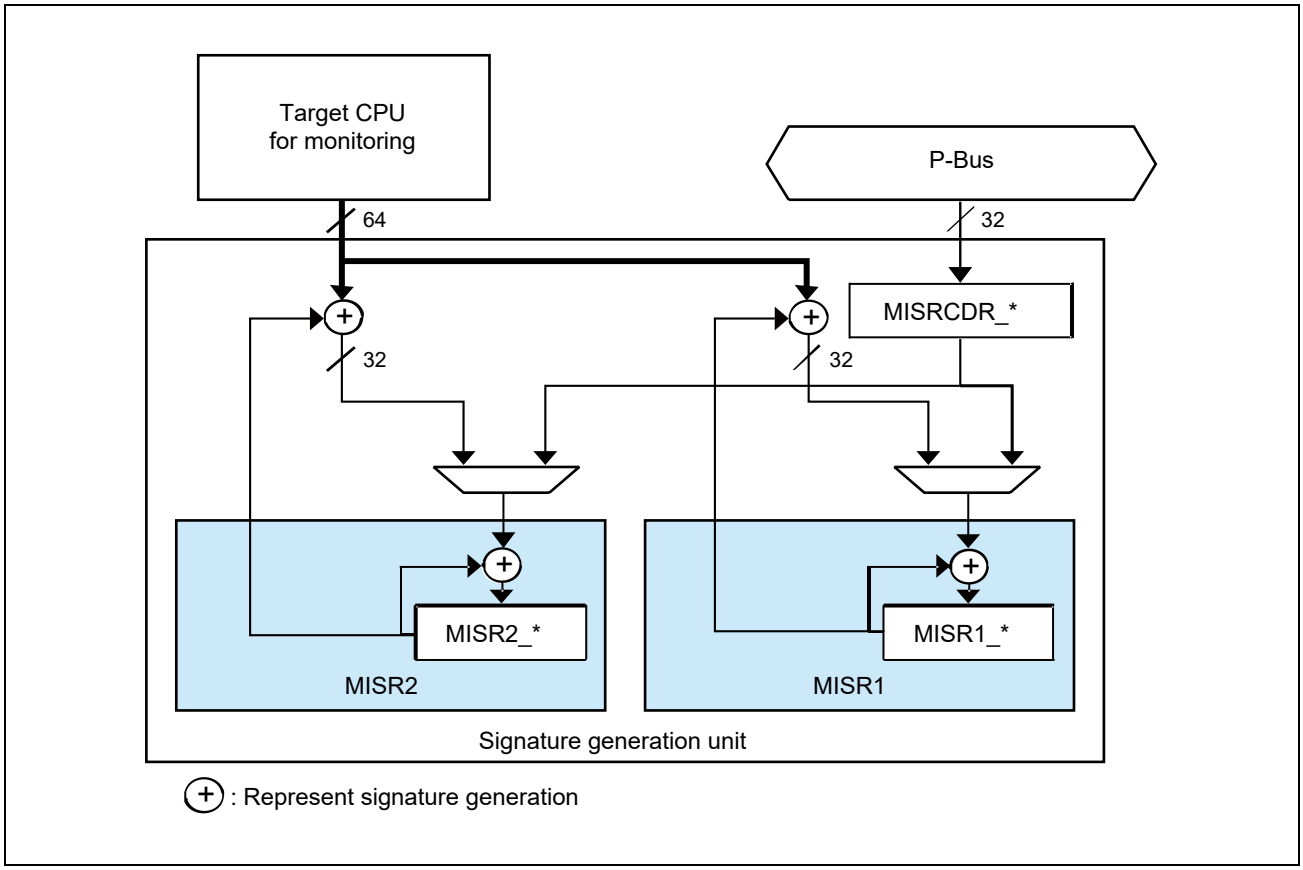


Figure 29.7 Signature Generation Units

A block diagram of signature generation of MISR1_* and the polynomial used for this is shown below.

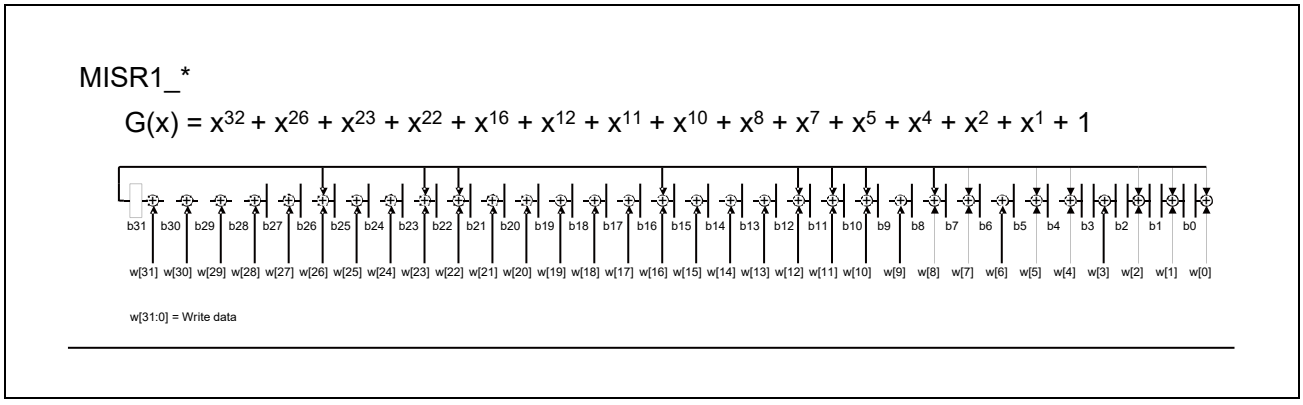


Figure 29.8 Block Diagram of Signature Generation in MISR1_*

A block diagram of signature generation of MISR2_* and the polynomial used for this is shown below.

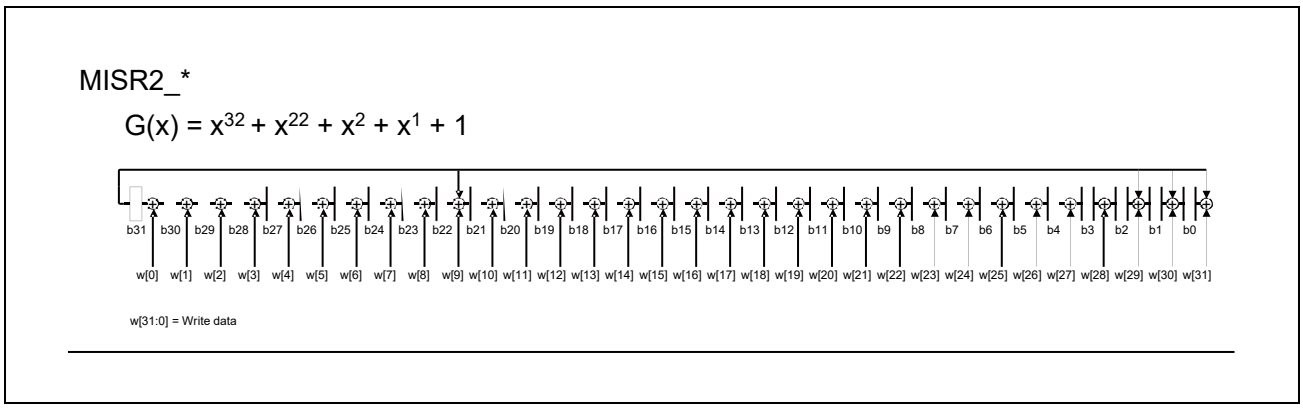


Figure 29.9 Block Diagram of Signature Generation in MISR2_*

29.5.3 Functional Specification

29.5.3.1 Conditions for Signature Generation

The conditions for signature generation in MISR1 and MISR2 can be selected by the setting of the MISR control register (MISRCR).

Table 29.106 Signature Generation Conditions in MISRi (i = 1, 2)

MISRCR. MISRiEN	MISRCR. MISEiCND	Signature Generation Conditions
0	Reserved	MISRi does not generate a signature.
1	0	Register write mode MISRi generates a signature after write access to MISRCDR_*.
1	1	Write monitoring mode MISRi generates a signature after the corresponding CPU writes to an address specified for monitoring.

(1) Register write mode

Writing to the MISR calculation register (MISRCDR_*) while MISR1 is in register write mode leads to the generation of a 32-bit signature from the value held in the multi-input signature register 1 (MISR1_*) and the data written to MISRCDR_*, and the result is retained in MISR1_*. Similarly, a 32-bit signature is generated from the value held in the multi-input signature register 2 (MISR2_*) and the data written to MISRCDR_*, and the result is retained in MISR2_*.

Writing to MISRCDR_* can proceed in 8-, 16-, or 32-bit units, and bits to which a value is not written are treated as 0. For example, writing to the 16 lower-order bits of MISRCDR_* leads to the generation of a signature with the 16 higher-order bits of write data treated as 0. Similarly, writing to the 16 higher-order bits of MISRCDR_* leads to the generation of a signature with the 16 lower-order bits of write data treated as 0. However, these cases will not arise if access is by using the IO header file.

MISR1 and MISR2 do not identify the bus master which writes to the MISRCDR_*. Writing by any bus master, whether a CPU, DMAC, or debugging master, produces a signature.

(2) Write monitoring mode

When the CPU writes to an address area specified for monitoring while MISR1 is in write-monitoring mode, MISR1 generates compressed 32-bit signatures from the value in MISR1_* and 64-bit data being written by the CPU, and generates signatures from the 32-bit compressed data and the value that is again held in MISR1_*. The generated signatures are stored in MISR1_*. Similarly, when the CPU writes to an address area specified for monitoring while MISR2 is in write-monitoring mode, MISR2 generates compressed 32-bit signatures from the value in MISR2_* and 64-bit data being written by the CPU, and generates signatures from the 32-bit compressed data and the value that is again held in MISR2_*. The generated signatures are stored in MISR2_*.

The address area specified for monitoring for signature generation is set by the MISR monitoring area base address register (MISRBASEADR) and the MISR monitoring area address mask register (MISRADRMSK). If the CPU write address is within the address area specified for monitoring, a signature is generated.

Write monitoring mode monitors write access by the corresponding CPU in 8-, -16, -32, or -64 units. When writing proceeds in 8-, or -16 units, remaining bits to which a value is not written are treated as 0, and compressed 32 bit data from 64-bit data is always input to MISR1 or MISR2. Write data are allocated to the lower-order side regardless of the destination address. For example, when 16 bits are written to an address of the form 4N + 2, the 16 bits are allocated to the lower-order side and a signature is generated with the 16 higher-order bits treated as 0.

The CPU write access destinations that can be monitored by MISR1 and MISR2 are as follows:

- Local RAM and global RAM
- CPU peripheral (local APB)
- Peripheral circuit connected to the interconnect or to the P-Bus

The restrictions described below apply to the given store operations.

(1) Instructions that identify write data in slave responses:
Store operations such as BitOp, CAXI, and STC.W are not subject to monitoring.

(2) Instructions for saving units of data larger than 64 bits on the stack (PREPARE and PUSHSP):

Signature values between the masters may not be matched when the instruction is suspended because of an interrupt. Therefore, the area to be used with the instructions must not be subject to monitoring or steps must be taken to make sure that the operation is not suspended due to an interrupt.

Write monitoring mode can only be executed between the specified signature generation unit and the corresponding CPU. In this product, the following monitoring operations are possible.

Monitoring of CPU1 write access by signature generation unit 1

Monitoring of CPU2 write access by signature generation unit 2

Monitoring of SubCPU write access by signature generation unit 3

A signature is not generated if write monitoring mode is set for a signature generation unit that does not support write monitoring mode. At this time, reference to the values of MISRBASEADR and MISRADRMSK is not possible from anywhere.

29.5.3.2 Automatic Signature Comparison

Of the signature generation units, two or three signature generation units are selected for the comparison of signatures by the MISRCMPENi or MISR2CMPENi (i = 0, 1, 2) bit. Each signature generation unit has a data counter, and the signatures are compared if the values of the data counters generated in two signature generation units selected as targets for comparison match.

If two signatures are to be compared, use MISRCMPENi or MISR2CMPENi (i = 0, 1, 2) in the MISRCMPCTL register to enable signature comparison by two signature generation units.

If three signatures are to be compared, use two or three bits of MISRCMPENi or MISR2CMPENi to enable comparison by the corresponding signature generation units.

29.5.3.3 Data Counter

The MISR data counter register (MISRDCNT) counts how many times writing to MISRCDR_* or the address range specified by BASEADR and ADRMSK proceeds.

When the CNTSTA bit = 1 and the CNTTRG bit = 0 in the data counter control register (MISRDCNTCTL), MISRDCNT counts the number of write accesses to MISRCDR_*.

When the CNTSTA bit = 1 and the CNTTRG bit = 1 in MISRDCNTCTL, MISRDCNT counts the number of write accesses by the corresponding CPU to the address range specified by MISRBASEADR and MISRADRMSK.

MISRDCNT can count the number of times either MISR1 or MISR2 or both MISR1 and MISR2 generate signatures by setting the trigger for counting up by MISRDCNT as the signature generation condition in MISR1 or MISR2. Note, however, that if the signature generation conditions for MISR1 and MISR2 and the trigger for counting up by the data counter are not consistent, the value of MISRDCNT and the signature generation count will not match.

CAUTIONS

1. When the MISR1EN and MISR2EN bits in the MISRCR register are both 0, counting by MISRDCNT is not incremented even if writing to MISRCDR_* and the address range specified by MISRBASEADR and MISRADRMSK proceeded.
2. In the signature generation unit that does not support write monitoring mode, the data counter MISRDCNT is not incremented when the CPU write access to the monitoring address area is set as the trigger for counting up by the data counter.

29.5.3.4 Error Notification

When the CMPERREN bit in the error notification control register (MISRERRCTL) is set to 1, the ECM is notified of an error when signatures are compared and do not match. At the same time, the error flag in the compare error status register is set. An interrupt request for the INTC is not generated.

29.5.4 Register Specifications

29.5.4.1 Register Map

The table below lists the registers of the signature generation units.

The registers with symbols ending in “_PE1” are those of signature generation unit 1 (MISG_PE1).

The registers with symbols ending in “_PE2” are those of signature generation unit 2 (MISG_PE2).

The registers with symbols ending in “_PE3” are those of signature generation unit 3 (MISG_PE3).

Note that the endings of the register symbols (“_PE1”, “_PE2", and "PE3”) are omitted if signature generation units 1 to 3 do not require identification.

MISG_PE1_base = FFC5 1000_H

MISG_PE2_base = FFC5 2000_H

MISG_PE3_base = FF75 0000_H

Table 29.107 List of Registers of the Signature Generation Units

Module Name	Register Name	Symbol	Address
MISG	Multi input signature register 1 (PE1)	MISR1_PE1	<MISG_PE1_base> + 000_H
MISG	Multi input signature register 2 (PE1)	MISR2_PE1	<MISG_PE1_base> + 008_H
MISG	MISR calculation data register (PE1)	MISRCDR_PE1	<MISG_PE1_base> + 010_H
MISG	MISR control register (PE1)	MISRCR_PE1	<MISG_PE1_base> + 018_H
MISG	MISR monitoring area base address register (PE1)	MISRBASEADR_PE1	<MISG_PE1_base> + $01C_H$
MISG	MISR monitoring area address mask register (PE1)	MISRADRMSK_PE1	<MISG_PE1_base> + 020_H
MISG	MISR data count control register (PE1)	MISRDCNTCTL_PE1	<MISG_PE1_base> + 024_H
MISG	MISR data count register (PE1)	MISRDCNT_PE1	<MISG_PE1_base> + 028_H
MISG	Multi input signature register 1 (PE2)	MISR1_PE2	<MISG_PE2_base> + 000_H
MISG	Multi input signature register 2 (PE2)	MISR2_PE2	<MISG_PE2_base> + 008_H
MISG	MISR calculation data register (PE2)	MISRCDR_PE2	<MISG_PE2_base> + 010_H
MISG	MISR control register (PE2)	MISRCR_PE2	<MISG_PE2_base> + 018_H
MISG	MISR monitoring area base address register (PE2)	MISRBASEADR_PE2	<MISG_PE2_base> + $01C_H$
MISG	MISR monitoring area address mask register (PE2)	MISRADRMSK_PE2	<MISG_PE2_base> + 020_H
MISG	MISR data count control register (PE2)	MISRDCNTCTL_PE2	<MISG_PE2_base> + 024_H
MISG	MISR data count register (PE2)	MISRDCNT_PE2	<MISG_PE2_base> + 028_H
MISG	Multi input signature register 1 (PE3)	MISR1_PE3	<MISG_PE3_base> + 000_H
MISG	Multi input signature register 2 (PE3)	MISR2_ PE3	<MISG_PE3_base> + 008_H
MISG	MISR calculation data register (PE3)	MISRCDR_ PE3	<MISG_PE3_base> + 010_H
MISG	MISR control register (PE3)	MISRCR_ PE3	<MISG_PE3_base> + 018_H
MISG	MISR monitoring area base address register (PE3)	MISRBASEADR_ PE3	<MISG_PE3_base> + $01C_H$
MISG	MISR monitoring area address mask register (PE3)	MISRADRMSK_ PE3	<MISG_PE3_base> + 020_H
MISG	MISR data count control register (PE3)	MISRDCNTCTL_ PE3	<MISG_PE3_base> + 024_H
MISG	MISR data count register (PE3)	MISRDCNT_ PE3	<MISG_PE3_base> + 028_H

The table below lists the registers of the signature comparison unit (MSD sub-block).

MSD_base = $FFC5\ 0000_H$

Table 29.108 List of Registers of the Signature Comparison Units (MSD Sub-Block)

Module Name	Register Name	Symbol	Address
MISG	MISR comparator control register	MISRCMPCTL	<MSD_base> + 00_H
MISG	MISR compare error status register	MISRCMPERSTR	<MSD_base> + 04_H
MISG	MISR compare error status clear register	MISRCMPERRSTC	<MSD_base> + 08_H
MISG	MISR error notification control register	MISRERRCTL	<MSD_base> + $0C_H$

Note: In case to access to a register in a unit smaller than 32 bits, the bits that are not specified are ignored when they are written and 0 is returned when they are read.

The following describes the control registers of the signature comparison unit.

29.5.4.2 MISRCDR_PE1/PE2/ PE3 — MISR calculation data register

The MISR calculation data register is a 32-bit write-only register.

When signature generation in register write mode is selected, a signature is generated in MISR1 or MISR2 by writing to this register. Data written to this register is the data input to the multi-input signature register 1 (MISR1_*) or the multi-input signature register 2 (MISR2_*). For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

This register can be written in 8-, 16-, or 32-bit units. When writing proceeds in 8- or 16-bit units, remaining bits to which a value is not written are treated as 0, and 32-bit data is always input to MISR1_* or MISR2_*.

Access: MISRCDR_ (PE1/PE2/PE3) is writable in 32-bit units.
MISRCDR_ (PE1/PE2/PE3) (L/H) is writable in 16-bit units.
MISRCDR_ (PE1/PE2/PE3) (LL/LH/HL/HH) is writable in 8-bit units.

Address: MISRCDR_PE1 : <MISG_PE1_base> + 010_H, MISRCDR_PE2 : <MISG_PE2_base> + 010_H,
MISRCDR_PE3 : <MISG_PE3_base> + 010_H,
MISRCDR_ (PE1/PE2/PE3)L : MISRCDR_ (PE1/PE2/PE3) + 00_H,
MISRCDR_ (PE1/PE2/PE3)H : MISRCDR_ (PE1/PE2/PE3) + 02_H,
MISRCDR_ (PE1/PE2/PE3)LL : MISRCDR_ (PE1/PE2/PE3) + 00_H,
MISRCDR_ (PE1/PE2/PE3)LH : MISRCDR_ (PE1/PE2/PE3) + 01_H,
MISRCDR_ (PE1/PE2/PE3)HL : MISRCDR_ (PE1/PE2/PE3) + 02_H,
MISRCDR_ (PE1/PE2/PE3)HH : MISRCDR_ (PE1/PE2/PE3) + 03_H

Value after reset: ---- ---- (undefined)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISRCDR [31:16]															
Value after reset:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISRCDR [15:0]															
Value after reset:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 29.109 MISRCDR_PE1/PE2/PE3 Register Contents

Bit Position	Bit Name	Function
31 to 0	MISRCDR31 to MISRCDR0	Calculation Data Input data to MISR1 or MISR2. When the MISR1EN bit = 1 and the MISR1CND bit = 0 in the MISRCR register, or the MISR2EN bit = 1 and the MISR2CND bit = 0, a new signature is generated each time the MISRCDR register is written, and the result is retained in MISR1 or MISR2.

29.5.4.3 MISR1_PE1/PE2/ PE3 — Multi-input signature register 1

The multi-input signature register is a 32-bit writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$G(x) = x32 + x26 + x23 + x22 + x16 + x12 + x11 + x10 + x8 + x7 + x5 + x4 + x2 + x + 1$

Access: MISR1_PE1, MISR1_PE2, and MISR1_PE3 are readable/writable in 32-bit units.

Address: MISR1_PE1 : <MISG_PE1_base> + 000H,
MISR1_PE2 : <MISG_PE2_base> + 000H,
MISR1_PE3 : <MISG_PE3_base> + 000H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR1 [31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR1 [15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.110 MISR1_PE1/PE2/ PE3 Register Contents

Bit Position	Bit Name	Function
31 to 0	MISR131 to MISR10	Multi-Input Signature Register 1 When read, a new signature is always read.

29.5.4.4 MISR2_PE1/PE2/ PE3 — Multiple-input signature register 2

The multi-input signature register 2 is a 32-bit writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$G(x) = x32 + x22 + x2 + x + 1$

Access: MISR2_PE1, MISR2_PE2, and MISR2_PE3 are readable/writable in 32-bit units.

Address: MISR2_PE1 : <MISG_PE1_base> + 008_H,
MISR2_PE2 : <MISG_PE2_base> + 008_H,
MISR2_PE3 : <MISG_PE3_base> + 008_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR2 [31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR2 [15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.111 MISR2_PE1/PE2/ PE3 Register Contents

Bit Position	Bit Name	Function
31 to 0	MISR231 to MISR20	Multi-Input Signature Register 2 When read, a new signature is always read.

29.5.4.5 MISRCR_PE1/PE2/ PE3 — MISR control register

MISRCR_* is an 8-bit readable and writable register.

The MISR1EN and MISR2EN bits are used to enable or disable signature generation in MISR1 and MISR2.

When the MISR1EN or MISR2EN bit is 1, MISR1 or MISR2 generates a signature and retains the generated value. When the MISR1EN or MISR2EN bit is 0, MISR1 or MISR2 does not generate signatures and the values of these registers are not updated.

When the MISR1EN or MISR2EN bit is 1, the MISR1CND or MISR2CND bit selects the signal generation condition in MISR1 orMISR2. Setting the MISR1CND or MISR2CND bit to 0 selects signature generation in register write mode by MISR1 or MISR2, so writing to MISRCDR_* leads to signature generation. Setting the MISR1CND or MISR2CND bit to 1 selects signature generation in write monitoring mode by MISR1 or MISR2, writing by the CPU to the address range specified by the MISRBASEADR and MISRADRMSK registers leads to signature generation. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Access: MISRCR_PE1, MISRCR_PE2, and MISRCR_PE3 are readable/writable in 8-bit units.

Address: MISRCR_PE1 : <MISG_PE1_base> + 018_H,
MISRCR_PE2 : <MISG_PE2_base> + 018_H,
MISRCR_PE3 : <MISG_PE3_base> + 018_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MISR2CND	MISR1CND	MISR2EN	MISR1EN
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.112 MISRCR_PE1/PE2/ PE3 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	The read value is 0. The write value should be 0.
3	MISR2CND	MISR2 Signature Write Control This bit sets the signature generation condition in MISR2 when the MISR2EN bit = 1. 0: Register write mode 1: Write monitoring mode
2	MISR1CND	MISR1 Signature Write Control This bit sets the signature generation condition in MISR1 when the MISR1EN bit = 1. 0: Register write mode 1: Write monitoring mode
1	MISR2EN	MISR2 Enable 0: MISR2 does not generate a signature. 1: MISR2 generates a signature and the MISR2H and MISR2_* values are updated.
0	MISR1EN	MISR1 Enable 0: MISR1 does not generate a signature. 1: MISR1 generates a signature and the MISR1H and MISR1_* values are updated.

29.5.4.6 MISRBASEADR_PE1/PE2/ PE3 — MISR monitoring area base address register

MISRBASEADR specifies the CPU write access area to be monitored by MISG when signature generation in write monitoring mode is selected. In combination with the setting of the monitoring area mask address register, this register specifies the address range of the monitoring area. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Access: MISRBASEADR_ (PE1/PE2/PE3) is readable/writable in 32-bit units.
MISRBASEADR_ (PE1/PE2/PE3) (L/H) is readable/writable in 16-bit units.
MISRBASEADR_ (PE1/PE2/PE3) (LL/LH/HL/HH) is readable/writable in 8-bit units.

Address: MISRBASEADR_PE1 : <MISG_PE1_base> + 01C$_H$,
MISRBASEADR_PE2 : <MISG_PE2_base> + 01C$_H$,
MISRBASEADR_PE3 : <MISG_PE3_base> + 01C$_H$,
MISRBASEADR_ (PE1/PE2/PE3)L : MISRBASEADR_ (PE1/PE2/PE3) + 00$_H$,
MISRBASEADR_ (PE1/PE2/PE3)H : MISRBASEADR_ (PE1/PE2/PE3) + 02$_H$,
MISRBASEADR_ (PE1/PE2/PE3)LL : MISRBASEADR_ (PE1/PE2/PE3) + 00$_H$,
MISRBASEADR_ (PE1/PE2/PE3)LH : MISRBASEADR_ (PE1/PE2/PE3) + 01$_H$,
MISRBASEADR_ (PE1/PE2/PE3)HL : MISRBASEADR_ (PE1/PE2/PE3) + 02$_H$,
MISRBASEADR_ (PE1/PE2/PE3)HH : MISRBASEADR_ (PE1/PE2/PE3) + 03$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BASEADR[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BASEADR[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.113 MISRBASEADR_PE1/PE2/ PE3 Register Contents

Bit Position	Bit Name	Function
31 to 0	BASEADR31 to BASEADR0	Monitoring Area Base Address Register

For the mechanism to judge access to the monitoring area, see **Section 29.5.4.7, MISRADRMSK_PE1/PE2/PE3 — MISR monitor area address mask register**.

29.5.4.7 MISRADRMSK_PE1/PE2/PE3 — MISR monitor area address mask register

MISRADRMSK specifies the CPU write access area to be monitored by MISG when signature generation in write monitoring mode is selected. In combination with the setting of the monitoring area base address register, this register specifies the address range of the monitoring area. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Access: MISRADRMSK_ (PE1/PE2/PE3) readable/writable in 32-bit units.
MISRADRMSK_ (PE1/PE2/PE3) (L/H) is readable/writable in 16-bit units.
MISRADRMSK_ (PE1/PE2/PE3) (LL/LH/HL/HH) is readable/writable in 8-bit units.

Address: MISRADRMSK_PE1 : <MISG_PE1_base> + 020H,
MISRADRMSK_PE2 : <MISG_PE2_base> + 020H,
MISRADRMSK_PE3 : <MISG_PE3_base> + 020H,
MISRADRMSK_ (PE1/PE2/PE3)L : MISRADRMSK_ (PE1/PE2/PE3) + 00H,
MISRADRMSK_ (PE1/PE2/PE3)H : MISRADRMSK_ (PE1/PE2/PE3) + 02H,
MISRADRMSK_ (PE1/PE2/PE3)LL : MISRADRMSK_ (PE1/PE2/PE3) + 00H,
MISRADRMSK_ (PE1/PE2/PE3)LH : MISRADRMSK_ (PE1/PE2/PE3) + 01H,
MISRADRMSK_ (PE1/PE2/PE3)HL : MISRADRMSK_ (PE1/PE2/PE3) + 02H,
MISRADRMSK_ (PE1/PE2/PE3)HH : MISRADRMSK_ (PE1/PE2/PE3) + 03H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADRMSK[31:16]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADRMSK[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.114 MISRADRMSK_PE1/PE2/ PE3 Register Contents

Bit Position	Bit Name	Function
31 to 0	ADRMSK31 to ADRMSK0	Monitoring Area Mask Address Register

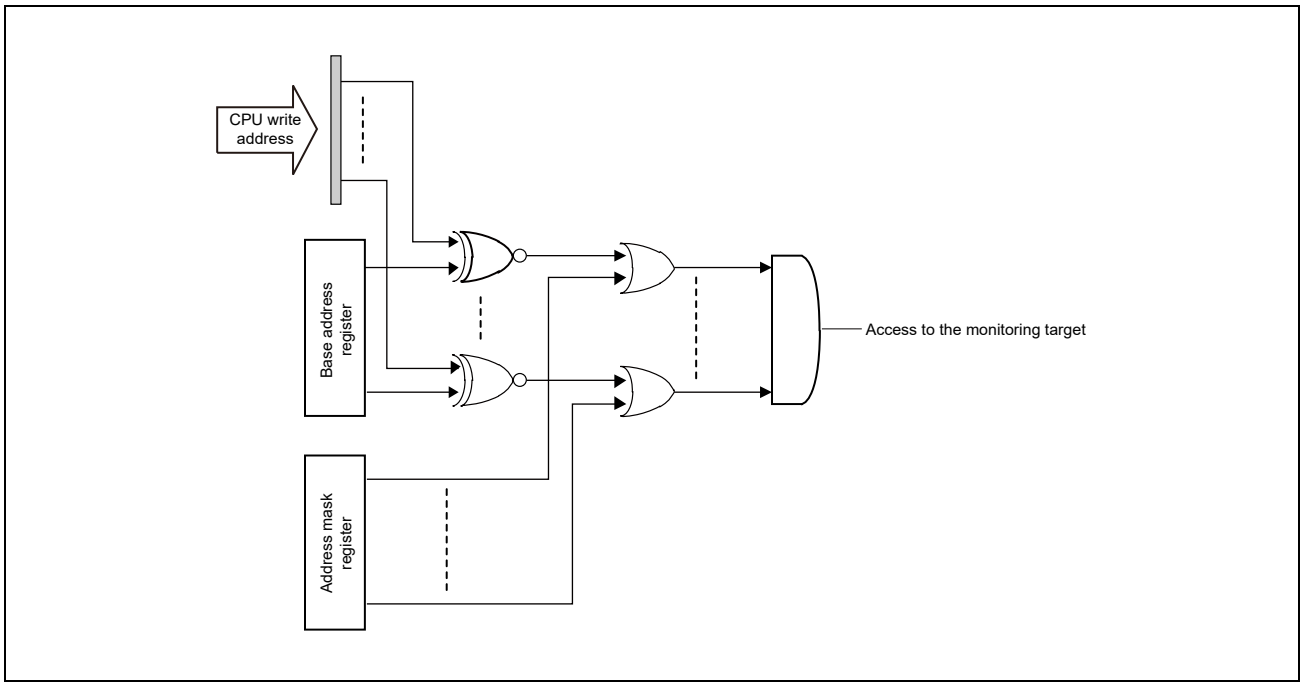


Figure 29.10 How to Detect Access to the Monitoring Area

29.5.4.8 MISRDCNTCTL_PE1/PE2/ PE3 — MISR data counter control register

This register controls operation of the MSIR data counter register. If the event selected by the CNTTRG bit occurs while the CNTSTA bit is 1, the data counter is incremented. For operation of the data counter, see **Section 29.5.3.3, Data Counter**.

Access: MISRDCNTCTL_PE1, MISRDCNTCTL_PE2, and MISRDCNTCTL_PE3 are readable/writable in 8-bit units.

Address: MISRDCNTCTL_PE1 : <MISG_PE1_base> + 024H,
MISRDCNTCTL_PE2 : <MISG_PE2_base> + 024H,
MISRDCNTCTL_PE3 : <MISG_PE3_base> + 024H

Value after reset: 00H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CNTTRG	CNTSTA
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 29.115 MISRDCNTCTL_PE1/PE2/ PE3 Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	These bits are always read as 0. The write value should be 0.
1	CNTTRG	Count-Up Trigger Select This bit selects the trigger for counting up by the data counter. 0: Write access to the MISRCDR_* register 1: Write access to the address area specified by the MISRBASEADR and MISRADRMSK registers
0	CNTSTA	Data Counter Start This bit is an enable bit for the data counter. If the event selected by the CNTTRG bit occurs while CNTSTA = 1, the data counter is incremented. 0: The data counter is stopped. 1: The data counter is operating.

29.5.4.9 MISRDCNT_PE1/PE2/ PE3 — MISR data counter register

The data counter is a 16-bit readable and writable register.

If the data counter values of two signature generation units selected for comparison match, automatic comparison of the signatures proceeds.

When the CNTTRG bit in the data counter control register is 0, the data counter is incremented by write access to the MISR calculation data register. When the CNTTRG bit in the data control register is 1, the data counter is incremented by write access by the corresponding CPU to the address area specified by the MISRBASEADR and MISRADRMSK registers. For operation of the data counter, see **Section 29.5.3.3, Data Counter**.

Access: MISRDCNT_PE1, MISRDCNT_PE2, and MISRDCNT_PE3 are readable/writable in 16-bit units.
MISRDCNT_PE1L, MISRDCNT_PE1H, MISRDCNT_PE2L, MISRDCNT_PE2H, MISRDCNT_PE3L, and MISRDCNT_PE3H are readable/writable in 8-bit units.

Address: MISRDCNT_PE1 : <MISG_PE1_base> + 028_H,
MISRDCNT_PE2 : <MISG_PE2_base> + 028_H,
MISRDCNT_PE3 : <MISG_PE3_base> + 028_H,
MISRDCNT_PE1L : <MISG_PE1_base> + 028_H, MISRDCNT_PE1H : <MISG_PE1_base> + $02A_H$,
MISRDCNT_PE2L : <MISG_PE2_base> + 028_H, MISRDCNT_PE2H : <MISG_PE2_base> + $02A_H$,
MISRDCNT_PE3L : <MISG_PE3_base> + 028_H, MISRDCNT_PE3H : <MISG_PE3_base> + $02A_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCNT[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.116 MISRDCNT_PE1/PE2/PE3 Register Contents

Bit Position	Bit Name	Function
15 to 0	DCNT15 to DCNT0	Data Counter Register

The following describes the control registers of the MSD unit (signature comparison unit).

29.5.4.10 MISRCMPCTL — MISR comparator control register

MISRCMPCTL is a 16-bit readable and writable register.

This register controls the comparator that compares signatures generated in the signature generation units. For automatic comparison of signatures, see **Section 29.5.3.2, Automatic Signature Comparison**.

Access: MISRCMPCTL is readable/writable in 16-bit units.
MISRCMPCTLL and MISRCMPCTLH are readable/writable in 8-bit units.

Address: MISRCMPCTL : <MSD_base> + 00_H,
MISRCMPCTLL : <MSD_base> + 00_H, MISRCMPCTLH : <MSD_base> + 01_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	MISR2CMPEN2	MISR2CMPEN1	MISR2CMPEN0	MISR1CMPEN2	MISR1CMPEN1	MISR1CMPEN0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.117 MISRCMPCTL Register Contents

Bit Position	Bit Name	Function
15 to 6	Reserved	These bits are always read as 0. The write value should be 0.
5	MISR2CMPEN2	MISR2 Signature Compare Enable 2 This bit controls comparison of signatures retained in MISR2 of MISG_PE3 and MISG_PE1. 0: Disables comparison 1: Enables comparison
4	MISR2CMPEN1	MISR2 Signature Compare Enable 1 This bit controls comparison of signatures retained in MISR2 of MISG_PE2 and MISG_PE3. 0: Disables comparison 1: Enables comparison
3	MISR2CMPEN0	MISR2 Signature Compare Enable 0 This bit controls comparison of signatures retained in MISR2 of MISG_PE1 and MISG_PE2. 0: Disables comparison 1: Enables comparison
2	MISR1CMPEN2	MISR1 Signature Compare Enable 2 This bit controls comparison of signatures retained in MISR1 of MISG_PE3 and MISG_PE1. 0: Disables comparison 1: Enables comparison
1	MISR1CMPEN1	MISR1 Signature Compare Enable 1 This bit controls comparison of signatures retained in MISR1 of MISG_PE2 and MISG_PE3. 0: Disables comparison 1: Enables comparison
0	MISR1CMPEN0	MISR1 Signature Compare Enable 0 This bit controls comparison of signatures retained in MISR1 of MISG_PE1 and MISG_PE2. 0: Disables comparison 1: Enables comparison

29.5.4.11 MISRCMPERSTR — MISR compare error status register

The compare error status register is an 8-bit readable register.

If a mismatch occurs in signature comparison enabled by the comparator control register, the corresponding error flag is set.

The error flag is cleared by writing 1 to the corresponding clear bit in the compare error status clear register. The flag is also cleared by a reset.

Access: MISRCMPERSTR is readable in 8-bit units.

Address: MISRCMPERSTR : <MSD_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	MISR2ERR2	MISR2ERR1	MISR2ERR0	MISR1ERR2	MISR1ERR1	MISR1ERR0
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 29.118 MISRCMPERSTR Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	The read value is 0. The write value should be 0.
5	MISR2ERR2	MISR2 Signature Compare Error Flag 2 This flag is set when signatures retained in MISR2 of MISG_PE1 and MISG_PE3 are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.
4	MISR2ERR1	MISR2 Signature Compare Error Flag 1 This flag is set when signatures retained in MISR2 of MISG_PE2 and MISG_PE3 are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.
3	MISR2ERR0	MISR2 Signature Compare Error Flag 0 This flag is set when signatures retained in MISR2 of MISG_PE1 and MISG_PE2 are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.
2	MISR1ERR2	MISR1 Signature Compare Error Flag 2 This flag is set when signatures retained in MISR1 of MISG_PE1 and MISG_PE3 are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.
1	MISR1ERR1	MISR1 Signature Compare Error Flag 1 This flag is set when signatures retained in MISR1 of MISG_PE2 and MISG_PE3 are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.
0	MISR1ERR0	MISR1 Signature Compare Error Flag 0 This flag is set when signatures retained in MISR1 of MISG_PE1 and MISG_PE2 are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.

29.5.4.12 MISRCMPERRSTC — MISR compare error status clear register

The compare error status clear register is an 8-bit write-only register.

When an error flag in the compare error status register is 1, writing 1 to corresponding clear bit clears the error flag. Read the MISR compare error status register and write 1 to the clear bit for the flag being 1.

Access: MISRCMPERRSTC is writable in 8-bit units.

Address: MISRCMPERRSTC : <MSD_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	MISR2CLR2	MISR2CLR1	MISR2CLR0	MISR1CLR2	MISR1CLR1	MISR1CLR0
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	W	W	W	W	W	W

Table 29.119 MISRCMPERRSTC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	The read value is 0. The write value should be 0.
5	MISR2CLR2	MISR2 Signature Compare Error Clear 2 Writing 1 to this bit clears the MISR2ERR2 bit in the MISRCMPERSTR register.
4	MISR2CLR1	MISR2 Signature Compare Error Clear 1 Writing 1 to this bit clears the MISR2ERR1 bit in the MISRCMPERSTR register.
3	MISR2CLR0	MISR2 Signature Compare Error Clear 0 Writing 1 to this bit clears the MISR2ERR0 bit in the MISRCMPERSTR register.
2	MISR1CLR2	MISR1 Signature Compare Error Clear 2 Writing 1 to this bit clears the MISR1ERR2 bit in the MISRCMPERSTR register.
1	MISR1CLR1	MISR1 Signature Compare Error Clear 1 Writing 1 to this bit clears the MISR1ERR1 bit in the MISRCMPERSTR register.
0	MISR1CLR0	MISR1 Signature Compare Error Clear 0 Writing 1 to this bit clears the MISR1ERR0 bit in the MISRCMPERSTR register.

29.5.4.13 MISRERRCTL — MISR error notification control register

The error notification control register is an 8-bit readable and writable register.

This register enables or disables error notification when signatures are compared by the automatic signature comparison and do not mach. For the automatic signature comparison and error notification, see **Section 29.5.3.2, Automatic Signature Comparison** and **29.5.3.4, Error Notification.**

Access: MISRERRCTL is readable/writable in 8-bit units.

Address: MISRERRCTL : <MSD_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CMPERREN
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 29.120 MISRERRCTL Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	These bits are always read as 0. The write value should be 0.
0	CMPERREN	Compare Error Notification Enable This bit enables or disables error notification when signatures are compared and do not mach. 0: Error notification does not proceed even if a mismatch has occurred in signature comparison. 1: Error notification proceeds if a mismatch has occurred in signature comparison.

29.5.5 Usage Example

29.5.5.1 Usage example 1

The self-diagnostic program eases diagnosis by CPU. By using the MISG to compress intermediate transitions, the self-diagnostic program makes the saving and comparison of all intermediate results unnecessary.

Results of self-diagnosis including intermediate transitions can be judged by comparing the results of compression by the MISG with the expected results when the program ends.

This has the effect of reducing requirements for memory capacity and times for processing comparisons (substitution by CRCs is also possible).

Setting example (when PE1 is running the self-diagnosis program)

The descriptions of registers of an MISG below applies to the registers of signature generation unit 1

(MISG_PE1), which is for PE1.

(1) Initialize the multi-input signature register 1 (MISR1_*), multi-input signature register 2 (MISR2_*) and data counter register (MISRDCNT).

(2) Set the MISR1CND or MISR2CND bit in the MISR control register (MISRCR) to 1 to select signature generation by MISR1 or MISR2 in write monitoring mode.

(3) Use the MISR monitoring area base address register (MISRBASEADR) and MISR monitoring area address mask register (MISRADRMSK) to set the address area for monitoring.

(4) Set the MISR1EN or MISR2EN bit in the MISR control register to 1 to enable signature generation in MISR1 or MISR2.

(5) Run the self-diagnostic program on PE1.

(6) Upon completion of execution of the self-diagnostic program, MISR1_* or MISR2_* data is compared with the expected value in flash memory.

29.5.5.2 Usage example 2

Use multiple processors to run the same processing (including the self-diagnostic program) to confirm the correctness of results. Comparing results from different hardware raises reliability.

Setting example (when PE1 and PE3 are running the same task)

The descriptions of registers of an MISG below applies to the MISG registers of PE1 and PE3.

(1) Initialize the multi-input signature register 1 (MISR1_*), multi-input signature register 2 (MISR2_*), and data counter register (MISRDCNT).

(2) Set the MISR1CND or MISR2CND bit in the MISR control register (MISRCR) to 0 to select signature generation by MISR1 or MISR2 in register write mode.

(3) Set MISR1CMPEN1 or MISR2CMPEN1 in the MISR comparator control register (MISRCMPCTL) to 1 to enable signature comparison by the comparator.

(4) Set the CMPERREN bit in the MISR error notification control register to 1 to enable error notification to the ECM.

(5) Set the CNTTRG bit in the MISR data counter control register (MISRDCNTCTL) to 0 to set writing to MISRCDR_* as the trigger for counting up by the data counter. Set the CNTSTA bit to 1 to enable operation of the data counter.

(6) Set the MISR1EN or MISR2EN bit in the MISR control register to 1 to enable signature generation in MISR1 or MISR2.

(7) Run the self-diagnostic program on all CPUs.

(8) The self-diagnostic program stores intermediate results while the program is running in MISRCDR_* of the signature generation units. The signatures in the MISR1 and MISR2 registers of MISG_PE1 and MISG_PCU are compared whenever the values of the data counter registers (MISRDCNT) of MISG_PE1 and MISG_PCU match.

(9) Check the comparison status register to see if there were errors in comparison.

29.6 Clock Monitors

29.6.1 Overview

This product incorporates the clock monitors to monitor the clock operation by detecting the abnormal frequency of the clock to be monitored. The clock monitors provide the following functions.

- Monitors to see if the frequency of the clock to be monitored is within the specified range based on the sampling clock.
- Issues an error notice to the ECM upon detection of the abnormal state of the clock.

The table below shows the clocks monitored by the clock monitors and the sampling clocks used.

Table 29.121 List of Clocks Monitored by Each Clock Monitor and Sampling Clocks Used

Clock Monitor Channel	Monitor Clock	Sampling Clock
CLMA0	Low-speed peripheral clock (40 MHz SSG)	Unmodulated low-speed peripheral clock (40 MHz clean)
CLMA1	Non-modulated low-speed peripheral clock (40 MHz clean)	10 MHz (1/2 main OSC)
CLMA2	WDTA counter clock (1/80 of the main OSC or LS IntOSC)*1	LS IntOSC (when the WDTA counter clock is 1/80 of the main OSC) or 1/80 of the main OSC (when the WDTA counter clock is LS IntOSC)
CLMA3	LS IntOSC (when the WDTA counter clock is 1/80 of the main OSC) or 1/80 of the main OSC (when the WDTA counter clock is LS IntOSC)	WDTA counter clock (1/80 of the main OSC or the LS IntOSC)*1

Note 1. For switching the WDTA counter clock, see **Section 10, Clock Controller**.

29.6.2 List of Registers

29.6.2.1 Clock Monitor Channel Register

Table 29.122 List of Registers

Module Name	Register Name	Symbol	Address
CLMAn	CLMAn control register 0	CLMAnCTL0	<Base_adr> + 00_H
CLMAn	CLMAn compare register L	CLMAnCMPL	<Base_adr> + 08_H
CLMAn	CLMAn compare register H	CLMAnCMPH	<Base_adr> + $0C_H$
CLMAn	CLMAn protection command register	CLMAnPCMD	<Base_adr> + 10_H
CLMAn	CLMAn protection command register	CLMAnPS	<Base_adr> + 14_H

The base addresses of the registers are shown below.

Table 29.123 List of Base Address Registers

Clock Monitor Channel	<Base_addr>
CLMA0	FFF8 8400_H
CLMA1	FFF8 8420_H
CLMA2	FFF8 8440_H
CLMA3	FFF8 8460_H

29.6.2.2 Shared registers

Table 29.124 List of Shared Registers

Module Name	Register Name	Symbol	Address	Protection
CLMAC	CLMA Self-Test Register	CLMATEST	FFF8 8204_H	PROT1PHCMD*1
CLMAC	CLMA Self-Test Status Register	CLMATESTS	FFF8 8208_H	

Note 1. For this register, see **Section 10, Clock Controller**.

29.6.3 Details of Registers

29.6.3.1 CLMAnCTL0 — CLMAn control register 0

CLMAnCTL0 controls the operation of the clock monitors. CLMAnCTL0 is protected by the CLMAnPCMD register.

CLMAnCTL0 can be initialized by either an internal reset or an external reset.

Access: CLMAnCTL0 is readable/writable in 8-bit units.

Address: CLMAnCTL0 : <Base_adr> + 00_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CLMAnCLME
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W*1

Note 1. This register can be cleared by a reset. Writing 0 to this register is ignored.

Table 29.125 CLMAnCTL0 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	These bits are always read as 0. When writing, follow the procedure of the protection release sequence and write the value after reset or the inverse.
0	CLMAnCLME	Clock Monitor Operation 0: Disables operation. 1: Enables operation.

29.6.3.2 CLMAnCMPL — CLMAn compare register L

CLMAnCMPL sets the lower limit of the normal frequency range used for comparison.

CLMAnCMPL can be initialized by either an internal reset or an external reset.

Access: CLMAnCMPL is readable/writable in 16-bit units.

Address: CLMAnCMPL : <Base_adr> + 08H

Value after reset: 0001H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLMAnCMPL[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.126 CLMAnCMPL Register Contents

Bit Position	Bit Name	Function
15 to 12	Reserved	The read value is 0. The write value should be 0.
11 to 0	CLMAnCMPL[11:0]	Lower Limit of Normal Frequency Range These bits can be written to when CLMAnCTL0.CLMAnCLME is 0. Once CLMAnCTL0.CLMAnCLME is set to 1, writing to these bits is invalid.

29.6.3.3 CLMAnCMPH — CLMAn compare register H

CLMAnCMPH sets the upper limit of the normal frequency range used for comparison.

CLMAnCMPH can be initialized by either an internal reset or an external reset.

Access: CLMAnCMPH is readable/writable in 16-bit units.

Address: CLMAnCMPH : <Base_adr> + 0CH

Value after reset: 03FFH

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLMAnCMPH[11:0]											
Value after reset:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.127 CLMAnCMPH Register Contents

Bit Position	Bit Name	Function
15 to 12	Reserved	The read value is 0. The write value should be 0.
11 to 0	CLMAnCMPH[11:0]	Upper Limit of Normal Frequency Range These bits can be written to when CLMAnCTL0.CLMAnCLME is 0. Once CLMAnCTL0.CLMAnCLME is set to 1, writing to these bits is invalid.

29.6.3.4 CLMAnPCMD — CLMAn protection command register

CLMAnPCMD is a special sequential register for CLMAnCTL0.

CLMAnPCMD can be initialized by either an internal reset or an external reset.

Access: CLMAnPCMD is writable in 8-bit units.

Address: CLMAnPCMD : <Base_adr> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CLMAnPCMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

See **Section 29.6.6.1, Writing to protected registers** for details on the protection method.

29.6.3.5 CLMAnPS — CLMAn protection command status register

CLMAnPS is a special sequential register for CLMAnCTL0.

CLMAnPS can be initialized by either an internal reset or an external reset

Access: CLMAnPS is readable in 8-bit units.

Address: CLMAnPS : <Base_adr> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CLMAnPRERR
Value after reset:	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 29.128 CLMAnPS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	These bits are always read as 0.
0	CLMAnPRERR	Protection Error Detection 0: A protection error has not occurred. 1: A protection error has occurred.

Operating conditions of the CLMAnPRERR bit:

Setting condition: Access to CLMAnCTL0, which is the target for protection specified by CLMAnPCMD, without following the protection cancelling sequence.

Clearing condition: Writing of $A5_H$ to the CLMAnPCMD register (step 1 in the protection cancelling sequence).

29.6.3.6 CLMATEST — CLMA self-test register

CLMATEST is used for self-testing of CLMA3 to CLMA0.

CLMATEST can be protected by the PROT1PHCMD register.

CLMATEST is initialized by an internal reset or an external reset.

Access: CLMATEST is readable/writable in 32-bit units.

Address: CLMATEST : FFF8 8204H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CLMA3 TESTEN	CLMA2 TESTEN	CLMA1 TESTEN	CLMA0 TESTEN	ERRMSK	MONCL KMSK	RESCLM
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.129 CLMATEST Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	These bits are always read as 0. When writing, follow the procedure of the protection release sequence and write the value after reset or the inverse.
6 to 3	CLMA3TESTEN CLMA2TESTEN CLMA1TESTEN CLMA0TESTEN	These bits enable or disable self-testing of CLMA3 to CLMA0. 0: Disables self-testing of the corresponding CLMAn. 1: Enables self-testing of the corresponding CLMAn.
2	ERRMSK	Masks an error notification to the ECM when CLMAn detects an error. When the ERRMSK is set for a certain CLMAn, the associated CLMAn does not issue an error notification to the ECM even if it detects an error. ERRMSK setting is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 3) is set to 1. 0: Does not mask an error notification to the ECM. 1: Masks an error notification to the ECM.
1	MONCLKMSK	Fixes the level of the clock input to the CLMAn that should be monitored, to the low level. Setting of MONCLKMSK is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 3) is set to 1. 0: Does not fix the clock input to the CLMAn that should be monitored to the low level. 1: Fixes the clock input to the CLMAn that should be monitored to the low level.
0	RESCLM	Initializes CLMAn forcibly. RESCLM setting is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 3) is set to 1. 0: Does not initialize CLMAn. 1: Initializes CLMAn

29.6.3.7 CLMATESTS — CLMA self-test status register

CLMATESTS indicates the self-testing result of CLMA3 to CLMA0.

CLMATESTS is initialized by an internal reset or an external reset.

Access: CLMATESTS is readable in 32-bit units.

Address: CLMATESTS : FFF8 8208H

Value after reset: 0000 0000H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CLMA3 ERRS	CLMA2 ERRS	CLMA1 ERRS	CLMA0 ERRS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.130 CLMATESTS Register Contents

Bit Position	Bit Name	Function
31 to 4	Reserved	These bits are always read as 0.
3 to 0	CLMA3ERRS CLMA2ERRS CLMA1ERRS CLAM0ERRS	These bits indicate whether or not CLMA3 to CLMA0 have detected an error. These bits are not affected by CLMATEST.ERRMSK. 0: The corresponding CLMAn has not detected an error. 1: The corresponding CLMAn has detected an error.

29.6.4 Detection of Abnormal Clock Frequency

1. CLMAn counts the number of rising edges of the monitored clock signal within 16 sampling clock cycles, and compares the count with the specified thresholds.
   - The lower threshold is specified using the CLMAnCMPL[11:0] bits in CLMAnCMPL register.
   - The upper threshold is specified using the CLMAnCMPH[11:0] bits in CLMAnCMPH register.
2. When the frequency of the monitored clock is so low*1 that the counter value falls below the value set in CLMAnCMPL[11:0] in the CLMAnCMPL register, CLMAn notifies the ECM of the abnormal clock. CLMAn also notifies the ECM of the abnormal clock when the clock frequency is so high that the counter value exceeds the value set in CLMAnCMPH[11:0] in the CLMAnCMPH register.

Note that even if the frequency of the monitored clock fluctuates during the sampling period, an error is not notified as long as the number of detected edges falls within the specified range.

Note 1. The abnormal state of the clock may not be detected when the monitored clock comes to a complete stop.

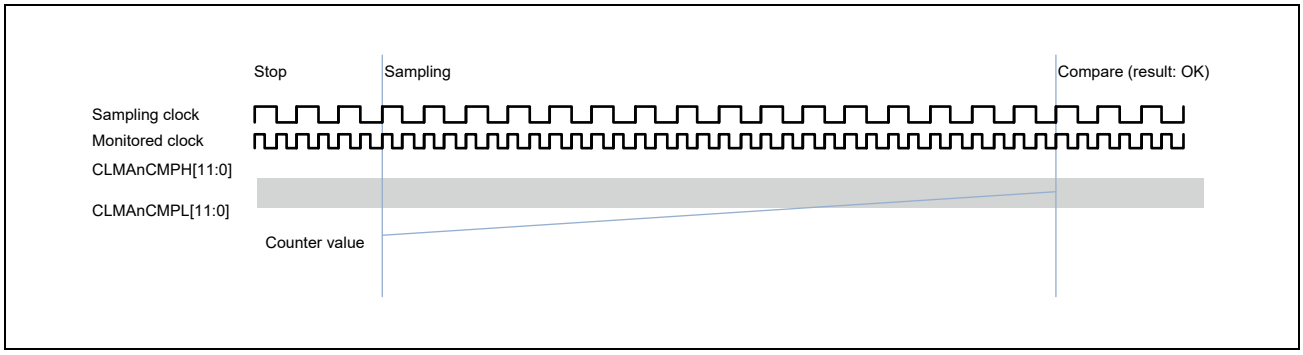


Figure 29.11 Operation when Clock Frequency is within the Specified Range

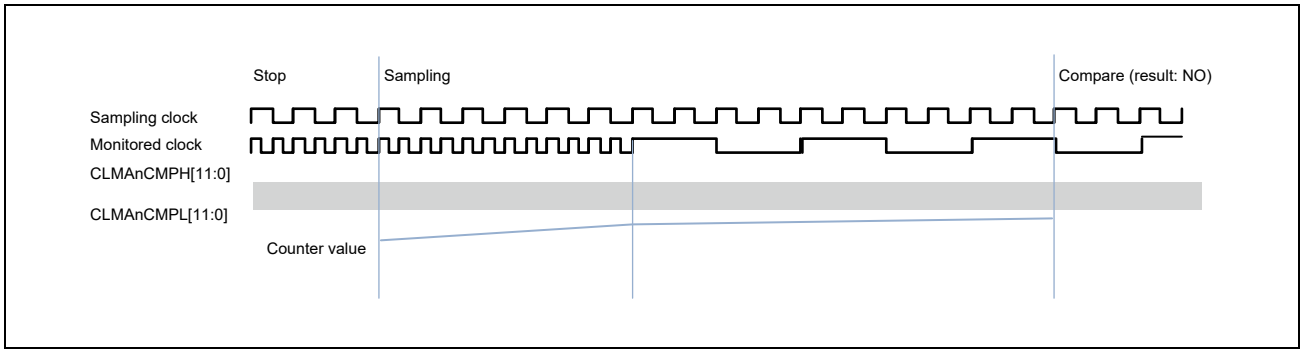


Figure 29.12 Operation when Clock Frequency is Lower than the Specified Range

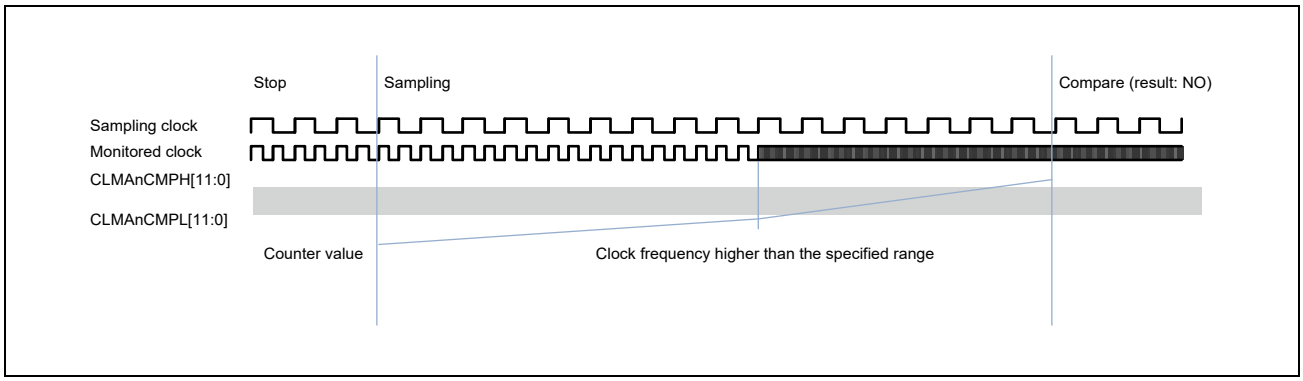


Figure 29.13 Operation when Clock Frequency is Higher than the Specified Range

(1) Calculating the thresholds set by CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]

The minimum and maximum values of the clock cycles of the monitored clock CLMATMON that are assumed to be valid within 16 cycles of the sampling clock CLMATSMP are set in the comparison registers CLMAnCMPL and CLMAnCMPH.

The expected clock cycle is indicated by "N".

$$\frac{16}{f_{CLMATSMP}} = \frac{N}{f_{CLMATMON}}$$

$$N = \frac{f_{CLMATMON}}{f_{CLMATSMP}} \times 16$$

Considering the allowed frequency deviations of CLMATMON and CLMATSMP, the threshold values can be calculated by the following formulas:

Lower threshold $= N_{min}$

$$= \frac{f_{CLMATMON\ (min)}}{f_{CLMATSMP\ (max)}} \times 16 - 1$$

Upper threshold $= N_{max}$

$$= \frac{f_{CLMATMON\ (max)}}{f_{CLMATSMP\ (min)}} \times 16 + 1$$

NOTE

PLL jitter is covered by " + 1" and " – 1" as in the formulas.

Example

If $f_{CLMATSMP}$ = 240 kHz (±8%) and $f_{CLMATMON}$ = 16 MHz (±5%), the recommendable thresholds will be as follows:

N_{min} = 15,200 / 259.2×16 – 1

= 937.27

CLMAnCMPL = 937 = $03A9_H$

N_{max} = 16,800 / 220.8×16 + 1

= 1,218.39

CLMAnCMPH = 1,219 = $04C3_H$

Minimum threshold:

The following restrictions must be taken into consideration.

- CLMAnCMPL ≥ 0001_H
- CLMAnCMPH ≥ CLMAnCMPL + 0003_H

(2) Definition of initial values input to the threshold registers

The values of the threshold registers after a reset are as follows.

- CLMAnCMPL [11:0] = 001_H
- CLMAnCMPH [11:0] = $3FF_H$

29.6.5 Self-Diagnosis

Self-diagnosis of the clock monitor is available as described below.

(1) Set the thresholds for the clock monitor to be subject to self-diagnosis (set CLMAnCMPL and CLMAnCMPH). At this time, the thresholds must be set such that an error will be produced.

(2) Specify the clock monitor to be subject to self-diagnosis.
Setting the CLMATEST.CLMAnTESEN bit to 1 enables specifying the corresponding clock monitor to be subject to self-diagnosis.

(3) To prevent error notice to the ECM based on self-diagnosis, set CLMATEST.ERRMSKL simultaneously with step (2).

(4) Set the CLMAnCTL0.CLMAnCLME bit to 1 to enable operation for clock monitoring.

(5) Wait for the time long enough to allow error occurrence, read the CLMATESTS register to see if an error has been generated in the clock monitor to be self-diagnosed. The time from the start of self-diagnosis to the error occurrence depends on the sampling period. A maximum of two sampling cycles are required.

(6) Clear the error generated by self-diagnosis.
Setting CLMATEST.RESCLM to 1 enables initializing the clock monitor to be self-diagnosed.

(7) Terminate self-diagnosis.
Setting all the bits in CLMATEST to 0 enables terminating self-diagnosis.

Before restarting the clock monitor that has been self-diagnosed, set the registers again as required.

29.6.6 Notes on Register Setting

29.6.6.1 Writing to protected registers

Writing to the CLMAnCTL0 register (n = 0 to 3) of each clock monitor is only possible with the protection release sequence described below.

Procedure

(1) Write the fixed value ($A5_H$) to register CLMAnPCMD.

(2) Write the new setting to register CLMAnCTL0. Write the value after a reset to the reserved bits.

(3) Write the bitwise inverse of the setting value to register CLMAnCTL0. Write the inverse of the value after a reset to the reserved bits.

(4) Write the new setting to register CLMAnCTL0. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not released if the procedure is not followed correctly. The setting will not be written to the write-protected register and the CLMAnPS.CLMAnPRERR bit will be set to 1. (You can confirm if the values set in the targeted register was correctly written by checking that the setting of the CLMAnPS.CLMAnPRERR bit is 0 after step 4, although this procedure is not a requirement).

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps (1) to (4) of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the CLMAnPS.CLMAnPRERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

For writing to CLMATEST, see the description of the PROT1PHCMD register.

Note 1. “Another register in the same module” indicates a register under the same module name as that containing the write-protected register. For the module names of the registers, see **Section 29.6.2, List of Registers**.

29.6.6.2 Setting CLMAnCMPL/CLMAnCMPH register

The CLMAnCMPL/CLMAnCMPH registers should be set so that the following conditions are satisfied. If the clock monitor is otherwise used, operation cannot be guaranteed.

- $1 \leq$ CLMAnCMPL
- CLMAnCMPL + 3 $\leq$ CLMAnCMPH (n = 0, 1, 2, 3)

29.7 ECM

The ECM monitors various failure detection states in the LSI chip, and defines the operation to be carried out upon failure detection. For the details of the ECM, see **Section 30, Error Control Module (ECM)**.

Section 30 Error Control Module (ECM)

This section describes an error control module (ECM).

The first part of this section describes all RH850/C1M-A specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of the ECM.

30.1 Features of RH850/C1M-A ECM

30.1.1 Number of Units

This microcontroller has the following number of units of the ECM.

Table 30.1 Number of Units

Product	RH850/C1M-A
Number of Units	1
Name	ECM

30.1.2 Register Base Address

ECM base addresses are listed in the following table.

ECM register addresses are given as offsets from the base addresses in general.

Table 30.2 Register Base Address

Base Address Name	Base Address
ECM master <ECMM_base>	FFCB 0000_H
ECM checker <ECMC_base>	FFCB 1000_H
ECM common part <ECM_base>	FFCB 2000_H

30.1.3 Clock Supply

ECM clock is listed in the following table.

Table 30.3 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
ECM	PCLK	CLK_LSB (Low-speed peripheral clock)

30.1.4 Interrupt and DMA/DTS

ECM interrupt requests are listed in the following table.

Table 30.4 Interrupt Request

Interrupt Name	Interrupt Number	DMA Trigger Number		DTS Trigger Number	
		1st	2nd	1st	2nd
Error control module NMI interrupt	(FEINT)	—	—	—	—
Error control module interrupt	8	—	—	—	—

Note: The ECM does not support DMA or DTS trigger sources.

30.1.5 Reset Sources

ECM reset sources are listed in the following table. ECM is initialized by these reset sources.

However, the ECM master/checker error source status register is initialized only by external reset. For details, see **Section 30.3, Registers (30.3.4, 30.3.5)**.

Table 30.5 Reset Sources

Unit Name	Reset Source
ECM	Reset by all reset sources.

30.1.6 External Input/Output Signals

External input/output signals of ECM are listed in the following table.

Table 30.6 External Input/Output Signals

Unit Signal Name	Description	Pin Signal Name
$\overline{\text{ERROROUT\_M}}$	ERROROUT pin (master)	$\overline{\text{ERROROUT\_M}}$
$\overline{\text{ERROROUT\_C}}$	ERROROUT pin (checker)	$\overline{\text{ERROROUT\_C}}$

30.2 Overview

30.2.1 Function Overview

The error control module (ECM) collects error signals coming from different error sources and monitoring circuits. It also outputs error signals from the ERROROUT pins ($\overline{\text{ERROROUT\_M}}$ and $\overline{\text{ERROROUT\_C}}$) and generates interrupts and internal reset signals. **Table 30.7** lists the functions of ECM.

Table 30.7 List of Functions

Item	Description
Safety processing	ECM can handle the following processing in response to error signal inputs from individual modules. • Error flag set • Maskable interrupt generation Maskable interrupt generation can be controlled (enabled/disabled) for individual errors. • FE level interrupt generation FE level interrupt generation can be controlled (enabled/disabled) for individual errors. • Internal reset generation Internal reset generation can be controlled (enabled/disabled) for individual errors. • ERROROUT output Pin output mask can be controlled (enabled/disabled) for individual errors. Output can be toggled in response to a timer input or made at a fixed level.
Error status	ECM incorporates the ECM master/checker error source status register, which can be used to confirm the error status from the error flag. The error flags are only cleared by an external reset or by writing the corresponding bit of the ECM error source status clear trigger register to 1. In case of an internal reset, the error flags are kept and the reset generation source can be confirmed by reading the ECM master/checker error source status register after reset.
Debug, self-diagnosis	Pseudo errors can be generated for debug and self-diagnosis. The operation during injection of pseudo errors is identical to that for the occurrence of real errors. All configurations for the mask to the ERROROUT output, interrupt, or internal reset apply in the same way. • ECM incorporates a loop-back function of the ERROROUT output that is used to diagnose the path to the ERROROUT pin. The status of the ERROROUT pin is reflected to the ECM master/checker error source status register and can be confirmed by reading the register.
Timeout function	ECM incorporates a function that generates an ERROROUT output or internal reset when the value counted by the delay timer matches with the ECM delay timer compare register because the delay timer was not stopped during the interrupt processing after being started simultaneously with the occurrence of an interrupt request. The value counted by the delay timer is in cycles of the low-speed peripheral clock.
Register protection	A write-protection with a special sequence is incorporated to protect registers from inadvertent write access.
Others	ECM is duplexed. The ERROROUT pin is duplexed to the two pins of the master pin and the checker pin. The ERROROUT outputs from the ECM master and ECM checker are constantly compared. If they do not match, an ECM compare error (error source 26) occurs.

30.2.2 Block Diagram

Figure 30.1 is a block diagram of the ECM.

The ERROROUT output, internal reset, and ERROROUTZ signals are active low, while the maskable and FE level interrupt signals are active high.

Note that the ERROROUTZ signal for PIC1B is not toggled even if the ECM is in dynamic mode.

See **Section 24, Peripheral Interconnection (PIC)** for details on PIC.

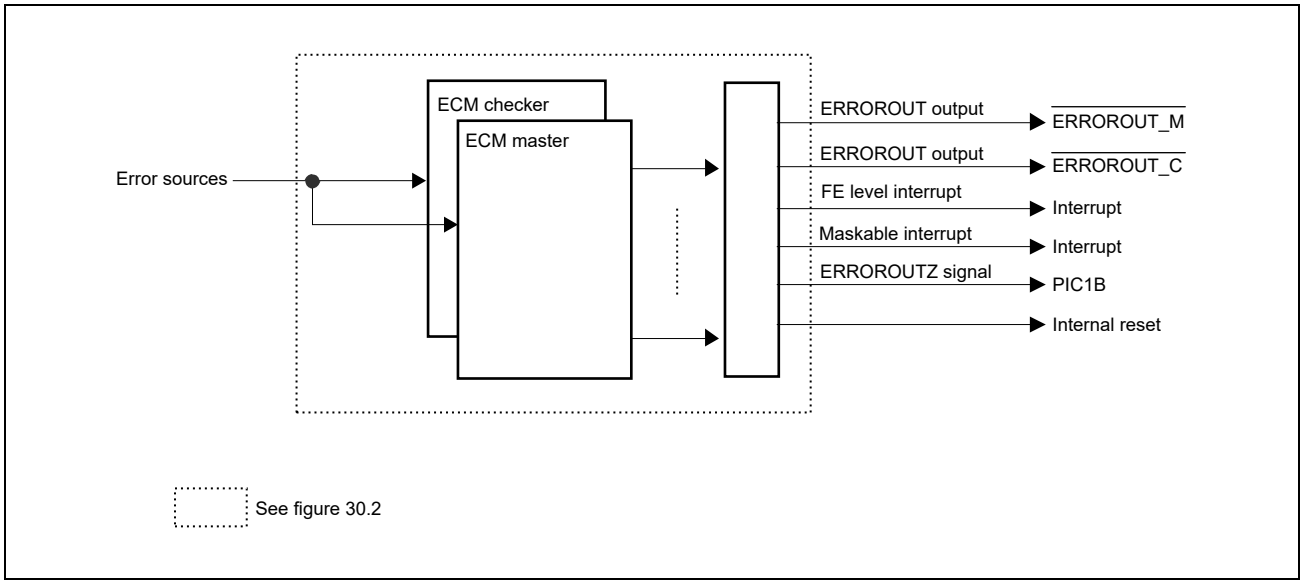


Figure 30.1 Outline of ECM

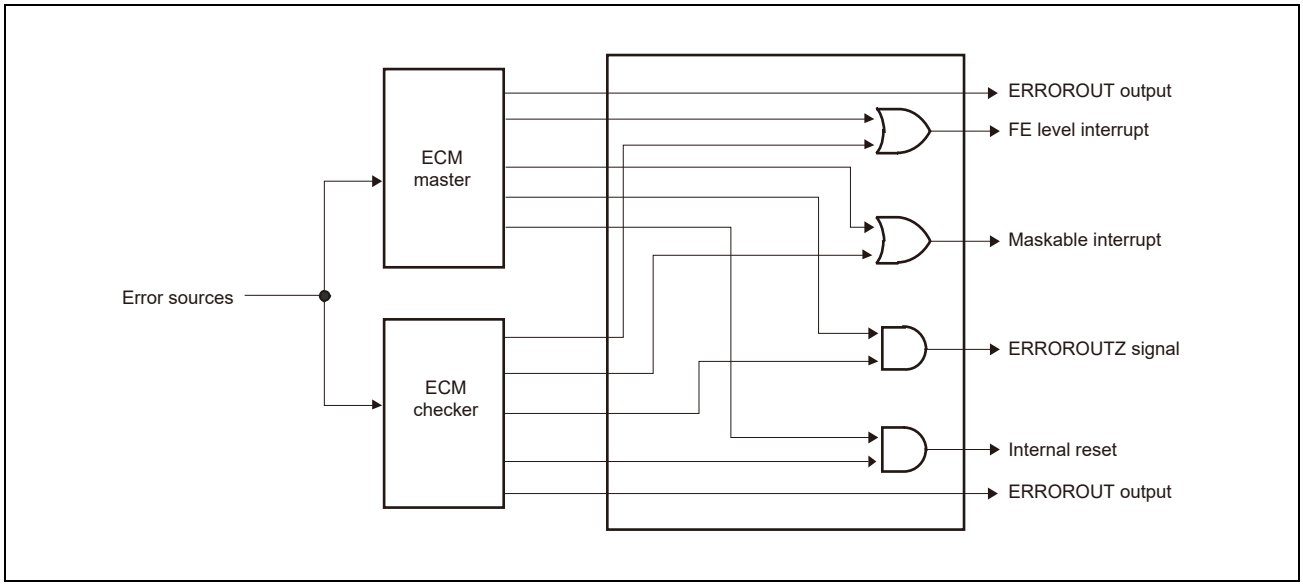


Figure 30.2 Connection of ECM

30.2.3 Error Sources and Safety Processing

Table 30.8 lists the error sources and safety processing of the ECM of RH850/C1M-A.

Table 30.8 List of Error Sources and Safety Processing (1/2) (√: supported,—: not supported)

Error Source No.	Module	Error Source	Error Flag Set	Maskable Interrupt	FE Level Interrupt	Internal reset	ERROR OUT Output	Delay Timer Start
0	WDTA	WDTA error*2	√	√	√	√ *1	√	√
1	SWDT0	SWDT error*2	√	√	√	√ *1	√	√
2	SWDT1	SWDT error*2	√	√	√	√ *1	√	√
3	Reserved		—	—	—	—	—	—
4	Lock step	Lock step compare error*3	√	√	√	√	√	√
5	MISG	MISG compare error*3	√	√	√	√	√	√
6	RAM	Local RAM (CPU1, CPU2) 2-bit ECC error and local RAM (CPU1) address parity error*3	√	√	√	√	√	√
7		Local RAM (CPU1, CPU2) 1-bit ECC error and local RAM (CPU1) parity bit error*3	√	√	√	√	√	√
8		Global RAM 2-bit ECC error and address parity error*3	√	√	√	√	√	√
9		Global RAM 1-bit ECC error*3	√	√	√	√	√	√
10		Peripheral RAM (CAN, CSIH, DTS) 2-bit ECC error*3	√	√	√	√	√	√
11		Peripheral RAM (CAN, CSIH, DTS) 1-bit ECC error*3	√	√	√	√	√	√
12	Code flash	Code flash 2-bit ECC error and address parity error*3	√	√	√	√	√	√
13		Code flash 1-bit ECC error*3	√	√	√	√	√	√
14	Instruction cache	Instruction cache data (CPU1, CPU2) 2-bit ECC error*3	√	√	√	√	√	√
15		Instruction cache data (CPU1, CPU2) 1-bit ECC error*3	√	√	√	√	√	√
16		Instruction cache tag (CPU1, CPU2) 2-bit ECC error*3	√	√	√	√	√	√
17		Instruction cache tag (CPU1, CPU2) 1-bit ECC error*3	√	√	√	√	√	√
18	Data flash	Data flash 2-bit ECC error*3	√	√	√	√	√	√
19		Data flash 1-bit ECC error*3	√	√	√	√	√	√
20	PE guard (PEG)	PEG error (CPU1, CPU2, SubCPU)*4	√	√	√	√	√	√
21	GRG (Global RAM Guard)	GRG error*3	√	√	√	√	√	√
22	PBG	PBG error*3	√	√	√	√	√	√
23	Reserved		—	—	—	—	—	—
24	Reserved		—	—	—	—	—	—
25	Data parity	Data parity error*3	√	√	√	√	√	√
26	ECM	ECM compare error*5	√	√	√	√	√	√

Table 30.8 List of Error Sources and Safety Processing (2/2) (√: supported,—: not supported)

Error Source No.	Module	Error Source	Error Flag Set	Maskable Interrupt	FE Level Interrupt	Internal reset	ERROR OUT Output	Delay Timer Start
27	Clock monitor	Clock monitor error (PLL0 (SSCG))*3	√	√	√	√	√	√
28		Clock monitor error (PLL1 (clean))*3	√	√	√	√	√	√
29		Clock monitor error (WDTA clock (main OSC / LS IntOSC)*3	√	√	√	√	√	√
30		Clock monitor error (LS IntOSC / main OSC)*3	√	√	√	√	√	√
31	SubCPU RAM	SubCPU RAM 2-bit ECC error	√	√	√	√	√	√
32		SubCPU RAM 2-bit ECC error	√	√	√	√	√	√
33	SubCPU code flash	SubCPU code flash 2-bit ECC error and address parity error*3	√	√	√	√	√	√
34		SubCPU code flash 1-bit ECC error*3	√	√	√	√	√	√
35	Reserved		—	—	—	—	—	—
36			—	—	—	—	—	—
37	ADCC	AD parity error*6	√	√	√	√	√	√
38	Flash	Flash access error*7	√	√	√	√	√	√
39		FACI reset transfer error*7,*13	√	—	—	—	√	—
40	EMU3	EMU3 error signal*11	√	√	√	√	√	√
41	DMAC	DTS RAM Data ECC SEC-DED*8	√	√	√	√	√	√
42	Duplex circuit	Error detection using on-chip self-diagnostic (duplex) circuit*3	√	√	√	√	√	√
43	DMAC	DMA violation access notice*8	√	√	√	√	√	√
44	OSTM	OSTM1 interrupt*9	√	√	√	√	√	√
45	OSTM	OSTM3 interrupt*9	√	√	√	√	√	√
46	TSG3	TSG3 error signal (INTTSG3nIER)*10	√	√	√	√	√	√
47	RDC3A	RDC3A error detection signal*12	√	√	√	√	√	√
48 to 60	Reserved		—	—	—	—	—	—
61	ECM	Time out function of the ECM delay timer*5	√	—	—	√	√	—
62		Error set by ECMmESET*5	√	—	—	—	—	—
63		Error output loop back state*5	√	—	—	—	—	—

Note 1. In the initial state, generation of an internal reset is allowed.
Note 2. For the details of this error, see **Section 16, Window Watchdog Timer A (WDTA)**.
Note 3. For the details of this error, see **Section 29, Functional Safety**.
Note 4. For the details of this error, see **Section 3, CPU System**.
Note 5. For the details of this error, see **Table 30.7, List of Functions**.
Note 6. For the details of this error, see **Section 27, A/D Converter (ADCC)**.
Note 7. For the details of the errors No. 38 and No. 39, refer to the *RH850/C1M-A Flash Memory User's Manual: Hardware Interface*.
Note 8. For the details of this error, see **Section 7, DMA Controller**.
Note 9. For the details of this error, see **Section 17, OS Timer (OSTM)**.
Note 10. For the details of this error, see **Section 20, Motor Control Timer (TSG3)**.
Note 11. For the details of this error, see **Section 25, Enhanced Motor Control Unit 3 (EMU3)**.
Note 12. For the details of this error, see **Section 26, R/D Converter (RDC3A)**.
Note 13. Operation of the device is not guaranteed if an FACI reset transfer error has occurred. When clearing the error signal output by the ERROROUT pin after release from the reset state, confirm that an FACI reset transfer error has not occurred by checking that an error state is not being indicated again.

Error sources are merged as described in **Table 30.9**.

Table 30.9 Merging of Error Sources (1/2)

Error Source No.	Module	Error Source	Note
0	WDTA	WDTA error (WDTA0, WDTA1)	Errors of WDTA0 and WDTA1 are merged. In the initial state, generation of an internal reset is allowed.
6	RAM	Local RAM (CPU1, CPU2): 2-bit ECC error and address parity error	2-bit ECC errors and address parity errors of the local RAM for CPU1and CPU2 are merged.
7		Local RAM (CPU1, CPU2): 1-bit ECC error and parity bit error	1-bit ECC errors and parity bit errors of the local RAM for CPU1, CPU2 are merged.
8		Global RAM: 2-bit ECC error and address parity error	2-bit ECC errors and address parity errors when global RAM is accessed from each master are merged.
9		Global RAM: 1-bit ECC error and parity bit error	1-bit ECC errors and parity bit errors when global RAM is accessed from each master are merged.
10		Peripheral RAM (CAN, CSIH, DTS): 2-bit ECC error	2-bit ECC errors of the RAM for peripheral circuits are merged.
11		Peripheral RAM (CAN, CSIH, DTS): 1-bit ECC error	1-bit ECC errors of the RAM for peripheral circuits are merged.
12	Code flash	Code flash 2-bit ECC error Code flash address parity error	2-bit ECC errors when code flash is accessed from CPU1 and CPU2 are merged.
13		Code flash 1-bit ECC error	1-bit ECC errors when code flash is accessed from CPU1 and CPU2 are merged.
14	Instruction cache	Instruction cache data (CPU1, CPU2): 2-bit ECC error	2-bit ECC errors of PE1 and PE2 cache data array are merged.
15		Instruction cache data (CPU1, CPU2): 1-bit ECC error	1-bit ECC errors of PE1 and PE2 cache data array are merged.
16		Instruction cache tag (CPU1, CPU2): 2-bit ECC error	2-bit ECC errors of PE1 and PE2 cache data array are merged.
17		Instruction cache tag (CPU1, CPU2): 1-bit ECC error	1-bit ECC errors of PE1 and PE2 cache tag array are merged.
20	PE guard (PEG)	PEG error (CPU1, CPU2, SubCPU)	PEG errors of CPU1, CPU2, and SubCPU are merged.
21	Global RAM Guard (GRG)	GRG error	GRG errors for each access path of CPU1, CPU2, and DMA are merged.

Table 30.9 Merging of Error Sources (2/2)

Error Source No.	Module	Error Source	Note
22	PBG	PBG error	PBG errors for each peripheral circuit are merged.
25	Data parity	Data parity error	Data parity errors of each access path are merged.
31	SubCPU RAM 2-bit ECC error	Local RAM (SubCPU): 2-bit ECC error and address parity error, Instruction cache data (SubCPU): 2-bit ECC error, Instruction cache tag (SubCPU): 2-bit ECC error	2-bit ECC errors and address parity errors of the local RAM of the SubCPU, 2-bit ECC errors of cache data array, and 2-bit ECC errors of cache tag array are merged.
32	SubCPU RAM 1-bit ECC error	Local RAM (SubCPU): 1-bit ECC error and address parity error, Instruction cache data (SubCPU): 1-bit ECC error, Instruction cache tag (SubCPU): 1-bit ECC error	1-bit ECC errors and address parity errors of local RAM of the SubCPU, 1-bit ECC errors of cache data array, and 1-bit ECC errors of cache tag array are merged.
33	SubCPU code flash	Code flash 2-bit ECC error Code flash address parity error	2-bit ECC errors when code flash is accessed from SubCPU are merged.
34		Code flash 1-bit ECC error	1-bit ECC errors when code flash is accessed from SubCPU are merged.
37	ADCC	AD parity error	AD parity errors or ADCC0, ADCC1, and ADCC2 are merged.
40	EMU3	EMU3 error signal	EMU30 interrupt 7 and EMU31 interrupt 7 are merged.
41	DMA	DTS RAM Data 2-bit and 1-bit ECC errors	2-bit and 1-bit ECC errors of DTS RAM are merged.
42	Duplex	Error detection using on-chip self-diagnostic (duplex) circuit	Errors of each self-diagnostic circuit are merged.
46	TSG3	TSG3 error (INTTSG3nIER)	Errors of TSG30, TSG31, and TSG32 are merged.
47	RDC3A	RDC3A error detection	Errors of RDC3A0 and RDC3A1are merged.

Note: Error sources not listed in **Table 30.9** are not merged.

30.3 Registers

30.3.1 List of Registers

ECM registers are assigned to three address areas: the address area for ECM common registers, the address area for ECM master registers, and the address area for ECM checker registers. The address area for ECM common registers is used for both master and checker registers. Writing to this common area is performed by master and checker registers simultaneously. Reading from the common area reads the master register's value. ECM master and checker registers can be written separately.

The ECM master registers are listed in the following table.

See **Section 30.1.2, Register Base Address** for <ECMM_base>, <ECMC_base>, and <ECM_base>.

Table 30.10 List of Registers (ECM Master)

Module Name	Register Name	Symbol	Protection by Sequence	Address
ECMM	ECM master error set trigger register	ECMMESET	Protected	<ECMM_base> + 00_H
ECMM	ECM master error clear trigger register	ECMMECLR	Protected	<ECMM_base> + 04_H
ECMM	ECM master error source status register 0	ECMMESSTR0	Not protected	<ECMM_base> + 08_H
ECMM	ECM master error source status register 1	ECMMESSTR1	Not protected	<ECMM_base> + $0C_H$
ECMM	ECM master protection command register	ECMMPCMD0	Not protected	<ECMM_base> + 10_H

The ECM checker registers are listed in the following table.

Table 30.11 List of Registers (ECM Checker)

Module Name	Register Name	Symbol	Protection by Sequence	Address
ECMC	ECM checker error set trigger register	ECMCESET	Protected	<ECMC_base> + 00_H
ECMC	ECM checker error clear trigger register	ECMCECLR	Protected	<ECMC_base> + 04_H
ECMC	ECM checker error source status register 0	ECMCESSTR0	Not protected	<ECMC_base> + 08_H
ECMC	ECM checker error source status register 1	ECMCESSTR1	Not protected	<ECMC_base> + $0C_H$
ECMC	ECM checker protection command register	ECMCPCMD0	Not protected	<ECMC_base> + 10_H

The ECM common part registers are listed in the following table.

Table 30.12 List of Registers (ECM Common Part)

Module Name	Register Name	Symbol	Protection by Sequence	Address
ECM	ECM error pulse configuration register	ECMEPCFG	Protected	<ECM_base> + 00_H
ECM	ECM maskable interrupt configuration register 0	ECMMICFG0	Protected	<ECM_base> + 04_H
ECM	ECM maskable interrupt configuration register 1	ECMMICFG1	Protected	<ECM_base> + 08_H
ECM	ECM FE level interrupt configuration register 0	ECMNMICFG0	Protected	<ECM_base> + $0C_H$
ECM	ECM FE level interrupt configuration register 1	ECMNMICFG1	Protected	<ECM_base> + 10_H
ECM	ECM internal reset configuration register 0	ECMIRCFG0	Protected	<ECM_base> + 14_H
ECM	ECM internal reset configuration register 1	ECMIRCFG1	Protected	<ECM_base> + 18_H
ECM	ECM error mask register 0	ECMEMK0	Protected	<ECM_base> + $1C_H$
ECM	ECM error mask register 1	ECMEMK1	Protected	<ECM_base> + 20_H
ECM	ECM error source status clear trigger register 0	ECMESSTC0	Protected	<ECM_base> + 24_H
ECM	ECM error source status clear trigger register 1	ECMESSTC1	Protected	<ECM_base> + 28_H
ECM	ECM protection command register	ECMPCMD1	Not protected	<ECM_base> + $2C_H$
ECM	ECM protection status register	ECMPS	Not protected	<ECM_base> + 30_H
ECM	ECM pseudo error trigger register 0	ECMPE0	Protected	<ECM_base> + 34_H
ECM	ECM pseudo error trigger register 1	ECMPE1	Protected	<ECM_base> + 38_H
ECM	ECM delay timer control register	ECMDTMCTL	Protected	<ECM_base> + $3C_H$
ECM	ECM delay timer register	ECMDTMR	Not protected	<ECM_base> + 40_H
ECM	ECM delay timer compare register	ECMDTMCMP	Protected	<ECM_base> + 44_H
ECM	ECM delay timer configuration register 0	ECMDTMCFG0	Protected	<ECM_base> + 48_H
ECM	ECM delay timer configuration register 1	ECMDTMCFG1	Protected	<ECM_base> + $4C_H$
ECM	ECM delay timer configuration register 2	ECMDTMCFG2	Protected	<ECM_base> + 50_H
ECM	ECM delay timer configuration register 3	ECMDTMCFG3	Protected	<ECM_base> + 54_H

30.3.2 ECMmESET (m = M/C) ─ ECM Master/Checker Error Set Trigger Register

The ECM master/checker error set trigger register is for selecting output of the error signal from the ERROROUT pin. When the ECMmEST bit is set to 1, the ERROROUT pin immediately outputs the error signal. The output cannot be masked. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence. This register is always read as 00_H.

Access: This register can be written in 8-bit units.

Address: <ECMM_base>
<ECMC_base>

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMmEST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 30.13 ECMmESET Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMmEST	Error Set Trigger 0: Writing 0 is invalid 1: Set the output level from the ERROROUT pin to error output.

CAUTIONS

Setting the ERROROUT output from the ERROROUT pin via the ECMmESET register will set the ECMmSSE026 bit of the ECMmESSTR0 register (ECM compare error). Therefore, the ECMmESET register has to be set following the sequence below.

1. Set the ECMEMK026 bit of the ECMEMK0 register to “masked”.
2. Prevent the generation of interrupts by setting the ECMMIE026 bit of the ECMMICFG0 register to “prohibited” and the ECMNMIE026 bit of the ECMNMICFG0 register to “prohibited”.
3. Prevent generation of an internal reset by setting the ECMIRE026 bit of the ECMIRCFG0 register to “prohibited”.
4. Set the ERROROUT output with the ECMmESET register.
5. Clear error flags by setting the ECMCLSSE026 bit of the ECMESSTC0 register.
6. Make the following settings in accord with the condition of usage for the ECM compare error.
   • To allow output of the ERROROUT signal from the ERROROUT pin, set the ECMEMK026 bit of the ECMEMK0 register to “not masked”.
   • To enable generation of interrupts, set the ECMMIE026 bit of the ECMMICFG0 register to “enabled” or the ECMNMIE026 bit of the ECMNMICFG0 register to “enabled”.
   • To enable generation of an internal reset, set the ECMIRE026 bit of the ECMIRCFG0 register to “enabled”.

30.3.3 ECMmECLR (m = M/C) ─ ECM Master/Checker Error Clear Trigger Register

The ECM master/checker error clear trigger register is for setting the error signal from the ERROROUT pin to normal output. When the ECMmECT bit is set to 1, the ERROROUT pin immediately outputs a signal to indicate normal operation as long as no sources lead to setting of the ERROROUT pin to indicate an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence. This register is always read as 00_H.

Access: This register can be written in 8-bit units.

Address: <ECMM_base> + 04_H
<ECMC_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMmECT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 30.14 ECMmECLR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMmECT	Error Clear Trigger 0: Writing 0 is invalid 1: Set the output level from the ERROROUT pin to normal output.

CAUTIONS

Clearing of the ERROROUT output is only possible if all errors, not masked by ECMEMK0/1, are cleared beforehand.

Clearing the ERROROUT output via the ECMmECLR register will set the ECMmSSE026 bit of the ECMmESSTR0 register (ECM compare error). Therefore, the ECMmECLR register has to be set following the sequence below.

1. Set the ECMEMK026 (ECM compare error) bit of the ECMEMK0 register to “masked”.
2. Disable interrupts by setting the ECMMIE026 bit of the ECMMICFG0 register to “disabled” and the ECMNMIE026 bit of the ECMNMICFG0 register to “disabled”.
3. Disable an internal reset by setting the ECMIRE026 bit of the ECMIRCFG0 register to “disabled”.
4. Use the ECMmECLR register to clear the ERROROUT output.
5. Clear error flags by setting the ECMCLSSE026 bit of the ECMESSTC0 register.
6. Make the following settings in accord with the condition of usage for the ECM compare error.
   • To allow output of the ERROROUT signal from the ERROROUT pin, set the ECMEMK026 bit of the ECMEMK0 register to “not masked”.
   • To enable generation of interrupts, set the ECMMIE026 bit of the ECMMICFG0 register to “enabled” or the ECMNMIE026 bit of the ECMNMICFG0 register to “enabled”.
   • To enable generation of an internal reset, set the ECMIRE026 bit of the ECMIRCFG0 register to “enabled”.

30.3.4 ECMmESSTR0 (m = M/C) ─ ECM Master/Checker Error Source Status Register 0

The ECM master/checker error source status register 0 indicates the state of individual error sources, which is irrelevant to the setting of the error mask. The status can be cleared only by an external reset or by writing 1 to the corresponding bit of the ECM error source status clear trigger register 0. An internal reset will not affect this register.

Access: This register can be read in 32-bit units.

Address: <ECMM_base> + 08_H
<ECMC_base> + 08_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMmSSE031	ECMmSSE030	ECMmSSE029	ECMmSSE028	ECMmSSE027	ECMmSSE026	ECMmSSE025	—	—	ECMmSSE022	ECMmSSE021	ECMmSSE020	ECMmSSE019	ECMmSSE018	ECMmSSE017	ECMmSSE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMmSSE015	ECMmSSE014	ECMmSSE013	ECMmSSE012	ECMmSSE011	ECMmSSE010	ECMmSSE009	ECMmSSE008	ECMmSSE007	ECMmSSE006	ECMmSSE005	ECMmSSE004	—	ECMmSSE002	ECMmSSE001	ECMmSSE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 30.15 ECMmESSTR0 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMmSSE031 to ECMmSSE025	Error Source Status ECMmSSE031 to ECMmSSE025 correspond to error sources 31 to 25. 0: Error not occurred 1: Error occurred
24, 23	—	Reserved. When read, the value returned is undefined.
22 to 4	ECMmSSE022 to ECMmSSE004	Error Source Status ECMmSSE022 to ECMmSSE004 correspond to error sources 22 to 4. 0: Error not occurred 1: Error occurred
3	—	Reserved. When read, the value returned is undefined.
2 to 0	ECMmSSE002 to ECMmSSE000	Error Source Status ECMmSSE002 to ECMmSSE000 correspond to error sources 2 to 0. 0: Error not occurred 1: Error occurred

30.3.5 ECMmESSTR1 (m = M/C) ─ ECM Master/Checker Error Source Status Register 1

The ECM master/checker error source status register 1 indicates the state of individual error sources, which is irrelevant to the setting of the error mask. The status can be cleared only by writing 1 to the corresponding bit of the ECM error source status clear trigger register 1 or by an external reset. An internal reset will not affect this register.

Access: This register can be read in 32-bit units.

Address: <ECMM_base> + $0C_H$
<ECMC_base> + $0C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMmSSE131	ECMmSSE130	ECMmSSE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMmSSE115	ECMmSSE114	ECMmSSE113	ECMmSSE112	ECMmSSE111	ECMmSSE110	ECMmSSE109	ECMmSSE108	ECMmSSE107	ECMmSSE106	ECMmSSE105	—	—	ECMmSSE102	ECMmSSE101	ECMmSSE100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 30.16 ECMmESSTR1 Register Contents

Bit Position	Bit Name	Function
31	ECMmSSE131	Indicates the ERROROUT output loopback status. 0: ERROROUT output is error output. 1: ERROROUT output is normal output.
30	ECMmSSE130	Indicates the ECMmESET write status. 0: No error 1: Error is set by the ECMmEST bit of the ECMmESET register
29	ECMmSSE129	Indicates whether delay timer overflow has occurred. 0: Delay timer overflow not occurred 1: Delay timer overflow occurred
28 to 16	—	Reserved. When read, the value returned is undefined.
15 to 5	ECMmSSE115 to ECMmSSE105	Error Source Status ECMmSSE115 to ECMmSSE105 correspond to error sources 47 to 37. 0: Error not occurred 1: Error occurred
4, 3	—	Reserved. When read, the value returned is undefined.
2 to 0	ECMmSSE102 to ECMmSSE100	Error Source Status ECMmSSE102 to ECMmSSE100 correspond to error sources 34 to 32. 0: Error not occurred 1: Error occurred

30.3.6 ECMmPCMD0 (m = M/C) ─ ECM Master/Checker Protection Command Register

The ECM master/checker protection command register protects the write-protected registers against illegal writing due to incorrect operation of the program, etc.

Refer to **Section 30.3.1, List of Registers**, for the registers to be protected by the ECM master/checker protection command register.

Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECMM_base> + 10_H
<ECMC_base> + 10_H

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ECMmREG0[7:0]							
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 30.17 ECMmPCMD0 Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved. When writing to these bits, write 0.
7 to 0	ECMmREG0[7:0]	Protection command that enables writing to write protected ECMm registers. Fixed value 0000 00A5$_H$

30.3.7 ECMEPCFG ─ ECM Error Pulse Configuration Register

The ECM error pulse configuration register sets the type of ERROROUT signal that is output on the ERROROUT pin. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 8-bit units.

Address: <ECM_base>

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMSL0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 30.18 ECMEPCFG Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMSL0	ERROROUT Pin Operation Configuration Setting for the type of ERROROUT signal that is output on the ERROROUT pin 0: Non-dynamic mode 1: Dynamic mode

30.3.8 ECMMICFG0 ─ ECM Maskable Interrupt Configuration Register 0

The ECM maskable interrupt configuration register 0 is used to set the generation of ECM maskable interrupts. The generation of maskable interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 04_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMMIE031	ECMMIE030	ECMMIE029	ECMMIE028	ECMMIE027	ECMMIE026	ECMMIE025	—	—	ECMMIE022	ECMMIE021	ECMMIE020	ECMMIE019	ECMMIE018	ECMMIE017	ECMMIE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMMIE015	ECMMIE014	ECMMIE013	ECMMIE012	ECMMIE011	ECMMIE010	ECMMIE009	ECMMIE008	ECMMIE007	ECMMIE006	ECMMIE005	ECMMIE004	—	ECMMIE002	ECMMIE001	ECMMIE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 30.19 ECMMICFG0 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMMIE031 to ECMMIE025	ECM Maskable Interrupt Generation Control ECMMIE031 to ECMMIE025 correspond to error sources 31 to 25. 0: Interrupt generation disabled 1: Interrupt generation enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMMIE022 to ECMMIE004	ECM Maskable Interrupt Generation Control ECMMIE022 to ECMMIE004 correspond to error sources 22 to 4. 0: Interrupt generation disabled 1: Interrupt generation enabled
3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMMIE002 to ECMMIE000	ECM Maskable Interrupt Generation Control ECMMIE002 to ECMMIE000 correspond to error sources 2 to 0. 0: Interrupt generation disabled 1: Interrupt generation enabled

30.3.9 ECMMICFG1 ─ ECM Maskable Interrupt Configuration Register 1

The ECM maskable interrupt configuration register 1 is used to set the generation of ECM maskable interrupts. The generation of maskable interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 08_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMMIE115	ECMMIE114	ECMMIE113	ECMMIE112	ECMMIE111	ECMMIE110	ECMMIE109	ECMMIE108	—	ECMMIE106	ECMMIE105	—	—	ECMMIE102	ECMMIE101	ECMMIE100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W

Table 30.20 ECMMICFG1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 8	ECMMIE115 to ECMMIE108	ECM Maskable Interrupt Generation Control ECMMIE115 to ECMMIE108 correspond to error sources 47 to 40. 0: Interrupt generation disabled 1: Interrupt generation enabled
7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMMIE106, ECMMIE105	ECM Maskable Interrupt Generation Control ECMMIE106, ECMMIE105 correspond to error sources 38, 37. 0: Interrupt generation disabled 1: Interrupt generation enabled
4, 3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMMIE102 to ECMMIE100	ECM Maskable Interrupt Generation Control ECMMIE102 to ECMMIE100 correspond to error sources 34 to 32. 0: Interrupt generation disabled 1: Interrupt generation enabled

30.3.10 ECMNMICFG0 ─ ECM FE Level Interrupt Configuration Register 0

The ECM FE level interrupt configuration register 0 is used to set the generation of ECM FE level interrupts. The generation of maskable interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + $0C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMNMIE031	ECMNMIE030	ECMNMIE029	ECMNMIE028	ECMNMIE027	ECMNMIE026	ECMNMIE025	—	—	ECMNMIE022	ECMNMIE021	ECMNMIE020	ECMNMIE019	ECMNMIE018	ECMNMIE017	ECMNMIE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMNMIE015	ECMNMIE014	ECMNMIE013	ECMNMIE012	ECMNMIE011	ECMNMIE010	ECMNMIE009	ECMNMIE008	ECMNMIE007	ECMNMIE006	ECMNMIE005	ECMNMIE004	—	ECMNMIE002	ECMNMIE001	ECMNMIE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 30.21 ECMNMICFG0 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMNMIE031 to ECMNMIE025	ECM FE level Interrupt Generation Control ECMNMIE031 to ECMNMIE025 correspond to error sources 31 to 25. 0: Interrupt generation disabled 1: Interrupt generation enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMNMIE022 to ECMNMIE004	ECM FE level Interrupt Generation Control ECMNMIE022 to ECMNMIE004 correspond to error sources 22 to 4. 0: Interrupt generation disabled 1: Interrupt generation enabled
3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMNMIE002 to ECMNMIE000	ECM FE level Interrupt Generation Control ECMNMIE002 to ECMNMIE000 correspond to error sources 2 to 0. 0: Interrupt generation disabled 1: Interrupt generation enabled

30.3.11 ECMNMICFG1 ─ ECM FE Level Interrupt Configuration Register 1

The ECM FE level interrupt configuration register 1 is used to set the generation of FE level interrupt. The generation of FE level interrupt in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 10_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMN MIE115	ECMN MIE114	ECMN MIE113	ECMN MIE112	ECMN MIE111	ECMN MIE110	ECMN MIE109	ECMN MIE108	—	ECMN MIE106	ECMN MIE105	—	—	ECMN MIE102	ECMN MIE101	ECMN MIE100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W

Table 30.22 ECMNMICFG1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 8	ECMNMIE115 to ECMNMIE108	ECM FE level Interrupt Generation Control ECMNMIE115 to ECMNMIE108 correspond to error sources 47 to 40. 0: Interrupt generation disabled 1: Interrupt generation enabled
7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMNMIE106, ECMNMIE105	ECM FE level Interrupt Generation Control ECMNMIE106, ECMNMIE105 correspond to error sources 38, 37. 0: Interrupt generation disabled 1: Interrupt generation enabled
4, 3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMNMIE102 to ECMNMIE100	ECM FE level Interrupt Generation Control ECMNMIE102 to ECMNMIE100 correspond to error sources 34 to 32. 0: Interrupt generation disabled 1: Interrupt generation enabled

30.3.12 ECMIRCFG0 ─ ECM Internal Reset Configuration Register 0

The ECM internal reset configuration register 0 is used to set the generation of an internal reset in response to an internal error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to Section **30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 14_H

Value after reset: 0000 $000F_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMIRE031	ECMIRE030	ECMIRE029	ECMIRE028	ECMIRE027	ECMIRE026	ECMIRE025	—	—	ECMIRE022	ECMIRE021	ECMIRE020	ECMIRE019	ECMIRE018	ECMIRE017	ECMIRE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMIRE015	ECMIRE014	ECMIRE013	ECMIRE012	ECMIRE011	ECMIRE010	ECMIRE009	ECMIRE008	ECMIRE007	ECMIRE006	ECMIRE005	ECMIRE004	—	ECMIRE002	ECMIRE001	ECMIRE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 30.23 ECMIRCFG0 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMIRE031 to ECMIRE025	ECM Internal Reset Generation Control ECMIRE031 to ECMIRE025 correspond to error sources 31 to 25. 0: Internal reset generation disabled 1: Internal reset generation enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMIRE022 to ECMIRE004	ECM Internal Reset Generation Control ECMIRE022 to ECMIRE004 correspond to error sources 22 to 4. 0: Internal reset generation disabled 1: Internal reset generation enabled
3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMIRE002 to ECMIRE000	ECM Internal Reset Generation Control ECMIRE002 to ECMIRE000 corresponds to error sources 2 to 0. 0: Internal reset generation disabled 1: Internal reset generation enabled

30.3.13 ECMIRCFG1 ─ ECM Internal Reset Configuration Register 1

The ECM internal reset configuration register 1 is used to set the generation of an internal reset in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 18_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMIRE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMIRE115	ECMIRE114	ECMIRE113	ECMIRE112	ECMIRE111	ECMIRE110	ECMIRE109	ECMIRE108	—	ECMIRE106	ECMIRE105	—	—	ECMIRE102	ECMIRE101	ECMIRE100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W

Table 30.24 ECMIRCFG1 Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29	ECMIRE129	ECM Internal Reset Generation Control Corresponds to delay timer overflow. 0: Internal reset generation disabled 1: Internal reset generation enabled
28 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 8	ECMIRE115 to ECMIRE108	ECM Internal Reset Generation Control ECMIRE115 to ECMIRE108 correspond to error sources 47 to 40. 0: Internal reset generation disabled 1: Internal reset generation enabled
7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMIRE106, ECMIRE105	ECM Internal Reset Generation Control ECMIRE106, ECMIRE105 correspond to error sources 38, 37. 0: Internal reset generation disabled 1: Internal reset generation enabled
4, 3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

Table 30.24 ECMIRCFG1 Register Contents (2/2)

Bit Position	Bit Name	Function
2 to 0	ECMIRE102 to ECMIRE100	ECM Internal Reset Generation Control ECMIRE102 to ECMIRE100 correspond to error sources 34 to 32. 0: Internal reset generation disabled 1: Internal reset generation enabled

30.3.14 ECMEMK0 – ECM Error Mask Register 0

The ECM error mask register 0 is used to mask the individual error sources of the ERROROUT output. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + $1C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMEMK031	ECMEMK030	ECMEMK029	ECMEMK028	ECMEMK027	ECMEMK026	ECMEMK025	—	—	ECMEMK022	ECMEMK021	ECMEMK020	ECMEMK019	ECMEMK018	ECMEMK017	ECMEMK016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMEMK015	ECMEMK014	ECMEMK013	ECMEMK012	ECMEMK011	ECMEMK010	ECMEMK009	ECMEMK008	ECMEMK007	ECMEMK006	ECMEMK005	ECMEMK004	—	ECMEMK002	ECMEMK001	ECMEMK000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 30.25 ECMEMK0 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMEMK031 to ECMEMK025	ECM ERROROUT Output Mask Control ECMEMK031 to ECMEMK025 correspond to error sources 31 to 25. 0: ERROROUT output not masked 1: ERROROUT output masked
24, 23	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
22 to 4	ECMEMK022 to ECMEMK004	ECM ERROROUT Output Signal Mask Control ECMEMK022 to ECMEMK004 correspond to error sources 22 to 4. 0: ERROROUT output not masked 1: ERROROUT output masked
3	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
2 to 0	ECMEMK002 to ECMEMK000	ECM ERROROUT Output Signal Mask Control ECMEMK002 to ECMEMK000 correspond to error sources 2 to 0. 0: ERROROUT output not masked 1: ERROROUT output masked

30.3.15 ECMEMK1 – ECM Error Mask Register 1

The ECM error mask register 1 is used to mask the individual error sources of the ERROROUT output. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 20_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMEMK129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMEMK115	ECMEMK114	ECMEMK113	ECMEMK112	ECMEMK111	ECMEMK110	ECMEMK109	ECMEMK108	ECMEMK107	ECMEMK106	ECMEMK105	—	—	ECMEMK102	ECMEMK101	ECMEMK100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W

Table 30.26 ECMEMK1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
29	ECMEMK129	Corresponds to delay timer overflow. 0: ERROROUT output not masked 1: ERROROUT output masked
28 to 16	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
15 to 5	ECMEMK115 to ECMEMK105	ECM ERROROUT Output Mask Control ECMEMK115 to ECMEMK105 correspond to error sources 47 to 37. 0: ERROROUT output not masked 1: ERROROUT output masked
4, 3	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
2 to 0	ECMEMK102 to ECMEMK100	ECM ERROROUT Output Mask Control ECMEMK102 to ECMEMK100 correspond to error sources 34 to 32. 0: ERROROUT output not masked 1: ERROROUT output masked

30.3.16 ECMESSTC0 ─ ECM Error Source Status Clear Trigger Register 0

The ECM error source status clear trigger register 0 is used to clear the states of the individual error sources of the ECM master/checker error source status register 0. Both the error states of the ECM master and the ECM checker are cleared simultaneously. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 24_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMCLSSE031	ECMCLSSE030	ECMCLSSE029	ECMCLSSE028	ECMCLSSE027	ECMCLSSE026	ECMCLSSE025	—	—	ECMCLSSE022	ECMCLSSE021	ECMCLSSE020	ECMCLSSE019	ECMCLSSE018	ECMCLSSE017	ECMCLSSE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	R	R	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMCLSSE015	ECMCLSSE014	ECMCLSSE013	ECMCLSSE012	ECMCLSSE011	ECMCLSSE010	ECMCLSSE009	ECMCLSSE008	ECMCLSSE007	ECMCLSSE006	ECMCLSSE005	ECMCLSSE004	—	ECMCLSSE002	ECMCLSSE001	ECMCLSSE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	R	W	W	W

Table 30.27 ECMESSTC0 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMCLSSE031 to ECMCLSSE025	ECM Error Status Clear ECMCLSSE031 to ECMCLSSE025 correspond to error sources 31 to 25. 0: Corresponding error status unchanged 1: Corresponding error status cleared
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMCLSSE022 to ECMCLSSE004	ECM Error Status Clear ECMCLSSE022 to ECMCLSSE004 correspond to error sources 22 to 4. 0: Corresponding error status unchanged 1: Corresponding error status cleared
3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMCLSSE002 to ECMCLSSE000	ECM Error Status Clear ECMCLSSE002 to ECMCLSSE000 correspond to error sources 2 to 0. 0: Corresponding error status unchanged 1: Corresponding error status cleared

30.3.17 ECMESSTC1 ─ ECM Error Source Status Clear Trigger Register 1

The ECM error source status clear trigger register 1 is used to clear the states of the individual error sources of the ECM master/checker error source status register 1. Both the error states of the ECM master and the ECM checker are cleared simultaneously. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 28_H

Value after reset: 00000000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	ECMCLSSE130	ECMCLSSE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMCLSSE115	ECMCLSSE114	ECMCLSSE113	ECMCLSSE112	ECMCLSSE111	ECMCLSSE110	ECMCLSSE109	ECMCLSSE108	ECMCLSSE107	ECMCLSSE106	ECMCLSSE105	—	—	ECMCLSSE102	ECMCLSSE101	ECMCLSSE100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	R	R	W	W	W

Table 30.28 ECMESSTC1 Register Contents

Bit Position	Bit Name	Function
31	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
30, 29	ECMCLSSE130, ECMCLSSE129	ECM Error Status Clear ECMCLSSE130 and ECMCLSSE129 correspond to the write status of the ECMmESET register and delay timer overflow. 0: Corresponding error status unchanged 1: Corresponding error status cleared
28 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 5	ECMCLSSE115 to ECMCLSSE105	ECM Error Status Clear ECMCLSSE115 to ECMCLSSE105 correspond to error sources 47 to 37. 0: Corresponding error status unchanged 1: Corresponding error status cleared
4, 3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMCLSSE102 to ECMCLSSE100	ECM Error Status Clear ECMCLSSE102 to ECMCLSSE100 correspond to error sources 34 to 32. 0: Corresponding error status unchanged 1: Corresponding error status cleared

30.3.18 ECMPCMD1 ─ ECM Protection Command Register

Refer to **Section 30.3.1, List of Registers**, for the registers to be protected by the ECM protection command register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + $2C_H$

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ECMREG1[7:0]							
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 30.29 ECMPCMD1 Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
7 to 0	ECMREG17 to ECMREG10	Protection command that enables writing to write protected ECM registers. Fixed value 0000 $00A5_H$

30.3.19 ECMPS ─ ECM Protection Status Register

The ECM protection status register indicates whether the write protected register has been written successfully or not. For details, refer to **Section 30.4.5, Write Protected Registers**.

Access: This register can be read in 8-bit units.

Address: <ECM_base> + 30_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMPRERR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 30.30 ECMPS Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When read, the value after reset is returned.
0	ECMPRERR	ECM Protection Status Indicates whether the write protected register has been written successfully or not. 0: Writing was successfully completed. 1: Writing failed

30.3.20 ECMPE0 ─ ECM Pseudo Error Trigger Register 0

The ECM pseudo error trigger register 0 is used to generate a pseudo error for test purposes. The ECM operation in response to the generation of a pseudo error is identical to that in response to a real error source. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 34_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMPE031	ECMPE030	ECMPE029	ECMPE028	ECMPE027	ECMPE026	ECMPE025	—	—	ECMPE022	ECMPE021	ECMPE020	ECMPE019	ECMPE018	ECMPE017	ECMPE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	R	R	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMPE015	ECMPE014	ECMPE013	ECMPE012	ECMPE011	ECMPE010	ECMPE009	ECMPE008	ECMPE007	ECMPE006	ECMPE005	ECMPE004	—	ECMPE002	ECMPE001	ECMPE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	R	W	W	W

Table 30.31 ECMPE0 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMPE031 to ECMPE025	ECM Pseudo Error Trigger ECMPE031 to ECMPE025 correspond to error sources 31 to 25. 0: Pseudo error not generated 1: Generates corresponding pseudo error
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMPE022 to ECMPE004	ECM Pseudo Error Trigger ECMPE022 to ECMPE004 correspond to error sources 22 to 4. 0: Pseudo error not generated 1: Generates corresponding pseudo error
3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMPE002 to ECMPE000	ECM Pseudo Error Trigger ECMPE002 to ECMPE000 correspond to error sources 2 to 0. 0: Pseudo error not generated 1: Generates corresponding pseudo error

30.3.21 ECMPE1 – ECM Pseudo Error Trigger Register 1

The ECM pseudo error trigger register 1 is used to generate a pseudo error for test purposes. The ECM operation in response to the generation of a pseudo error is identical to that in response to a real error source. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 38_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMPE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMPE115	ECMPE114	ECMPE113	ECMPE112	ECMPE111	ECMPE110	ECMPE109	ECMPE108	ECMPE107	ECMPE106	ECMPE105	—	—	ECMPE102	ECMPE101	ECMPE100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	R	R	W	W	W

Table 30.32 ECMPE1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29	ECMPE129	ECM Pseudo Error Trigger Corresponds to delay timer overflow. 0: Pseudo error not generated 1: Generates corresponding pseudo error
28 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 5	ECMPE115 to ECMPE105	ECM Pseudo Error Trigger ECMPE115 to ECMPE105 correspond to error sources 47 to 37. 0: Pseudo error not generated 1: Generates corresponding pseudo error
4, 3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMPE102 to ECMPE100	ECM Pseudo Error Trigger ECMPE102 to ECMPE100 correspond to error sources 34 to 32. 0: Pseudo error not generated 1: Generates corresponding pseudo error

30.3.22 ECMDTMCTL – ECM Delay Timer Control Register

The ECM delay timer control register is used to control the delay timer. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 8-bit units.

Address: <ECM_base> + $3C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ECMSTP	ECMSTA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W

Table 30.33 ECMDTMCTL Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
1	ECMSTP	Delay Timer Stop Writing 1 to this bit initializes the delay timer counter, causing the delay timer to stop. The ECMSTA bit is set to 0 at the same time.
0	ECMSTA	Delay Timer Start Specifies the operation of the delay timer when an interrupt is occurred. Writing 1 to this bit initializes the delay timer counter, causing the timer to start operation. Writing 0 to this bit initializes the delay timer counter, causing the timer to stop. 0: Timer stops 1: Timer starts

30.3.23 ECMDTMR ─ ECM Delay Timer Register

The ECM delay timer register indicates the value of the delay timer counter. Writing 1 to the ECMSTP bit of the ECM delay timer control register (ECMDTMCTL) or writing 0 to the ECMSTA bit initializes the delay timer counter.

Access: This register can be read in 16-bit units.

Address: <ECM_base> + 40_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMDTMR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

30.3.24 ECMDTMCMP ─ ECM Delay Timer Compare Register

When the value of the ECM delay timer compare register matches with the value of the delay timer counter, a delay timer overflow signal is generated to set the ECMmSSE129n bit. Writing data to this register has to be conducted while the delay timer is stopped. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 16-bit units.

Address: <ECM_base> + 44_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMDTMCMP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

30.3.25 ECMDTMCFG0 ─ ECM Delay Timer Configuration Register 0

The ECM delay timer configuration register 0 is used to enable or disable the delay timer to be started by a maskable interrupt in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 48_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMTE031	ECMTE030	ECMTE029	ECMTE028	ECMTE027	ECMTE026	ECMTE025	—	—	ECMTE022	ECMTE021	ECMTE020	ECMTE019	ECMTE018	ECMTE017	ECMTE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE015	ECMTE014	ECMTE013	ECMTE012	ECMTE011	ECMTE010	ECMTE009	ECMTE008	ECMTE007	ECMTE006	ECMTE005	ECMTE004	—	ECMTE002	ECMTE001	ECMTE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 30.34 ECMDTMCFG0 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMTE031 to ECMTE025	ECM Delay Timer Start Control ECMTE031 to ECMTE025 correspond to maskable interrupts generated by error sources 31 to 25. 0: Delay timer start disabled 1: Delay timer start enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMTE022 to ECMTE004	ECM Delay Timer Start Control ECMTE022 to ECMTE004 correspond to maskable interrupts generated by error sources 22 to 4. 0: Delay timer start disabled 1: Delay timer start enabled
3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMTE002 to ECMTE000	ECM Delay Timer Start Control ECMTE002 to ECMTE000 correspond to the maskable interrupts generated by error sources 2 to 0. 0: Delay timer start disabled 1: Delay timer start enabled

30.3.26 ECMDTMCFG1 ─ ECM Delay Timer Configuration Register 1

The ECM delay timer configuration register 1 is used to enable or disable the delay timer to be started by a maskable interrupt in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + $4C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE115	ECMTE114	ECMTE113	ECMTE112	ECMTE111	ECMTE110	ECMTE109	ECMTE108	—	ECMTE106	ECMTE105	—	—	ECMTE102	ECMTE101	ECMTE100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W

Table 30.35 ECMDTMCFG1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 8	ECMTE115 to ECMTE108	ECM Delay Timer Start Control ECMTE115 to ECMTE108 correspond to maskable interrupts generated by error sources 47 to 40. 0: Delay timer start disabled 1: Delay timer start enabled
7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMTE106, ECMTE105	ECM Delay Timer Start Control ECMTE106, ECMTE105 correspond to maskable interrupts generated by error sources 38, 37. 0: Delay timer start disabled 1: Delay timer start enabled
4, 3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMTE102 to ECMTE100	ECM Delay Timer Start Control ECMTE102 to ECMTE100 correspond to maskable interrupts generated by error sources 34 to 32. 0: Delay timer start disabled 1: Delay timer start enabled

30.3.27 ECMDTMCFG2 ─ ECM Delay Timer Configuration Register 2

The ECM delay timer configuration register 2 is used to enable or disable the delay timer to be started by an FE level interrupt in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 50_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMTE231	ECMTE230	ECMTE229	ECMTE228	ECMTE227	ECMTE226	ECMTE225	—	—	ECMTE222	ECMTE221	ECMTE220	ECMTE219	ECMTE218	ECMTE217	ECMTE216
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE215	ECMTE214	ECMTE213	ECMTE212	ECMTE211	ECMTE210	ECMTE209	ECMTE208	ECMTE207	ECMTE206	ECMTE205	ECMTE204	—	ECMTE202	ECMTE201	ECMTE200
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 30.36 ECMDTMCFG2 Register Contents

Bit Position	Bit Name	Function
31 to 25	ECMTE231 to ECMTE225	ECM Delay Timer Start Control ECMTE231 to ECMTE225 correspond to FE level interrupts generated by error sources 31 to 25. 0: Delay timer start disabled 1: Delay timer start enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMTE222 to ECMTE204	ECM Delay Timer Start Control ECMTE222 to ECMTE204 correspond to FE level interrupts generated by error sources 22 to 4. 0: Delay timer start disabled 1: Delay timer start enabled
3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMTE202 to ECMTE200	ECM Delay Timer Start Control ECMTE202 to ECMTE200 correspond to FE level interrupts generated by error sources 2 to 0. 0: Delay timer start disabled 1: Delay timer start enabled

30.3.28 ECMDTMCFG3 ─ ECM Delay Timer Configuration Register 3

The ECM delay timer configuration register 3 is used to enable or disable the delay timer to be started by an FE level interrupt in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 54_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE315	ECMTE314	ECMTE313	ECMTE312	ECMTE311	ECMTE310	ECMTE309	ECMTE308	—	ECMTE306	ECMTE305	—	—	ECMTE302	ECMTE301	ECMTE300
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W

Table 30.37 ECMDTMCFG3 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 8	ECMTE315 to ECMTE308	ECM Delay Timer Start Control ECMTE315 to ECMTE308 correspond to FE level interrupts generated by error sources 47 to 40. 0: Delay timer start disabled 1: Delay timer start enabled
7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMTE306, ECMTE305	ECM Delay Timer Start Control ECMTE306, ECMTE305 correspond to FE level interrupts generated by error sources 38, 37. 0: Delay timer start disabled 1: Delay timer start enabled
4, 3	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	ECMTE302 to ECMTE300	ECM Delay Timer Start Control ECMTE302 to ECMTE300 correspond to FE level interrupts generated by error sources 34 to 32. 0: Delay timer start disabled 1: Delay timer start enabled

30.4 Functions

30.4.1 Operations for ERROROUT Output

After release from the reset state, the output on the $\overline{\text{ERROROUT\_M}}$ pin indicates an error.

Use this pin after clearing its indication of error by setting the ECMmECT bit to 1, which is described in **Section 30.3.3, ECMmECLR (m = M/C) ─ ECM Master/Checker Error Clear Trigger Register**. Since the $\overline{\text{ERROROUT\_C}}$ pin is a multi-functional pin that can be used also as a general-purpose I/O pin or for other functions, select the $\overline{\text{ERROROUT\_C}}$ function before using this function. Refer to **Section 2, Pins**, for how to configure the port.

The ERROROUT output signal can be configured for two different modes of operation, non-dynamic or dynamic. Furthermore, when the ERROROUT output is selected and the signal is in dynamic mode, output on the pin indicates errors on occurrence of error sources regardless of the pulse cycle.

Table 30.38 ERROROUT Output Operation

Error State ECMmSSE031 to ECMmSSE000 ECMmSSE115 to ECMmSSE100	Operating Mode ECMSL0 Bit	ERROROUT Output Operating Mode	ERROROUT Output Level
0	0	Non-dynamic	High level
(Normal)	1	Dynamic	Toggles*1 (according to timer input*2)
1	0	Non-dynamic	Low level
(Error)	1	Dynamic	Low level

Note 1. The ERROROUTZ signal for PIC1B is not toggled.

Note 2. For details, see **Section 17, OS Timer (OSTM)**.

30.4.1.1 Dynamic Mode

Enabling Dynamic Mode

1. Initialize OSTM0.
2. Set the ERROROUT output to normal output by setting the ECMmECT bit (m = M/C) in the ECM master/checker error clear trigger register to 1.
3. Set the ECMSL0 bit in the ECM error pulse configuration register to 1 to specify dynamic mode.
4. Start up OSTM0.

30.4.1.2 Non-Dynamic Mode

Disabling Dynamic Mode

1. Set the ERROROUT output to error output by setting the ECMmEST bit (m = M/C) in the ECM master/checker error set trigger register to 1.
2. Stop OSTM0.
3. Clear the ECMSL0 bit in the ECM error pulse configuration register to 0 to specify non-dynamic mode.

30.4.2 Loop-Back Function

ECM incorporates a loop-back function that is used to check the path to the ERROROUT pin. The output level of the ERROROUT pin can be checked with the ECMmSSE131 bit (m = M/C) in the ECM master/checker error source status register 1.

30.4.3 Pseudo Error Generation

ECM incorporates a function that can generate pseudo errors for test or debug purposes. The operation of the ECM during injection of pseudo errors is identical to that for the occurrence of real errors. All configurations for error masks, interrupt, internal reset, or delay timer apply to the operation of ECM.

30.4.4 Error State

The error state is indicated by the ECM master/checker error source status register 0 and the ECM master/checker error source status register 1. The error state is only cleared by an external reset or by writing 1 to the corresponding bit of the ECM error source status clear trigger register. In case of an internal reset, the error state is kept and the error of the reset source can be confirmed by reading the ECM master/checker error source status register 0 and the ECM master/checker error source status register 1 after release from the reset state.

30.4.5 Write Protected Registers

Write protected registers are protected from inadvertent write access due to erroneous program execution, etc.

30.4.5.1 Sequence of Writing to the Write-Protected Registers

Writing to the write-protected registers is only possible with the protection unlock sequence described below.

1. Write the fixed value (0000 00A5$_H$) to the ECM protection command register ECMPCMD1 and ECM master/checker protection command register ECMmPCMD0.
2. Write the new setting to the write-protected registers in the ECM and ECMm. Write the value after a reset*[1] to the reserved bits.
3. Write the bitwise inverse of the setting value to the same registers as in step 2. Write the inverse of the value after a reset*[1] to the reserved bits.
4. Write the new setting to the same registers as in step 2. Write the value after a reset*[1] to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Note 1. For the reserved bits of the ECMEMK0 and ECMEMK1 registers, write 1 in steps 2 and 4 and 0 in step 3.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the ECMPRERR bit of the ECM protection status register ECMPS is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the ECMPRERR bit of the ECM protection status register ECMPS is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register*[2] in the same module, writing to the protected register fails and the protection status bit in the protection status register is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 2. As to ECM, all registers listed in **Section 30.3.1, List of Registers** are treated as “another register in the same module”.

30.4.6 Timeout Function for Interrupt Processing

The ECM incorporates a function that generates an ERROROUT output or internal reset when the count value of the delay timer incorporated in the ECM matches with the value of the ECM delay timer compare register because the delay timer was not stopped during the interrupt processing after being started simultaneously with the occurrence of an interrupt request. The timer counting is not stopped when a debugger break occurs.

The counting of the delay timer always starts from 0 whenever an error occurred. Configure the duration until an internal reset or the ERROROUT output is generated with the settings of the ECM delay timer compare register (ECMDTMCMP) described in **Section 30.3.24, ECMDTMCMP ─ ECM Delay Timer Compare Register**.

If another error by which the delay timer is to be started occurs while the delay timer is running, the current value counted by the delay timer is not reset but the timer continues to run.

Section 31 Data CRC (DCRA)

This section contains a generic description of the data CRC function A (DCRA).

The first section describes all properties specific to the RH850/C1M-A, such as units, register base addresses, input/output signal names, etc.
The remainder of the section describes the functions and registers of the DCRA.

31.1 Features of RH850/C1M-A DCRA

31.1.1 Number of Units

This microcontroller has the following number of units of the DCRA.

Table 31.1 Number of Units

Product Name	RH850/C1M-A
Units	2
Name	DCRAn (n = 0 to 1)

Table 31.2 Index

Index	Meaning
n	Throughout this section, the individual channels of a DCRA is identified by the index “n” (n = 0 to 1), for example, DCRAnCTL for the DCRAn control register.

31.1.2 Register Base Addresses

DCRA base addresses are listed in the following table.

DCRA register addresses are given as offsets from the individual base address.

Table 31.3 Register Base Addresses

Base Address Name	Base Address
<DCRA0_base>	FFF7 0000_H
<DCRA1_base>	FF97 1000_H

31.1.3 Clock Supply

The DCRA clock supply is listed in the following table.

Table 31.4 DCRAn Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
DCRAn	PCLK	CLK_LSB (low-speed peripheral clock)

31.1.4 Reset Source

The DCRA reset sources are shown below. DCRA is initialized by the reset sources below.

Table 31.5 Reset Source

Unit Name	Reset Source
DCRAn	Reset by any reset source

31.2 Overview

31.2.1 Functional Overview

The data CRC function A can be used to verify or generate CRC protected data streams of arbitrary length and different bit widths.

- 32-bit Ethernet CRC
  $(X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X^1 + 1)$
- 16-bit CCITT CRC
  $(X^{16} + X^{12} + X^5 + 1)$
- CRC of an arbitrary data block length can be generated.
- After initialization of the CRC data register, every write access to the CRC input register generates a new CRC according to the selected polynomial, and the result is stored in the CRC data register.

31.2.2 Block Diagram

The following picture shows the block diagram of the data CRC function A.

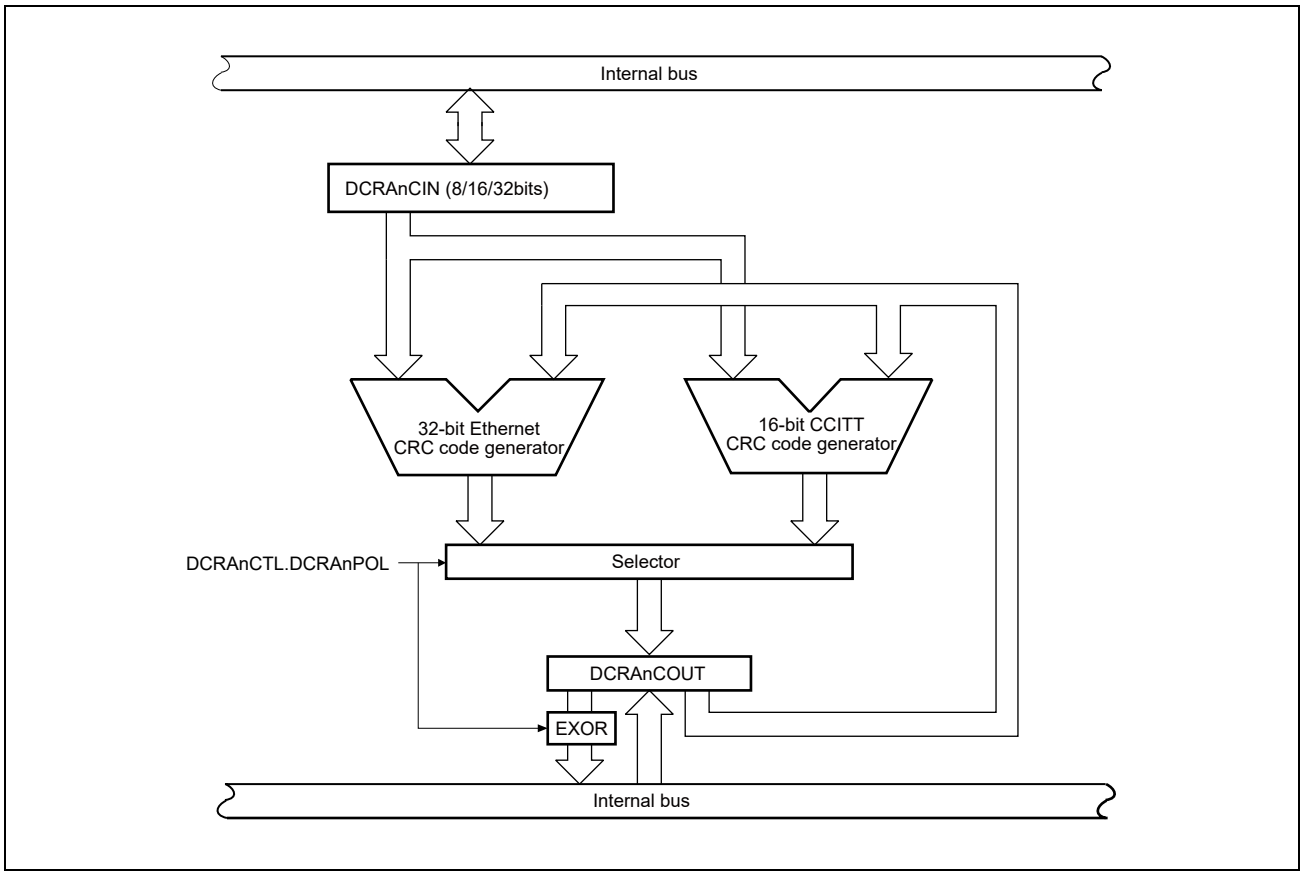
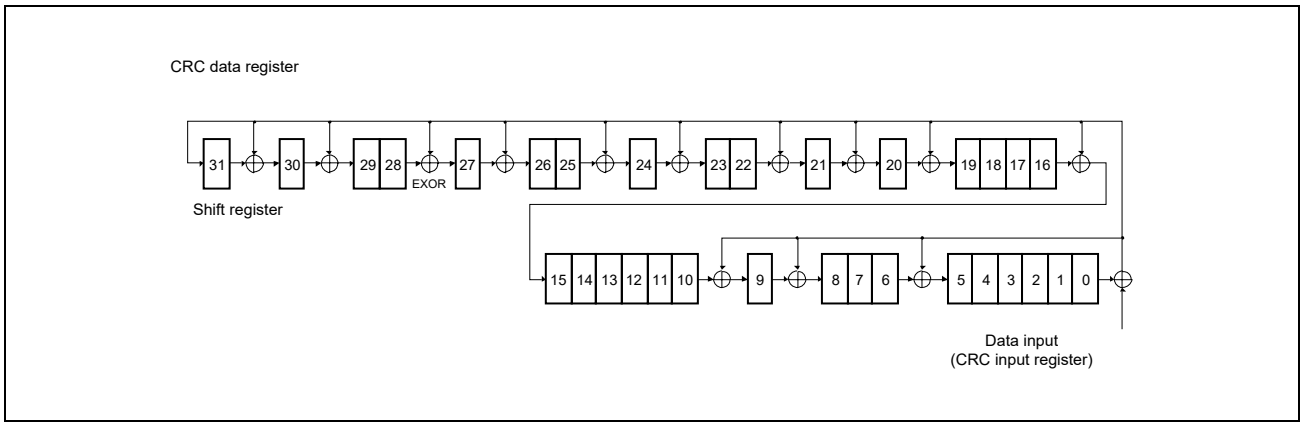


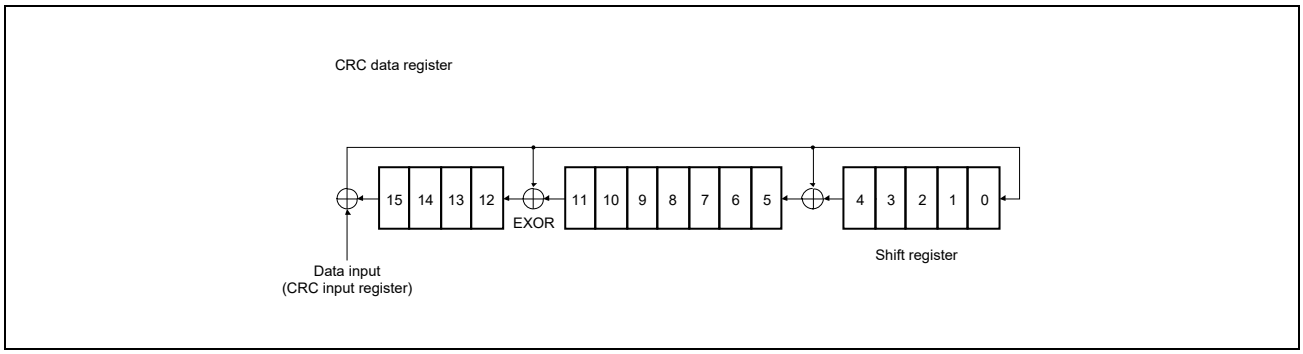
Figure 31.1 Block Diagram of Data CRC Function A

31.2.3 Operation Circuit

- 32-bit Ethernet



- 16-bit CCITT



31.3 Registers

31.3.1 List of Registers

The DCRA registers are listed in the following table.

See **Section 31.1.2, Register Base Addresses** for <DCRAn_base>.

Table 31.6 List of Registers

Module Name	Register Name	Symbol	Address
DCRAn	CRC input register	DCRAnCIN	<DCRAn_base> + 00_H
DCRAn	CRC data register	DCRAnCOUT	<DCRAn_base> + 04_H
DCRAn	CRC control register	DCRAnCTL	<DCRAn_base> + 20_H

31.3.2 DCRAnCIN — CRC Input Register

This register holds the input data for the CRC calculation. The effective bit width used for CRC calculation must be set by DCRAnCTL.DCRAnISZ[1:0].

When data is written to this register, the CRC code is generated.

The CRC calculation is immediately started after the DCRAnCIN register is written. The DCRAnCOUT register must be initialized, with the initial start value, before the first data of the data block is written to DCRAnCIN register.

Access: This register can be read/written in 32-bit units.

Address: <DCRAn_base>

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCRAnCIN[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCRAnCIN[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 31.7 DCRAnCIN Register Contents

Bit Position	Bit Name	Function
31 to 0	DCRAnCIN [31:0]	Input Data for CRC Calculation The valid bits are: • for 32 bits, the effective bit width: DCRAnCIN[31:0] • for 16 bits, the effective bit width: DCRAnCIN[15:0] • for 8 bits, the effective bit width: DCRAnCIN[7:0]

31.3.3 DCRAnCOUT – CRC Data Register

This register stores the result of the CRC code generated by the 32-bit Ethernet or 16-bit CCITT polynomial.

Access: This register can be read/written in 32-bit units.

Address: <DCRAn_base> + 4_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCRAnCOUT[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCRAnCOUT[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. The read value after reset is 0000 0000_H since the 32-bit Ethernet polynomial is selected as the CRC generating function after reset.

Table 31.8 DCRAnCOUT Register Contents

Bit Position	Bit Name	Function
31 to 0	DCRAnCOUT [31:0]	Result of the CRC Code Generation When the 16-bit CCITT polynomial is enabled, bits 15 to 0 show the CRC result. Bits 31 to 16 are undefined. Read value of this register results in Ex-ORed with the value below. • Using 32-bit Ethernet polynomial: FFFF $FFFF_H$ • Using 16-bit CCITT polynomial: 0000_H For example, AAAA $AAAA_H$ is read when DCRAnCOUT = 5555 5555_H by the 32-bit Ethernet polynomial.

CAUTION

This register must be initialized (the initial start value must be set) before the first data of the data block is written to the DCRAnCIN register.

31.3.4 DCRAnCTL ─ CRC Control Register

This register controls the CRC generation process.

Access: This register can be read/written in 8-bit units.

Address: <DCRAn_base> + 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DCRAnISZ[1:0]		DCRAnPOL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 31.9 DCRAnCTL Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2, 1	DCRAnISZ[1:0]	Specify the CRC input bit width: 00: 32 bits (DCRAnCIN[31:0]) 01: 16 bits (DCRAnCIN[15:0]) 10: 8 bits (DCRAnCIN[7:0]) 11: Setting prohibited
0	DCRAnPOL	Specifies the CRC generating function: 0: 32-bit Ethernet CRC polynomial generation. The byte order of the DCRAnCIN register is LSB (least significant bit) first. This means that, if the input bit width is 8 bits (DCRAnISZ[1:0]= 10_B), bit positions 7 to 0 of the DCRAnCIN register contain the input data and bit position 0 (LSB) is the start bit of the input data. 1: 16-bit CCITT CRC polynomial generation. The byte order of the DCRAnCIN register is MSB (most significant bit) first. This means that, if the input bit width is 8 bits (DCRAnISZ[1:0]= 10_B), bit positions 7 to 0 of the DCRAnCIN register contain the input data and bit position 7 (MSB) is the start bit of the input data.

CAUTIONS

- After changing the CRC generating function (DCRAnCTL.DCRAnPOL), the DCRAnCOUT register must be initialized (the initial start value must be set).
- The CRC input bit width (DCRAnCTL.DCRAnISZ[1:0]) must be set according to the data block bit width. Switching the CRC bit input width is not allowed during processing of a data block (a data block consists of N bytes, half words or words). After the final CRC result is read from the DCRAnCOUT register, the CRC input bit width can be changed. In this case, the DCRAnCOUT register must be initialized (the initial start value must be set) afterwards.

31.4 Function

The data CRC function A generates a CRC (cyclic redundancy check) of an arbitrary data block length. The data is forwarded to the data CRC function in 8-, 16- or 32-bit units. The CRC polynomial can either be selected for 32-bit Ethernet or 16-bit CCITT.

The initial start value must be set at the DCRAnCOUT register before the first write access to the CRC input register (DCRAnCIN) is performed.

The flowchart below shows the CRC generating procedure.

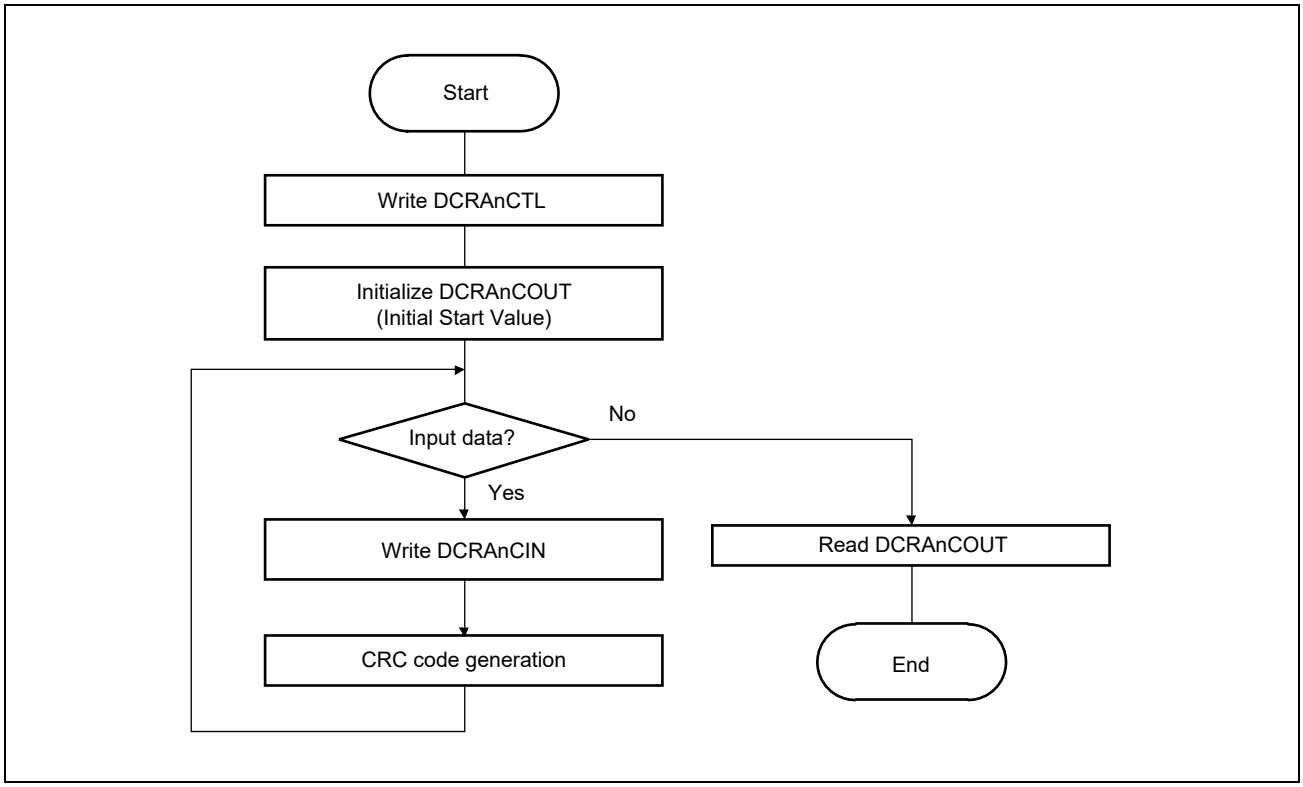


Figure 31.2 Flowchart of Data CRC Function A

NOTES

1. Before writing the first data to DCRAnCIN, the CRC output register DCRAnCOUT must be initialized (the initial start value must be set).
2. DCRAnCOUT must be re-initialized by setting the initial start value when the polynomial is changed by changing DCRAnCTL.DCRAnPOL.
3. Setting example of the initial start values of the respective polynomials
   The following is the example of setting values.

Table 31.10 Setting Example of Initial Start Values (when read at a reset)

	Initial Start Value	EXOR Value	DCRAnCOUT Read Value
32-bit Ethernet	FFFF FFFF$_H$	FFFF FFFF$_H$	0000 0000$_H$
16-bit CCITT	XXXX FFFF$_H$	XXXX 0000$_H$	XXXX FFFF$_H$

Section 32 Intelligent Cryptographic Unit E (ICUSE)

A separate document is provided for ICUSE (intelligent cryptographic unit (slave type) E). Please contact our sales representative for details.

Section 33 Secure Watchdog Timer A (SWDTA)

A separate document is provided for SWDTA (secure watchdog timer A). Please contact our sales representative for details.

Section 34 On-Chip Debugging Unit (OCD)

34.1 Debug Function

This microcontroller has an on-chip debug function. By using the on-chip debug emulator, programs can be debugged with the microcontroller mounted in the target system.

CAUTION

The debug functions described in this section are supported by the microcontroller but whether they are usable depends on the debugger. For details on debugging, see the user's manual of the debugger you are using.

(1) Debug interface

This microcontroller supports the NEXUS JTAG Interface, and low-pin debug interface (4 pins) (hereinafter referred to as 4-pin LPD) as debug interfaces.This microcontroller also incorporates the AUD-RAM monitor that supports monitoring and tuning of the on-chip RAM, data peripheral registers, etc by the AUD-RAM monitor interfaces. For the AUD-RAM monitor, see **Section 34.4, AUD-RAM Monitor (AUDR)**.

(2) Debug monitor function

In debug mode, the monitoring program is executed in the debugging-only area.

The basic debug functions below can be used by running a monitoring program.

- Downloading user programs
- Reading and writing user resources including the memory and registers while the user program is suspended
- Running the user program starting at any address

(3) On-chip break function

12 break points are included in each CPU. Four of them can be designated for any access (access address and access data).

(4) Software break function

A software break point can be designated for any address.

(5) Forced break function

Execution of the user program can be forcibly suspended.

(6) Forced reset function

The microcontroller (this product) can be forcibly reset.

(7) Real-time RAM monitoring (RRM)

The memory can be read during program execution. Because this read access uses debug-dedicated DMA, it has minimal effect on program execution.

(8) Dynamic memory modification (DMM)

The memory can be written during program execution. Because this write access uses debug-dedicated DMA, it has minimal effect on program execution.

(9) Timer function

Using a 32-bit counter, the time for running the user program can be measured based on the clock for debug.

(10) Mask function

A reset factor (external reset, software reset, and ECM reset) can be masked.

(11) Event detection function

Events can be detected by the following: execution address, access address, access data, range (comparison in size), and sequential execution.

(12) Hot plug-in function

Debugging can be started in normal operating mode without an input of an external reset.

(13) Security function

To prevent the contents of the flash memory from being read by an unauthorized person, a 128-bit ID code (OCD_ID) can be written to the microcontroller. If the code the user inputs when starting a debugger does not match the ID code written to the microcontroller, the flash memory cannot be accessed.

(14) Tracing function

Execution history, data changes, etc. of the user program can be obtained. For details, see **Section 34.2, Trace Control Function**.

(15) Multi Core debug function

The following functions are supported as the multi-core debugger for CPU1, CPU2*[1], and the SubCPU:

- Synchronization functions (including reset, execution, and break)
- Synchronous setting
- Simultaneous tracing for multiple cores.

Note 1. C1M-A1 does not have CPU2.

34.2 Trace Control Function

This product provides several trace functions including branch PC trace of the CPU, data trace, and DMA data trace.

(1) Trace RAM

This product has 32 Kbytes of the trace RAM.

The trace information in the trace RAM is accessible via the debug interface: NEXUS, 4-pin LPD.

(2) Software Trace

This function enables the obtaining of history of user program execution, data changes, etc.

The software trace information can be output via the debug interface 4-pin LPD.

34.3 Peripheral Break Control

34.3.1 Overview

The peripheral break function stops the peripheral modules if the user program is stopped, for instance upon a breakpoint hit.

The on-chip modules can be classified into two by its operation at the time of peripheral break as follows:

(1) Modules that are unconditionally stopped:

WDTA0, WDTA1*1, SWDTA0, SWDTA1*1

(2) Modules that can select abeyance or continuation*2:

OSTM0, OSTM1, OSTM2, OSTM3, TAPA0, TAPA1, TAPA2*1, TAPA3, TAPA4, TAPA5, TSG30, TSG31, TSG32, CSIH0, CSIH1, CSIH2, TAUJ0, TAUJ1*1, TAUD0, TAUD1, TAUD2, TAUD3*1, ENCA0, ENCA1, TPBA0, TPBA1*1

Note 1. This is not provided for C1M-A1.

Note 2. TAPAn forcibly places the output pins of TAUDn and TSG3n into Hi-Z when its function is stopped by peripheral break. The value of the TAPAnHOF[10:8] bits of the TAPAnFLG register becomes 111_B. The applicable output pins are as follow:
TAPAnUN, TAPAnUP, TAPAnVN, TAPAnVP, TAPAnWN, TAPAnWP, and TSG3nO1 to TSG3nO6

34.4 AUD-RAM Monitor (AUDR)

34.4.1 Overview

This product includes the AUD (Advanced User Debugger) -RAM monitor (AUDR) to support debugging of a user program under conditions as if it were actually mounted in the system. The AUDR is a function to read and write the resources mapped to memory spaces including an on-chip memory and a peripheral register during LSI operation.

Table 34.1 lists the outlines of the AUDR and **Figure 34.1** shows the block diagram of the AUDR.

Table 34.1 Outlines of the AUDR

Item	Outline
Transfer method	Clock-synchronous parallel interface (4 bits)
Transfer clock generation	Transfer clock is generated at the external host (RAM monitor tool) side.
Transfer clock frequency	Maximum 20 MHz
Access area	Physical address area on the system bus
Access data size	8, 16, 32, and 64 bits
Access address input bit width	8, 16, 24, and 32 bits The same values as those of the previous access address are used for the upper bits of access address, which are not input.
Data transfer method	• Single transfer: Single data is transferred to the access address that has been input. • Continuous transfer: Up to 16 data are continuously transferred from the access address that has been input. The access address is automatically incremented depending on data size.
I/O pin	7 pins ($\overline{\text{AUDRST}}$, AUDCK, $\overline{\text{AUDSYNC}}$, and AUDATA3 to AUDATA0)
Function	• RAM monitor function: Read and write are performed from a system bus to an accessible physical address area. This function enables reference to and modification of an on-chip memory, a peripheral register, and the like. • Configuration information retention (startup communication) function: The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained. This function is used for communication with a RAM monitor tool. • Synchronization communication (message board) function: This flag register is used for communication of the firmware operated by the CPU with a RAM monitor tool.

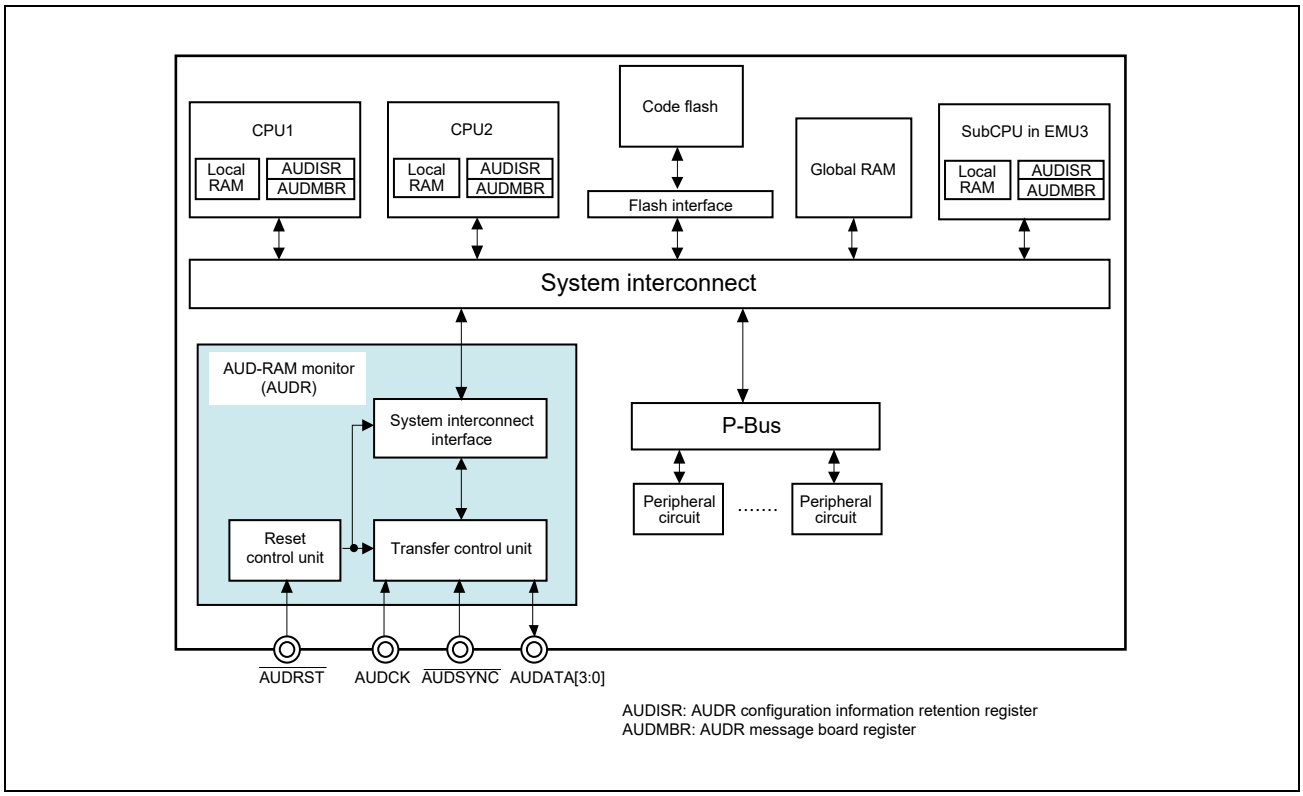


Figure 34.1 Block Diagram of the AUDR

34.4.2 I/O Pins

Table 34.2 lists the I/O pins of the AUDR.

Table 34.2 I/O Pins of AUDR

Pin Name	I/O	Description
$\overline{\text{AUDRST}}$	Input	AUDR reset input pin Inputting L to this pin resets the AUDR, but it does not initialize the AUDISR, AUDMBR and AUDMBRC (described below). When this pin is not connected, it is internally pulled-down.
AUDCK	Input	External clock input pin The frequency of clock that can be input is 20 MHz or under. When this pin is not connected, it is internally pulled-up.
$\overline{\text{AUDSYNC}}$	Input	Timing control signal input pin L: A command, an address, and write data are input and the status flag is output. H: The read-out data is output, and the pin enters into an idle state. When this pin is not connected, it is internally pulled-up. **CAUTION:** This pin should not be negated (to a high level) until the ready state is entered by inputting a command or the like from the outside to the AUDATA. For details, see the protocol mentioned below.
AUDATA3 to AUDATA0	Input/Output	4-bit parallel data I/O pin. The following information is input and output by time division. • Command (input) • Address (input) • Write data (input)/Read data (output) • Status flag (output) When this pin is not connected, it is internally pulled-up.

34.4.3 Description of Registers

Table 34.3 lists the registers related to the AUDR.

Table 34.3 List of Registers

Register Name	Abbreviation	Value after Reset	R/W	Address	Access Size	Reference Section
AUDR configuration information retention register	AUDISR	$000X_H$*1	R	FA00 5000_H (CPU)*4 F900 5000_H (AUDR)*4 F800 5000_H (AUDR)*4 F700 5000_H (AUDR)*4	16*5	**34.4.3.1**
AUDR message board register	AUDMBR	0000_H	R/W*2	FA00 5004_H (CPU)*4 F900 5004_H (AUDR)*4 F800 5004_H (AUDR)*4 F700 5004_H (AUDR)*4	16*5	**34.4.3.2**
	AUDMBRC		R/W*2, *3	FA00 5008_H (CPU)*4 F900 5008_H (AUDR)*4 F800 5008_H (AUDR)*4 F700 5008_H (AUDR)*4	16*5	

Note 1. The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained in bits 3 to 0.

Note 2. Only 1 can be written to the bits that have been set to 0. 0 cannot be written to the bits that have been set to 1.

Note 3. All bits are cleared to 0 after read.

Note 4. Different addresses are used for the AUDISR and AUDMBR/AUDMBRC bits when they are accessed from the CPU and when they are accessed from the AUDR. Use FA00 $500X_H$ for an access from the CPU. Use F900 $500X_H$ for an access from the AUDR to the CPU1 registers. Use F800 $500X_H$ for an access from the AUDR to the CPU2 registers. Use F700 $500X_H$ for an access from AUDR to the SubCPU registers in the EMU3.
Although these addresses are “reserved area”, it is capable of accessing to these addresses for communication with AUDR tool.

Note 5. If an access in data sizes other than 16 bits (half-word) is performed, correct operation is not guaranteed.

34.4.3.1 AUDISR — AUDR Configuration Information Retention Register

AUDISR is a 16-bit readable register. It can be read out from the CPU and the AUDR.

Use FA00 5000_H for an access from the CPU to the AUDISR.

Use F900 5000_H for an access from the AUDR to the AUDISR of CPU1.

Use F800 5000_H for an access from the AUDR to the AUDISR of CPU2.

Use F700 5000_H for an access from the AUDR to the AUDISR of the SubCPU in EMU3.

The AUDISR register is not initialized by AUDR reset. The AUDISR can be also read out from the CPU during AUDR reset.

This register is used in the way described below.

- Retention of configuration information
  On release from an internal reset, the register retains the values corresponding to the levels on pins AUDATA3 to AUDATA0. Setting the levels on pins AUDATA3 to AUDATA0 in the emulator configuration allows the CPU to judge the connection configuration of the emulator.
  If nothing is connected to the AUD RAM monitoring-related pins, pins AUDATA3 to AUDATA0 are pulled up and the register is read as $000F_H$. If an AUD RAM monitoring tool is connected, the levels on pins AUDATA3 to AUDATA0 will correspond to a value other than $000F_H$. Reading of the AUDISR by the CPU can then be used to judge that a tool is connected.
  In addition, the values from pins AUDATA3 to AUDATA0 can be used to identify the vendor that is the source of the connected tool.

For details, see **Table 34.3, List of Registers** about access size, address and value after reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	DATA			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	Values of the AUDATA3 to AUDATA0 pins			
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 34.4 AUDISR Register Contents

Bit Position	Bit Name	Function
15 to 4	—	Reserved These bits are always read as 0. When writing, always write 0.
3 to 0	DATA	The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained.

34.4.3.2 AUDMBR/AUDMBRC — AUDR Message Board Register

The AUDMBR/AUDMBRC registers are 16-bit readable/writable registers. The registers can be read and written from the CPU and the AUDR tool.

Use FA00 5004_H or FA00 5008_H for an access from the CPU to the AUDMBR/AUDMBRC. Use F900 5004_H or F900 5008_H for an access from the AUDR to the AUDMBR/AUDMBRC of CPU1. Use F800 5004_H or F800 5008_H for an access from the AUDR to the AUDMBR/AUDMBRC of CPU2. Use F700 5004_H or F700 5008_H for an access from the AUDR to the AUDMBR/AUDMBRC of the SubCPU in EMU3.

When the CPU reads the AUDMBRC from FA00 5008_H, all bits of the AUDMBR/AUDMBRC are cleared to 0 after reading. When the CPU reads the AUDMBR from FA00 5004_H, the bits of the AUDMBR/AUDMBRC are not cleared.

When the AUDR reads the AUDMBR of CPU1 from F900 5008_H, all bits of the AUDMBR/AUDMBRC of CPU1 are cleared to 0 after reading. When the AUDR tool reads the AUDMBR of CPU1 from F900 5004_H, the bits of the AUDMBR/AUDMBRC of CPU1 are not cleared. It is the same about the AUDMBR/AUDMBRC of CPU2 and the SubCPU in EMU3.

The CPU and the AUDR can write to the AUDMBR/AUDMBRC from the above addresses. However, writing 0 to the bits that have been set to 1 will be ignored (they can only be set to 1).

Table 34.6 summarizes the recommended accesses to the AUDMBR/AUDMBRC.

The AUDMBR/AUDMBRC registers are not initialized by AUDR reset. The AUDMBR/AUDMBRC can be also read and written from the CPU during AUDR reset.

AUDMBR and AUDMBRC are used in the way described below.

- Synchronous communications (message board) function
  The firmware (program) run by the CPU uses AUDMBR/AUDMBRC as a flag register for use in communications with the emulator, which reads AUDMBR/AUDMBRC by using the RAM monitoring function. This allows the emulator to follow the state of firmware (program) operations.

For details, see **Table 34.3, List of Registers** about access size, address and value after reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AUDMBR															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 34.5 AUDMBR/AUDMBRC Register Contents

Bit Position	Bit Name	Function
15 to 0	AUDMBR	Communication flag between the AUDR and the CPU

Table 34.6 Recommended Access to the AUDMBR/AUDMBRC

Accessing Master	Address	R/W	Access	Remarks
CPU	FA00 5004_H (AUDMBR)	Write	Only 1 can be written. Writing 0 is ignored.	—
		Read	Reading is enabled.	Not cleared after reading.
AUDR	AUDMBR (CPU1): F900 5008_H AUDMBR (CPU2): F800 5008_H AUDMBR (SubCPU in EMU3): F700 5008_H (AUDMBRC)	Write	Only 1 can be written. Writing 0 is ignored.	—
		Read	Reading is enabled.	All bits are cleared to 0 after reading.

34.4.4 RAM Monitoring

34.4.4.1 Communication Protocol

Input a command, counter value, address, and data to the AUDATA pin in the format shown in **Figure 34.2**. For details, see **Section 34.4.4.2, Operation**.

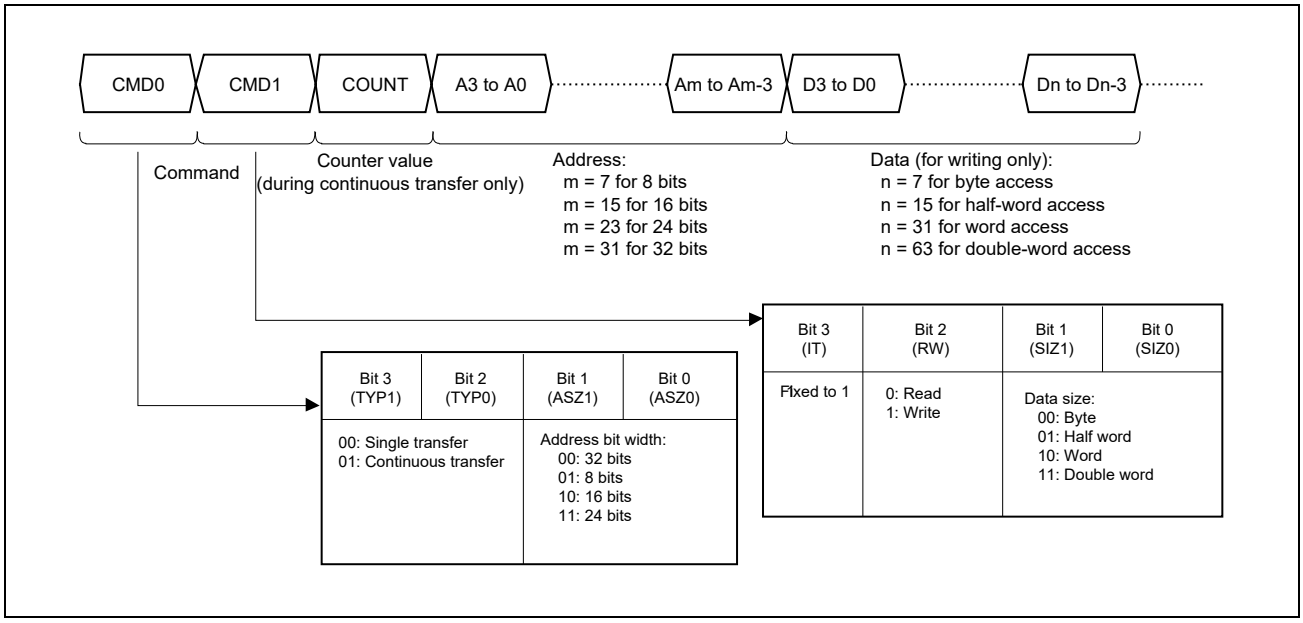


Figure 34.2 AUDATA Pin Input Format

34.4.4.2 Operation

(1) Single transfer

Single transfer is a method of transmitting single data to an address that has been input for access. **Figure 34.3** shows an example of reading operation during single transfer and **Figure 34.4** shows an example of writing operation during single transfer.

When the $\overline{\text{AUDSYNC}}$ pin is asserted and a command, address and data (only for writing) are input to the AUDATA pin in the format shown in **Figure 34.2**, the AUDR starts reading or writing operation for the specified address. The AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this internal execution. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin (see **Figure 34.3** and **Figure 34.4**).

For reading, after the Ready flag is output and the $\overline{\text{AUDSYNC}}$ pin is negated, the read data is output from the AUDATA pin (see **Figure 34.3**).

After output of the read data and until input of the next command, at least one AUDCK cycle is required for switching of the input/output states of the pins.

In addition, after the $\overline{\text{AUDSYNC}}$ pin is negated following the completion of processing for write access, when the next command is input, at least one AUDCK cycle is required for switching of the input/output states of the pins. Therefore, the $\overline{\text{AUDSYNC}}$ pin is negated for two cycles of AUDCK.

If a command other than those shown in **Figure 34.2** is input to CMD0 and CMD1, the AUDR disables this processing as a command error and sets the CFLG bit in the Ready flag to 1. When a bus error occurs during the internal execution, the AUDR disables this processing and sets the BFLG bit in the Ready flag to 1 (see **Figure 34.5**).

When an error is detected, the read data is not output.

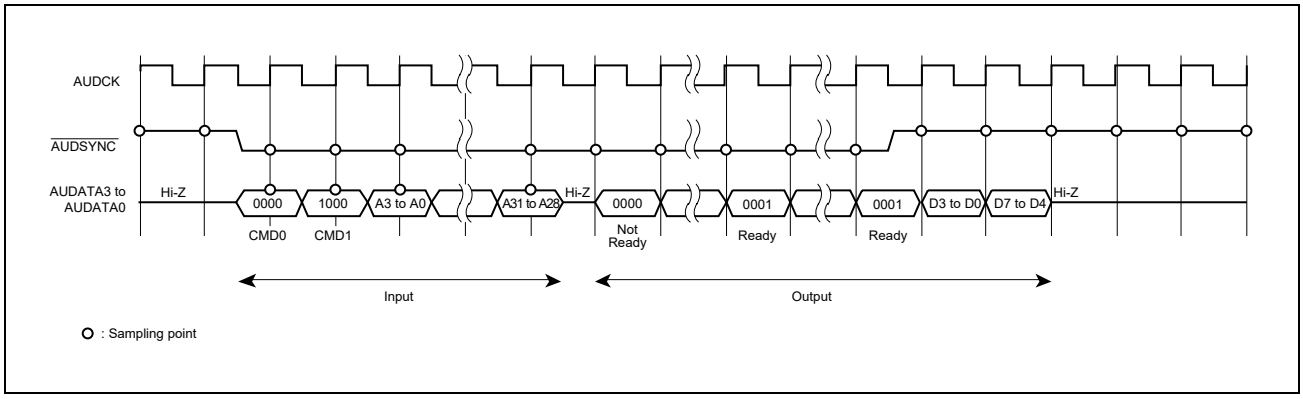


Figure 34.3 Example of Reading Operation in Single Transfer (address 32 bits; byte read)

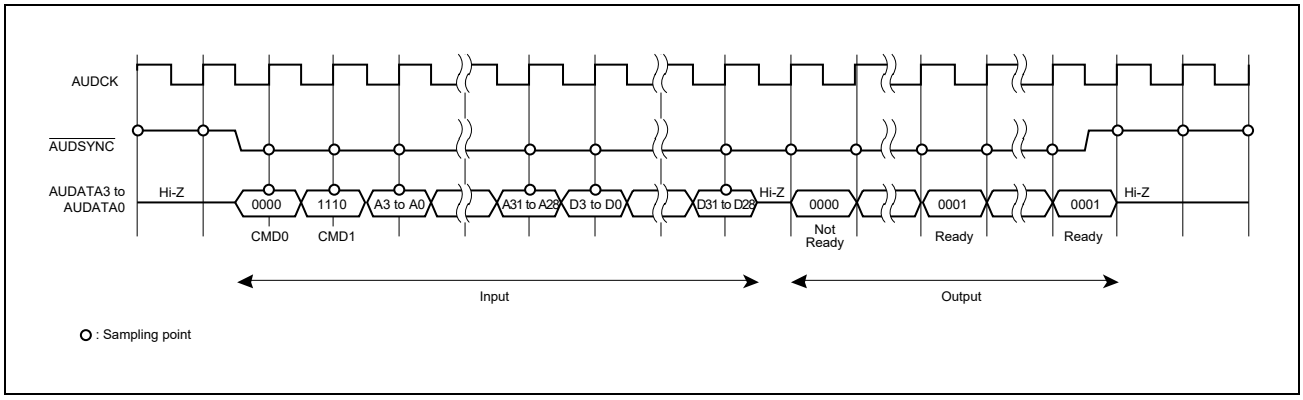


Figure 34.4 Example of Writing Operation in Single Transfer (address 32 bits; word write)

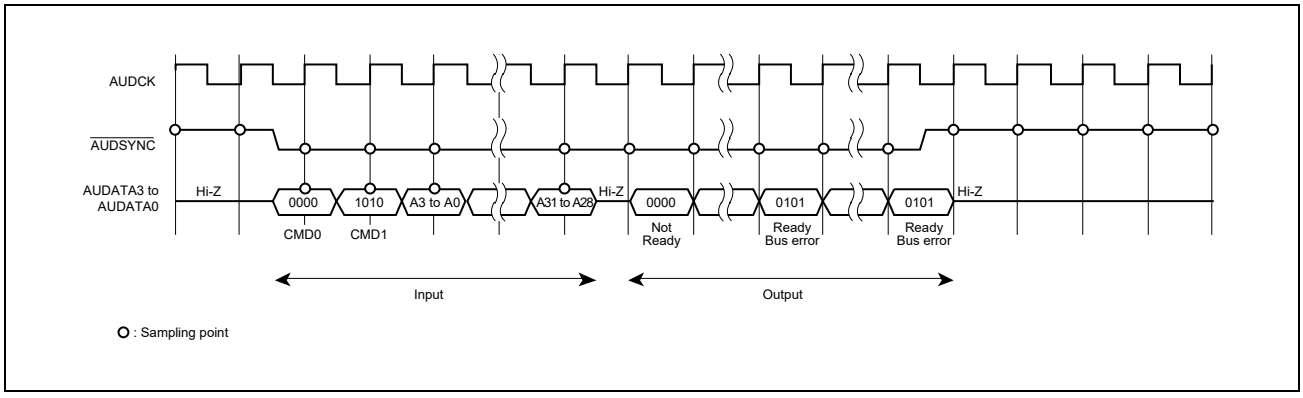


Figure 34.5 Example of Error Occurrence in Single Transfer (address 32 bits; word read)

(2) Continuous transfer

Continuous transfer is a continuous transmitting method of up to 16 data to the address that has been input. The address to be accessed is automatically incremented depending on the data size after completion of each data transfer. **Figure 34.6** shows an example of reading operation during continuous transfer and **Figure 34.7** shows an example of writing operation during continuous transfer.

The first data transfer is equivalent to single transfer except that COUNT (counter value) shown in **Figure 34.2** should be input. The number of data to be transferred minus 1 is input to the COUNT.

In the second and subsequent data transfer, the input of CMD0, CMD1, COUNT is skipped, and addresses are skipped to input. The second and subsequent data transfer for reading and writing proceed as follows.

For reading, when the $\overline{\text{AUDSYNC}}$ pin is asserted after completion of the previous data transfer, the AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this internal execution. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin. Then after the $\overline{\text{AUDSYNC}}$ pin is negated, the read data is output from the AUDATA pin. This operation is repeated until the number of data specified in COUNT is read (see **Figure 34.6**).

For writing, when the $\overline{\text{AUDSYNC}}$ pin is asserted after completion of the previous data transfer, the AUDR inputs data to be written to the AUDATA pin. After that, the AUDR starts the internal execution. The AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this operation. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin. This operation is repeated until the number of data specified in COUNT are written (see **Figure 34.7**).

When the $\overline{\text{AUDSYNC}}$ pin is re-asserted to input the next write data, at least two cycles of AUDCK are required for negating the $\overline{\text{AUDSYNC}}$ pin.

If a bus error occurs during the internal execution, the AUDR suspends data transfer operation and sets the BFLG bit in the Ready flag to 1 (see **Figure 34.8**).

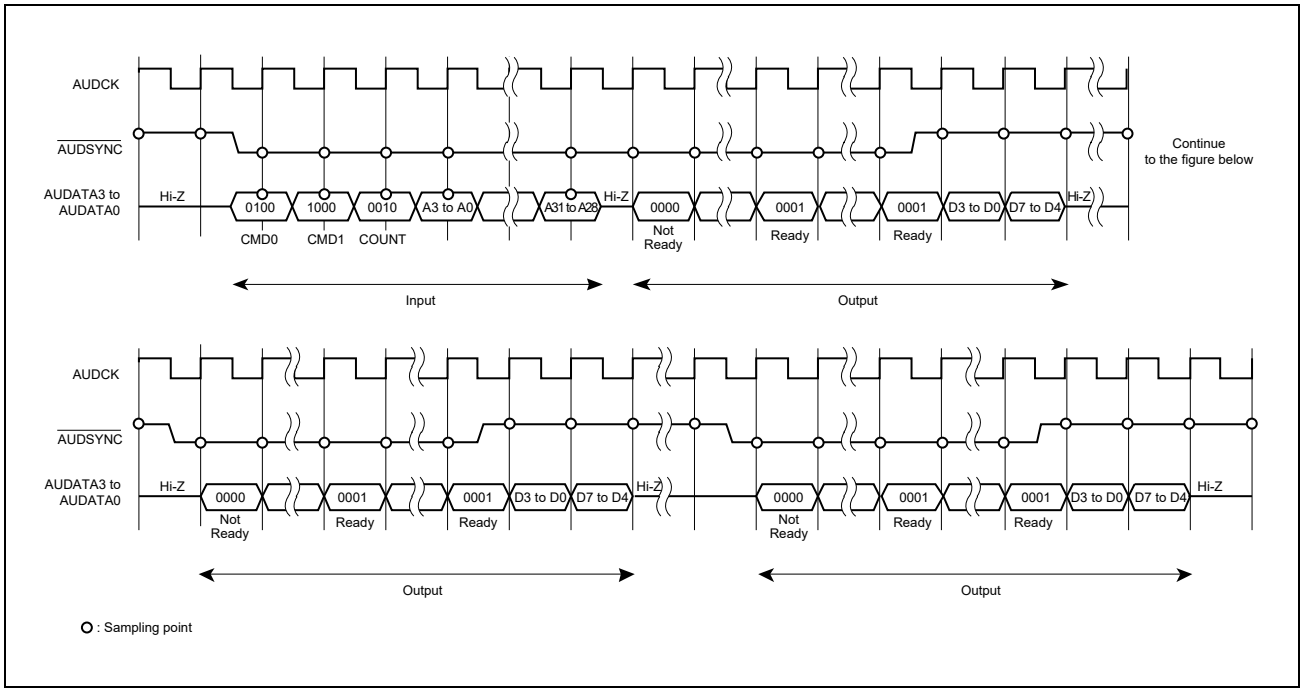


Figure 34.6 Example of Reading Operation in Continuous Transfer (address 32 bits; byte read × three units of data)

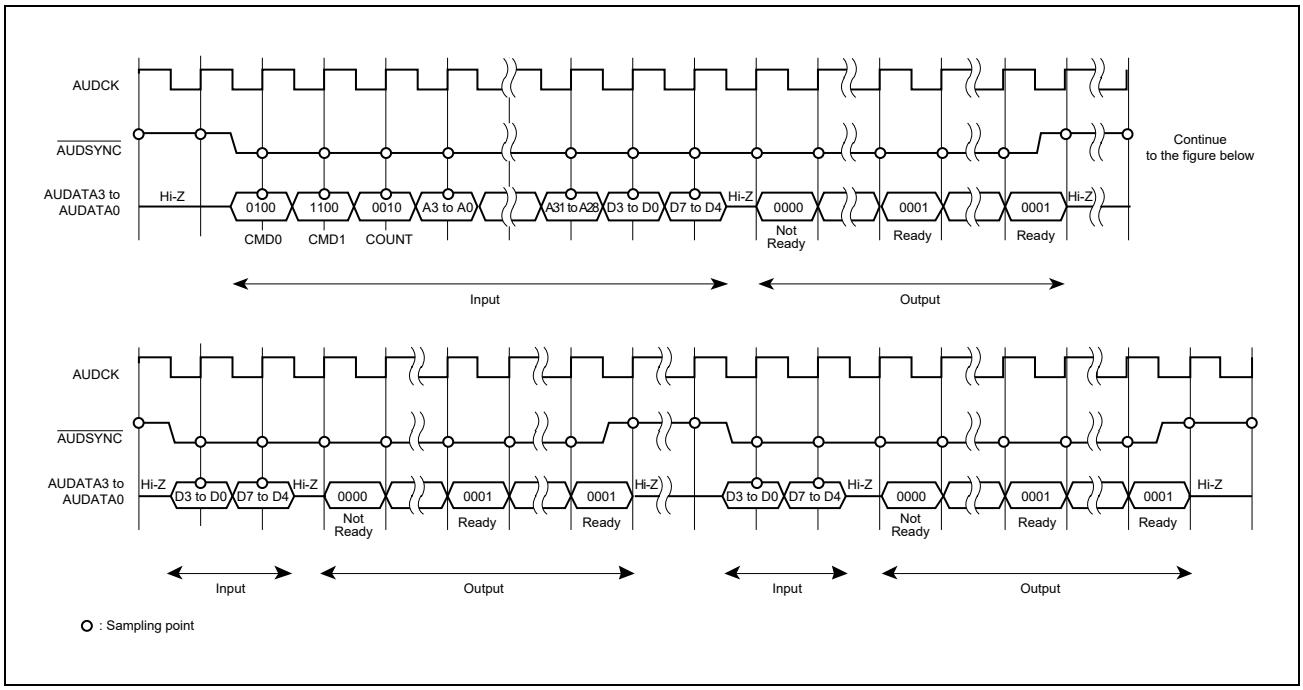


Figure 34.7 Example of Writing Operation in Continuous Transfer (address 32 bits; byte write × three units of data)

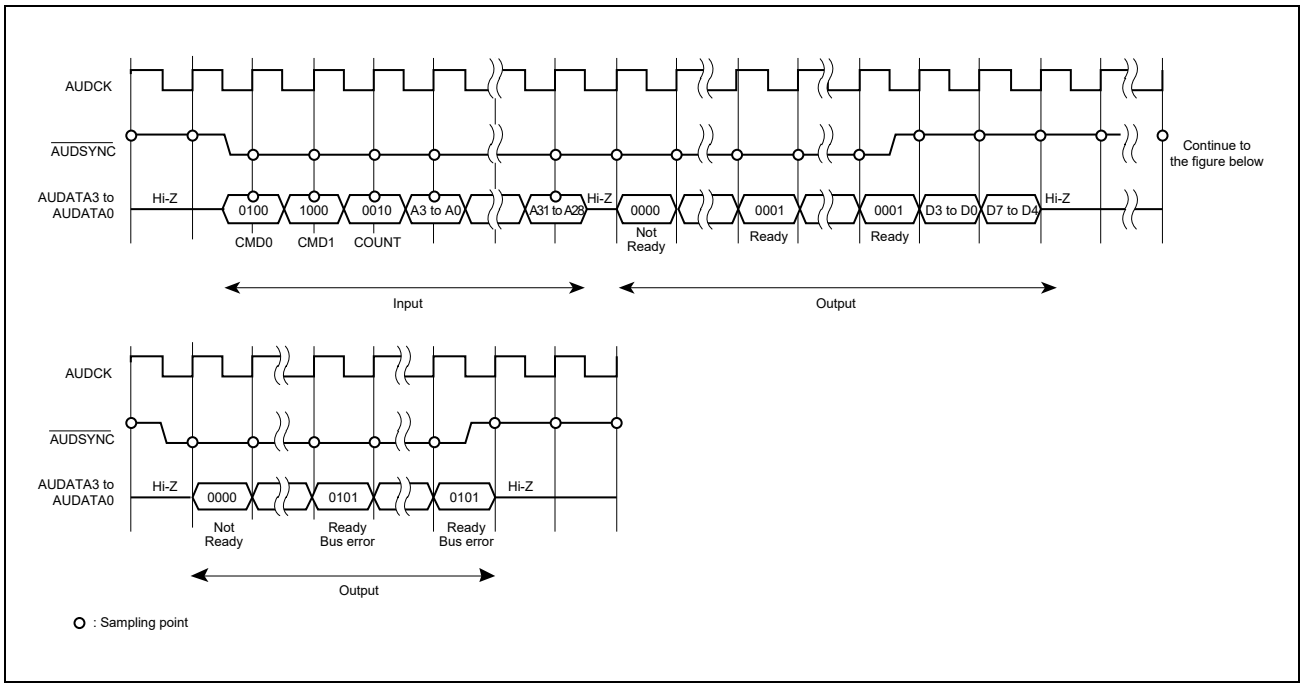


Figure 34.8 Example of Error Occurrence in Continuous Transfer (address 32 bits; byte read × three units of data)

(3) Command error conditions

Table 34.7 Command (CMD0) Error Conditions

Bit 3 (TYP1)	Bit 2 (TYP0)	Bit 1 (ASZ1)	Bit 0 (ASZ0)	Description
0	0	0	0	Single transfer Address bit width: 32 bits
0	0	0	1	Single transfer Address bit width: 8 bits
0	0	1	0	Single transfer Address bit width: 16 bits
0	0	1	1	Single transfer Address bit width: 24 bits
0	1	0	0	Continuous transfer Address bit width: 32 bits
0	1	0	1	Continuous transfer Address bit width: 8 bits
0	1	1	0	Continuous transfer Address bit width: 16 bits
0	1	1	1	Continuous transfer Address bit width: 24 bits
1	×	×	×	Command error

Table 34.8 Command (CMD1) Error Conditions

Bit 3 (IT)	Bit 2 (RW)	Bit 1 (SIZ1)	Bit 0 (SIZ0)	Description
0	x	x	x	Command error
1	0	0	0	Read: Byte
1	0	0	1	Read: Half word
1	0	1	0	Read: Word
1	0	1	1	Read: Double word
1	1	0	0	Write: Byte
1	1	0	1	Write: Half word
1	1	1	0	Write: Word
1	1	1	1	Write: Double word

(4) Bus error conditions

- Half-word access is made to the addresses of 4n + 1 and 4n + 3.
- Word access is made to the addresses of 4n + 1, 4n + 2, and 4n + 3.
- Double-word access is made to the addresses of 8n + 1, 8n + 2, 8n + 3, 8n + 4, 8n + 5, 8n + 6, and 8n + 7.
- An error response is received from the system bus.

(5) AUDATA pin input format

Table 34.9 Input Format Bit Position

Order of Input	Format Name	Bit Position				√: Required; —: Not Required			
		AUDATA3	AUDATA2	AUDATA1	AUDATA0				
Earlier	CMD0	TYP1	TYP0	ASZ1	ASZ0	√			
	CMD1	IT	RW	SIZ1	SIZ0	√			
	COUNT	C3	C2	C1	C0	— (single transfer); √ (continuous transfer)			
	Address					8 bits	16 bits	24 bits	32 bits
		A3	A2	A1	A0	√	√	√	√
		A7	A6	A5	A4	√	√	√	√
		A11	A10	A9	A8	—	√	√	√
		A15	A14	A13	A12	—	√	√	√
		A19	A18	A17	A16	—	—	√	√
		A23	A22	A21	A20	—	—	√	√
		A27	A26	A25	A24	—	—	—	√
		A31	A30	A29	A28	—	—	—	√
	Data (for writing only)					Byte write	Half-word write	Word write	Double-word write
		D3	D2	D1	D0	√	√	√	√
		D7	D6	D5	D4	√	√	√	√
		D11	D10	D9	D8	—	√	√	√
		D15	D14	D13	D12	—	√	√	√
		D19	D18	D17	D16	—	—	√	√
		D23	D22	D21	D20	—	—	√	√
		D27	D26	D25	D24	—	—	√	√
		D31	D30	D29	D28	—	—	√	√
		D35	D34	D33	D32	—	—	—	√
		D39	D38	D37	D36	—	—	—	√
		D43	D42	D41	D40	—	—	—	√
		D47	D46	D45	D44	—	—	—	√
		D51	D50	D49	D48	—	—	—	√
		D55	D54	D53	D52	—	—	—	√
		D59	D58	D57	D56	—	—	—	√
Later		D63	D62	D61	D60	—	—	—	√

Table 34.10 CMD0 Format

Bit Name	Function	Description
TYP[1:0]	Type of transfer	00: Single transfer 01: Continuous transfer
ASZ[1:0]	Specify an address bit width	These bits specify the bit width of an address input from the AUDATA pin. When 8-, 16-, or 24-bits are specified, the same values as the previous access address are used for the upper bits that are not input from the AUDATA pin. Input a 32-bit address for the first access after reset release or occurrence of a command or bus error. 00: 32 bits 01: 8 bits 10: 16 bits 11: 24 bits

Table 34.11 CMD1 Format

Bit Name	Function	Description
IT	Specifies access space	Set this bit to 1.
RW	Specifies read or write	0: Read 1: Write
SIZ[1:0]	Specify data size	These bits specify the size of data to be accessed. 00: Byte (8 bits) 01: Half word (16 bits) 10: Word (32 bits) 11: Double word (64 bits)

Table 34.12 COUNT Format

Bit Name	Function	Description
C3 to C0	Specify the number of data to be transferred	These bits specify the number of data to be transferred in continuous transfer. 0000: 1 data 0001: 2 data 0010: 3 data 0011: 4 data 0100: 5 data 0101: 6 data 0110: 7 data 0111: 8 data 1000: 9 data 1001: 10 data 1010: 11 data 1011: 12 data 1100: 13 data 1101: 14 data 1110: 15 data 1111: 16 data

Table 34.13 Address Format

Bit Name	Function	Description
A31 to A0	Specify address	These bits specify an address to be accessed. The required number of bits depends on the setting of the ASZ[1:0] bits in CMD0 (for details, see **Table 34.9**).

Table 34.14 Write Data Format

Bit Name	Function	Description
D63 to D0	Specify write data	These bits specify write data. The required number of bits depends on the setting of the SIZ[1:0] bits in CMD1 (for details, see **Table 34.9**).

(6) AUDATA pin output format

Table 34.15 Ready Flag Format

Bit Position	Bit Name	Function	Description
AUDATA3	0	—	—
AUDATA2	BFLG	This bit indicates a bus error.	0: Normal 1: A bus error occurred.
AUDATA1	CFLG	This bit indicates a command error.	0: Normal 1: A command error occurred.
AUDATA0	RFLG	This bit indicates completion of AUDR operation.	0: Not ready 1: Ready

Table 34.16 Read Data Bit Position

Output Order	Bit Position				√: Required; —: Not Required			
	AUDATA3	AUDATA2	AUDATA1	AUDATA0	Byte read	Half-word read	Word read	Double-word read
Earlier	D3	D2	D1	D0	√	√	√	√
	D7	D6	D5	D4	√	√	√	√
	D11	D10	D9	D8	—	√	√	√
	D15	D14	D13	D12	—	√	√	√
	D19	D18	D17	D16	—	—	√	√
	D23	D22	D21	D20	—	—	√	√
	D27	D26	D25	D24	—	—	√	√
	D31	D30	D29	D28	—	—	√	√
	D35	D34	D33	D32	—	—	—	√
	D39	D38	D37	D36	—	—	—	√
	D43	D42	D41	D40	—	—	—	√
	D47	D46	D45	D44	—	—	—	√
	D51	D50	D49	D48	—	—	—	√
	D55	D54	D53	D52	—	—	—	√
	D59	D58	D57	D56	—	—	—	√
Later	D63	D62	D61	D60	—	—	—	√

Table 34.17 Read Data Format

Bit Name	Function	Description
D63 to D0	Output read data	The number of output bits depends on the setting of the SIZ[1:0] bits in CMD1 (for details, see **Table 34.16**).

34.4.4.3 Usage Notes on the AUDR Function

- Do not negate the $\overline{\text{AUDSYNC}}$ pin until one cycle of AUDCK has elapsed after a command is input to the AUDATA pin and the Ready flag has been returned.
- When uninitialized memory is accessed through the AUDR, a bus error may occur due to ECC error detection.

34.4.4.4 Enabling and Disabling RAM Monitoring

Enabling or disabling of the RAM monitoring function can be specified by using the option byte (AUDREN).

AUDREN: Enable Bit of the AUDR

For setting of option byte, see **Section 35, Flash Memory**.

NOTE

The AUDR is disabled in serial programming mode regardless of this setting.

34.5 Cautions on Using On-Chip Debugger

(1) Treatment of devices used for debugging

Do not install a device that was used for debugging on a mass-produced product, because the flash memory was rewritten during system debugging and thus the number of write/erase count of flash memory cannot be guaranteed.

(2) Hot plug-out is not supported

This product does not support hot plug-out for power off (including removal of the connector) of the debug tool in debug mode. Do not turn off the power of the NEXUS tool (including removal of the connector) in debug mode.

(3) Handling of pins when ending on-chip debugging

When ending on-chip debugging, set the $\overline{\text{DCUTRST}}$ pin and external reset pin to the low level.

(4) When using the debugger

When a debugger is used, the program written to the microcomputer before preparation for communications between the OCD emulator and microcomputer is complete is executed from the reset vector. For this reason, take care to ensure that this does not lead to any operations which you do not expect or intend.

This communications preparation period depends on the host PC environment of the OCD emulator and the operating frequency of the microcomputer.

Section 35 Flash Memory

This section describes the characteristics of the flash memory mounted on the RH850/C1M-A, memory map, flash memory programming, and so on.

35.1 Features

- Code flash memory capacity:
  - User area C1M-A2: 4 Mbytes (two-bank configuration)
    C1M-A1: 2 Mbytes (one-bank configuration)
  - User boot area 32 Kbytes
- Data flash memory capacity:
  - Data area 64 Kbytes
- Method of programming
  - Programming by a dedicated flash memory programmer via a serial interface communication (serial programming)
  - Programming of the flash memory by a user program (self-programming)
- Support for security functions to protect against illicit tampering with or reading out of data in flash memory
- Support for protection functions to protect against erroneous overwriting of the flash memory
- Support for the error detection/correction function in the flash memory
- Support for the BGO (Back Ground Operation) function
  - The code flash memory can be read during programming the data flash memory
- The initial setting of this product can be set to an expansion area of the flash memory (option bytes)

35.2 Structure of Memory

Figure 35.1 and **Figure 35.2** illustrate the mapping of the code flash memory in the C1M-A2 and C1M-A1, respectively. The user area of the code flash memory of this product is divided into 8- and 32-Kbyte blocks, which serve as the units of erasure. The user area is available as an area for storing the user program.

As a protection area that cannot be rewritten by the self-programming, a single block of 32-Kbyte user boot area is incorporated. The user boot area is available as an area for storing a boot program and the like where you want to prohibit rewriting at the time of operation of the user program. The boot program includes the one for programming the code flash memory using any interface.

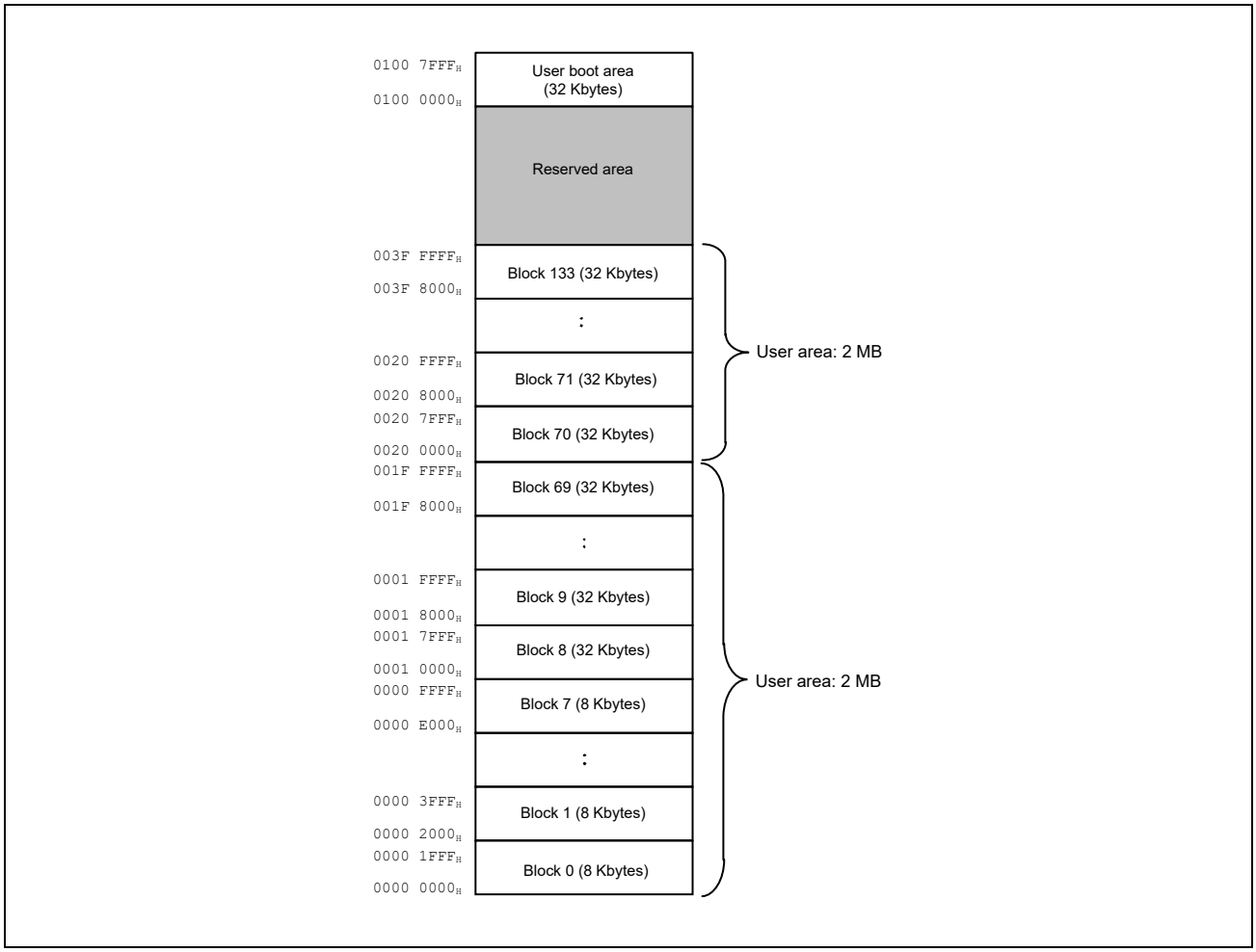


Figure 35.1 Mapping of the Code Flash Memory (8 KB × 8 + 32 KB × 126 configuration) in the C1M-A2

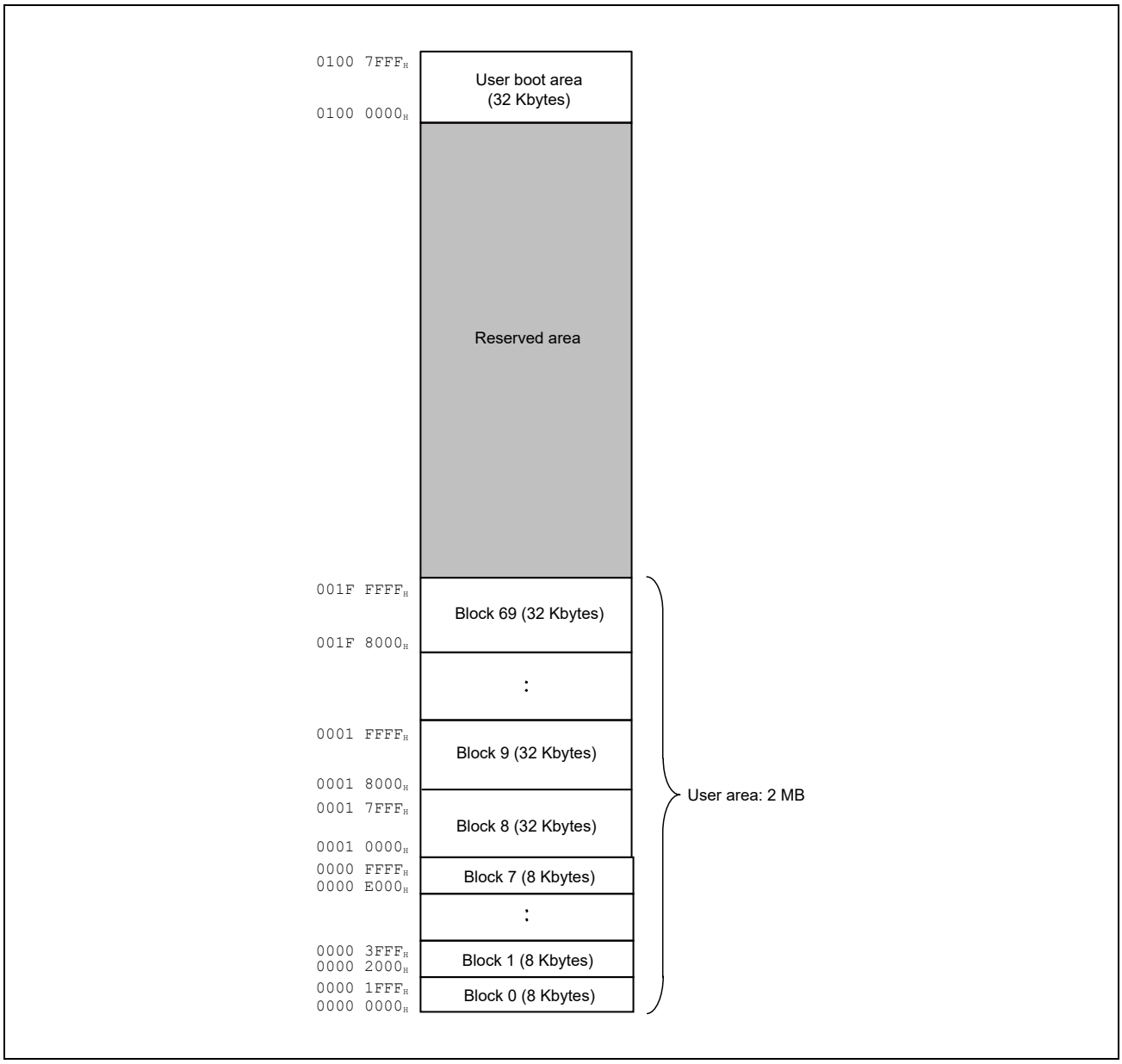


Figure 35.2 Mapping of the Code Flash Memory (8 KB × 8 + 32 KB × 62 configuration) in the C1M-A1

The data area of the data flash memory in this product is divided into 64-byte blocks, with each being a unit for erasure. **Figure 35.3** shows the mapping of the data flash memory.

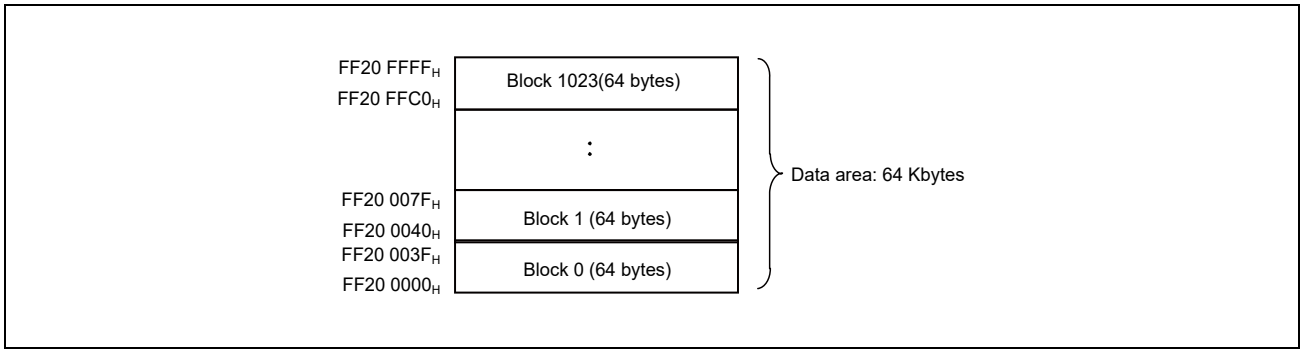


Figure 35.3 Mapping of the Data Flash Memory (64 bytes × 1024 configuration)

35.3 Operating Modes Associated with Flash Memory

Figure 35.4 is a diagram of the mode transitions associated with the flash memory. For the procedures for setting the modes, refer to **Section 5, Operating Mode**.

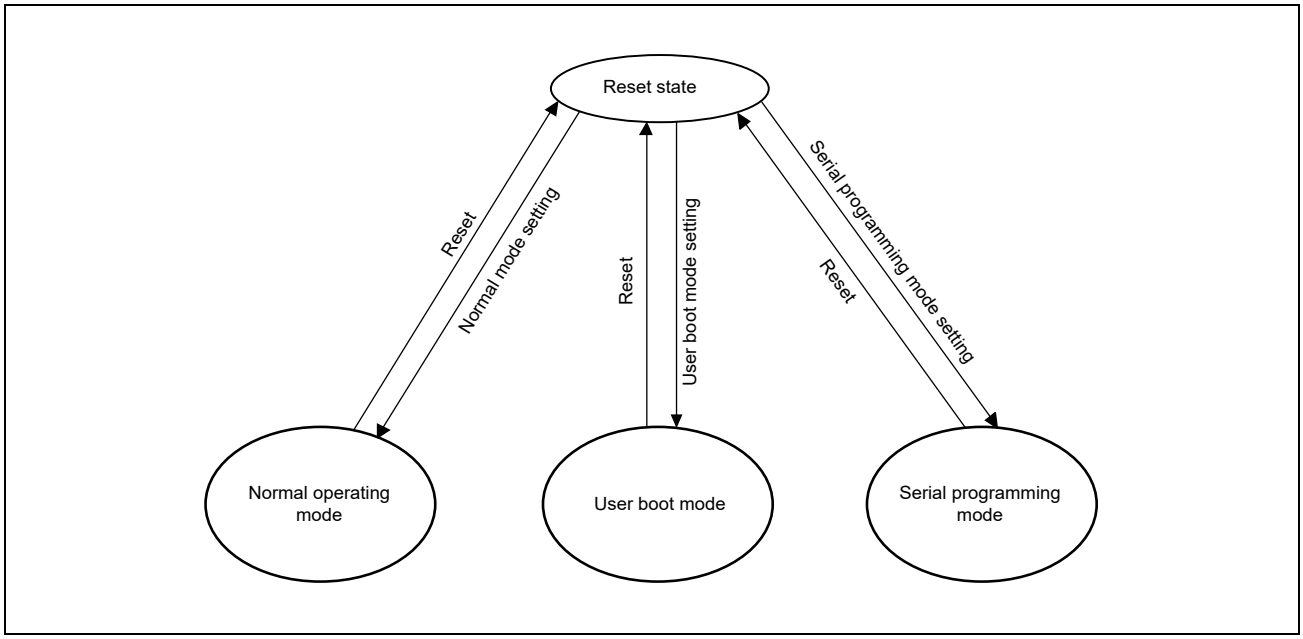


Figure 35.4 Mode Transition Associated with Flash Memory

The flash memory area which is programmable and erasable and the boot program after reset release are different in each mode. **Table 35.1** shows the differences.

Table 35.1 Differences in Each Mode

Item	Normal Operating Mode*1	User Boot Mode	Serial Programming Mode
Programmable and erasable area	• User area • Data area	• User area • Data area	• User area • User boot area • Data area
Boot program at the time of reset	Program in user area	Program in user boot area	Embedded program for serial programming

Note 1. Normal operating mode refers to a user boot mode whose activation area is a user area.

35.4 Functional Overview

The flash memory mounted on this product can be updated via a serial interface communication by a dedicated flash memory programmer (serial programming), whether before or after being mounted on the target system.

Furthermore, security functions to prohibit updating of the user program written in the mounted flash memory are incorporated, and this can prevent tampering by third parties.

Programming function by the user program (self-programming) is suited for applications where the target system program may require updating after manufacturing or shipping. Protection features for the safe rewriting of the flash memory area are also incorporated. In addition, interrupt processing during self-programming is supported, so programming can proceed at the same time as interrupt processing associated with external communication control, etc., and this allows programming under various conditions.

Table 35.2 gives an overview of the methods of programming and the corresponding operating modes.

Table 35.2 Methods of Programming

Methods of Programming	Functional Overview	Operating Mode
Serial programming	A dedicated flash-memory programmer is capable of on-board programming the flash memory after the device is mounted on the target system.	Serial programming mode
	A dedicated flash-memory programmer and dedicated programming adapter board allow off-board programming of the flash memory, i.e. programming of the device before it is mounted on the target system.	
Self programming	The user program that is written to code flash memory in advance of serial programming executing is also capable of programming the flash memory. The background operation capability makes it possible to fetch instructions or otherwise read data from code flash memory while the data flash memory is being programmed. For this reason, it is possible to program the data flash memory by executing a program written to the code flash memory. Instructions in the code flash memory cannot be fetched and data cannot be accessed while the code flash memory is being programmed by self-programming. In such cases, a program for updating must be transferred to the local RAM or global RAM in advance and executed.	Normal operating mode User boot mode

When performing self-programming, see the *RH850/C1M-A Flash Memory User's Manual: Hardware Interface* of this product.

Table 35.3 lists the functions of the mounted flash memory. Dedicated flash-memory programmer commands realize serial programming, while reading of the on-chip flash memory by interface operation of flash memory or the user program realizes self-programming.

Table 35.3 Basic Functions at a Glance

Function	Description in Overview	Level of Support (○: Supported, △: Conditionally Supported, ×: Not Supported)	
		Serial programming	Self-programming
Blank checking	This is used to check a specified block to ensure that writing to it has not already proceeded. Results of reading from code flash memory and data flash memory to which nothing has been written after erasure are not guaranteed, so use blank checking to confirm that writing to memory has not proceeded after erasure.	○	○
Block erasure	This is for erasing the contents of a specified block of memory.	○	○
Programming	This is for writing to a specified address.	○	○
Verification and checksum	Data that are read out from flash memory are compared with data transferred from the flash memory programmer.	○	○
Reading	Data that have been written to the flash memory are read out.	○	○
Setting for OTP (one-time programming)	A specified block of code flash memory is set for OTP (OTP can only be set, that is, it is not possible to release a block's OTP setting).	○	○
Setting an ID	An ID setting is made for use in controlling the connection of a dedicated flash memory programmer for serial programming and enabling of programming of the code flash memory by self-programming.	○	○
Security settings	Security settings are for use in serial programming.	○	△ (only when setting is prohibited after being permitted)
Protection settings	Lock bits for all blocks of code flash memory are provided.	○	○
Setting of option bytes	Option bytes are set to change them from the initial values for this product.	○	○
Clearing the configuration	ID setting, security settings, protection settings, and option byte settings are initialized.	○	×

For details on serial programming, see the *PG-FP5 Flash Memory Programmer User's Manual*, and the *Renesas Flash Programmer Flash Programming Software User's Manual.*

When performing self-programming, see the *RH850/C1M-A Flash Memory User's Manual: Hardware Interface* of this product.

The mounted flash memory supports various security functions.

The OTP setting and authentication of the ID code are security functions for use with serial programming and self-programming.

In serial programming, authentication of the ID code, prohibiting connection of a dedicated flash-memory programmer, and prohibition of commands (for block erasure, programming, and reading) are available for use as security functions.

The security functions supported by the mounted flash memory are listed in **Table 35.4**, and the operations when security is enabled are listed in **Table 35.5**.

Table 35.4 Summary of Security Functions

Function	Description
OTP	OTP can be individually set for each block of the user area and the user boot area of code flash memory. When the OTP setting is made for an area, programming by serial programming and by self-programming is prohibited. Once set, the OTP setting cannot be released. Furthermore, since execution of the configuration clearing command is prohibited for any area for which OTP has been set, changing a security setting from “prohibited” to “permitted” is not possible.
ID authentication	The result of ID authentication can be used to control the connection of a dedicated flash memory programmer for serial programming. The result of ID authentication can also be used to control enabling of programming of the code flash memory by self-programming.
Prohibition of connection of a dedicated flash memory programmer	The connection of a dedicated flash memory programmer for serial programming is prohibited. Since execution of the configuration clearing command is also prohibited when the connection of a dedicated flash memory programmer is prohibited, changing a security setting from “prohibited” to “permitted” is not possible.
Prohibition of block erasure commands	Block erasure commands at the time of serial programming are prohibited. Since execution of the configuration clearing command is also prohibited when block erasure commands are prohibited, changing a security setting from “prohibited” to “permitted” is not possible.
Prohibition of programming commands	Block erasure commands and programming commands at the time of serial programming are prohibited. Block erasure commands can be run only by the method of erasing all user areas from Block 0 in turn → erasing the user boot areas → erasing all data areas from Block 0 in turn. Only through execution of the configuration clearing command, the prohibition can be lifted.
Prohibition of read commands	Read commands at the time of serial programming are prohibited. Only through execution of the configuration clearing command, the prohibition can be lifted.

Table 35.5 Available Operations and Security Settings

Function	All Security Settings and Erasure, Programming, and Read Operations (○: Executable, ×: Not Executable, –: Not Supported)		Point for Caution Regarding the Security Setting	
	Serial programming	Self-programming	Serial programming	Self-programming
OTP	• Areas for which OTP is set Block erasure commands: × Programming commands: × Read commands: ○ • Areas for which OTP is not set Block erasure commands: ○ Programming commands: ○ Read commands: ○	• Areas for which OTP is set Block erasure: × Programming: × Reading: ○ • Areas for which OTP is not set Block erasure: ○ Programming: ○ Reading: ○	The OTP setting cannot be released. Execution of the configuration clearing command is not possible.	The OTP setting cannot be released.
ID authentication	• When the ID codes do not match Block erasure commands: × Programming commands: × Read commands: × • When the ID codes match Block erasure commands: ○ Programming commands: ○ Read commands: ○	• When the ID codes do not match - Code flash memory Block erasure: × Programming: × Reading: ○ - Data flash memory Block erasure: ○ Programming: ○ Reading: ○ • When the ID codes match Block erasure: ○ Programming: ○ Reading: ○	The configuration clearing command can initialize the setting for prohibition. The setting for prohibition of block erasure commands is not available. The setting for prohibition of programming commands is not available. The setting for prohibition of read commands is not available.	ID authentication is always in effect.
Prohibition of the connection of a dedicated flash memory programmer	Block erasure commands: × Programming commands: × Read commands: ×	Block erasure: ○ Programming: ○ Reading: ○	Since execution of the configuration clearing command is prohibited, initialization of the setting for prohibition is not possible.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of block erasure commands	Block erasure commands: × Programming commands: ○ Read commands: ○	Block erasure: ○ Programming: ○ Reading: ○	Since execution of the configuration clearing command is prohibited, initialization of the setting for prohibition is not possible. The setting for ID authentication to be effective for serial programming is not available.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of programming commands	Block erasure commands: ×*1 Programming commands: × Read commands: ○	Block erasure: ○ Programming: ○ Reading: ○	Executing the configuration clearing command only can initialize the setting for prohibition. The setting for ID authentication to be effective for serial programming is not available.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of read commands	Block erasure commands: ○ Programming commands: ○ Read commands: ×	Block erasure: ○ Programming: ○ Reading: ○		

Note 1. Block erasure commands can be run only by the method of erasing all user areas from Block 0 in turn → erasing the user boot areas → erasing all data areas from Block 0 in turn.

The mounted flash memory supports various protection functions. The protection functions supported by the mounted flash memory are listed in **Table 35.6.**

Table 35.6 Summary of Protection Functions

Function	Description
Block protection	Lock bit settings can be individually made to enable or disable programming and erasure of each block of the user area of code flash memory. Programming and erasure by self-programming of an area for which the lock bit is set and the lock bit function is enabled are prohibited. Programming or erasure can proceed again when the lock bit function is disabled after having been enabled. When a block of code flash memory is erased, the lock bit for that block is also erased.
User boot protection	Programming and erasure of the user boot area by self-programming are prohibited. Programming and erasure of the user boot area by serial programming are possible.

35.5 Serial Programming

A dedicated flash memory programmer can be used for programming of the flash memory in serial programming mode.

Serial Programming

The microcontroller is mounted on the system board at the time of serial programming. Providing a connector to the board enables writing to the microcontroller by the flash memory programmer to proceed.

35.5.1 Environments for Programming

The recommended environments for writing the data into the flash memory of the microcontroller are described below.

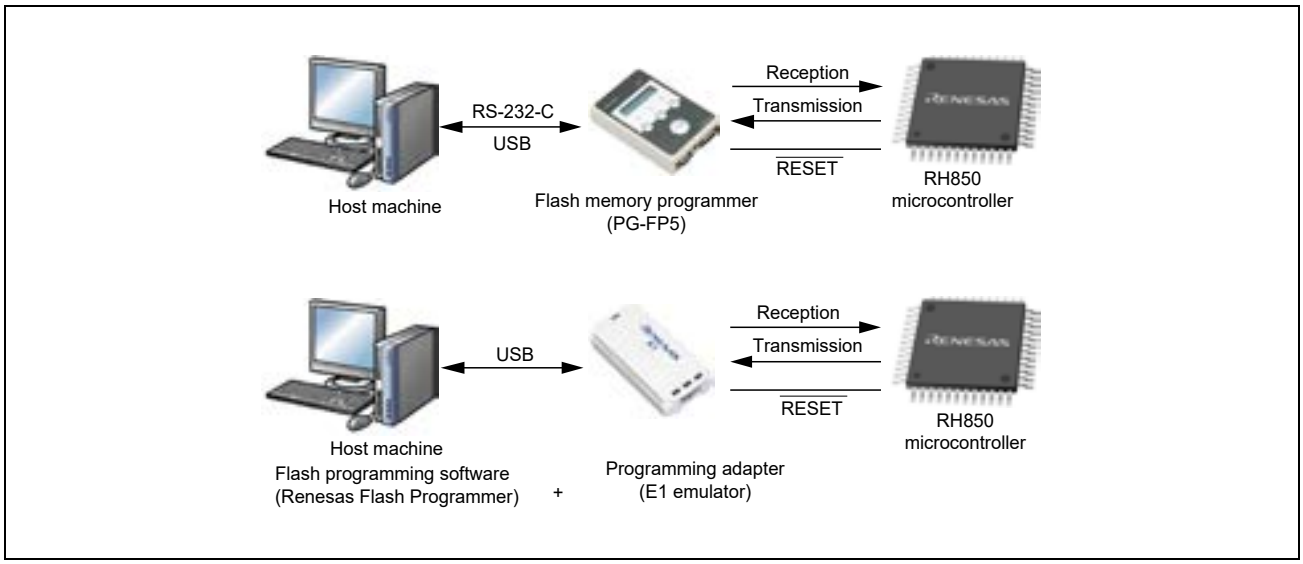


Figure 35.5 Environments for Writing Programs into the Flash Memory

By using the PG-FP5 flash memory programmer or the combination of the Renesas Flash Programmer (software for writing to flash memory) running on the host machine and the E1 emulator as an programming adaptor, the user is easily able to erase, program, and verify the contents of the on-chip memory of flash-memory-equipped microcontrollers from Renesas Electronics.

The PG-FP5 flash memory programmer handles programming from a host machine or programming in stand-alone mode while the Renesas Flash Programmer only handles programming from a host machine.

NOTE

Refer to the *PG-FP5 Flash Memory Programmer User's Manual* for the details of PG-FP5, and the *Renesas Flash Programmer Flash Programming Software User's Manual* for details of Renesas flash programmer for flash programming software.

35.5.2 Selection of the Communication Method

In this product, either 2-wire UART method or clock synchronous method can be selected as a serial communication method by the FLMODE pin. For details on how to set the FLMODE pin, see **Section 5, Operating Modes.** For details on how to set the programming environment in accordance with each communication method, see the *PG-FP5 Flash Memory Programmer User's Manual* and the *Renesas Flash Programmer Flash Programming Software User's Manual.*

35.6 Self-Programming

35.6.1 Overview

This product supports programming of the flash memory by the user program itself. The code flash and data flash memory can be programmed by using the commands of flash memory application command interface (FACI) for flash memory programming in user's applications. Therefore, update of the user program and programming of constant data fields can be possible.

When the data flash memory is programmed, the background operation facility makes it possible to execute a programming program from the code flash memory to program the data flash memory. Furthermore, the programming program can be copied to local RAM or global RAM in advance of the programming operation, and executed from the given destination to perform the programming.

For the comprehensive information on flash self-programming, see the *Code Flash Library,* the *Data Flash Library*, and the *RH850/C1M-A Flash Memory User's Manual: Hardware Interface* of this product.

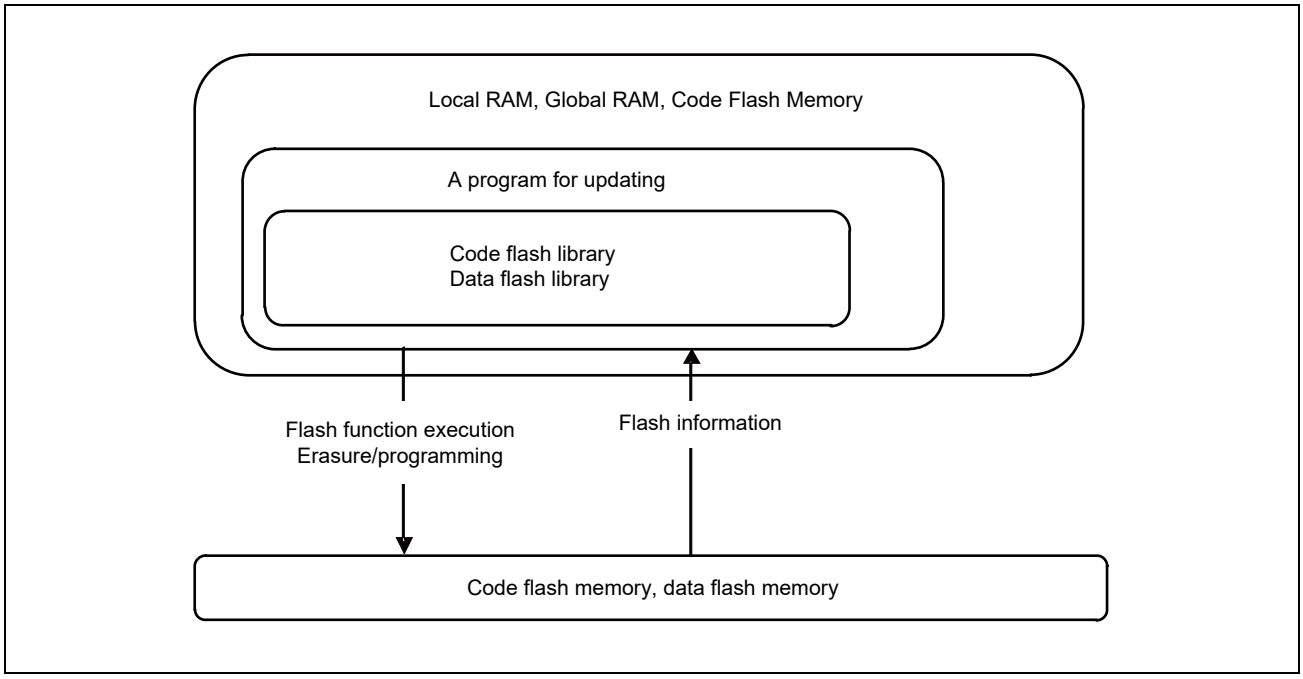


Figure 35.6 Concept of Self-Programming

35.6.2 Background Operation

Background operations can be used when the combination of the flash memory for writing and the flash memory for reading is any of those listed below.

Table 35.7 Conditions under which Background Operation is Usable

Range for Writing	Range for Reading
Data Flash Memory	Code Flash Memory

35.7 Reading Flash Memory

35.7.1 Reading Code Flash Memory

Special settings are not required to read code flash memory in normal mode and user boot mode. Data can simply be read out through access to addresses in the code flash memory. Code Flash

Reading from an area of code flash memory that has been erased but not yet been programmed again (i.e. that is in the non-programmed state) can lead to the detection of an ECC error and generation of the corresponding exception. In addition, since the data value is not guaranteed in case of an ECC error, use blank checking when you need to confirm that an area is in the non-programmed state. For details on the ECC function, see **Section 29, Functional Safety**.

35.7.2 Reading Data Flash Memory

Configure the number of read cycles in the FRDCYCLD register prior to reading data from data flash memory in normal mode and user boot mode. Once this register is properly configured, data can be read by simply accessing addresses in the data flash memory.

Values read from data flash memory that has been erased but not yet been programming again are undefined. Use blank checking when you need to confirm that an area is in the non-programmed state.

35.8 Register Descriptions

35.8.1 Registers Related to Data Flash Memory

Table 35.8 lists registers related to the data flash memory.

Table 35.8 List of Registers related to the Data Flash Memory

Module Name	Register Name	Abbreviation	R/W	Value after reset	Address	Access size
FLASH	Data flash memory read cycle setting register	FRDCYCLD	R/W	$0F_H$	FFC5 9810_H	8

35.8.1.1 FRDCYCLD - Data Flash Memory Read Cycle Setting Register

This register is used to set the data flash memory read cycle.

Access: This register can be read or written in 8-bit units.

Address: FFC5 9810_H

Value after reset: $0F_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	FRDCYCLD[3:0]			
Value after reset	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 35.9 FRDCYCLD Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved
3 to 0	FRDCYCLD [3:0]	Number of the data flash memory read cycle The data flash memory is read by the set value + 1 cycle. 0_H-2_H: Setting prohibited 3_H: Read cycle 4 4_H: Read cycle 5 5_H: Read cycle 6 6_H: Read cycle 7 7_H: Read cycle 8 8_H: Read cycle 9 9_H-F_H: Read cycle 10

Note: 1 read cycle is CLK_LSB.

35.8.2 Registers Related to Programming/Erasure Protection of Flash Memory

Table 35.10 lists registers related to the programming/erasure protection of the flash memory.

Table 35.10 List of Registers Related to the Programming/Erasure Protection of the Flash Memory

Module Name	Register Name	Abbreviation	R/W	Value after reset	Address	Access size
FLASH	FHVE15 control register	FHVE15	R/W	0000 0000_H	FFF8 $A430_H$	32
FLASH	FHVE3 control register	FHVE3	R/W	0000 0000_H	FFF8 2410_H	32

35.8.2.1 FHVE15 — FHVE15 Control Register

FHVE15 is a readable/writable register for software protection of flash memory against programming, erasure, and blank checking. Set both the FHVE15 and FHVE3 registers in the programmable, erasable, and blank-checkable state (0000 0001_H) to program, erase, or blank-check flash memory.

Access: This register can be read or written in 32-bit units.

Address: FFF8 $A430_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FHVE 15CNT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 35.11 FHVE15 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	FHVE15CNT	0: Programming, erasure, and blank checking are disabled. 1: Programming, erasure, and blank checking are enabled.

35.8.2.2 FHVE3 — FHVE3 Control Register

FHVE3 is a readable/writable register for software protection of flash memory against programming, erasure, and blank checking. Set both the FHVE15 and FHVE3 registers in the programmable, erasable, and blank-checkable state (0000 0001_H) to program, erase, or blank-check flash memory.

Access: This register can be read or written in 32-bit units.

Address: FFF8 2410_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FHVE3CNT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 35.12 FHVE3 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	FHVE3CNT	0: Programming, erasure, and blank checking are disabled. 1: Programming, erasure, and blank checking are enabled.

35.8.3 Registers Related to Product Information

The flash memory has the expansion area that stores the product information such as a product name and the size of included memory. The value stored in this expansion area is transferred to the registers related to product information at the time of reset.

Table 35.13 lists registers related to product information.

Table 35.13 List of Registers Related to Product Information

Module Name	Register Name	Abbreviation	R/W	Value after reset	Address	Access size
FLASH	Product name storage register (1)	PRDNAME1	R	See **Table 35.14**	FFCD 00D0$_H$	32
FLASH	Product name storage register (2)	PRDNAME2	R	See **Table 35.14**	FFCD 00D4$_H$	32
FLASH	Product name storage register (3)	PRDNAME3	R	See **Table 35.14**	FFCD 00D8$_H$	32
FLASH	Product name storage register (4)	PRDNAME4	R	See **Table 35.14**	FFCD 00DC$_H$	32

Table 35.14 Relation between Product Name and PRDNAME Initial Value (Value after reset)

Product Group Name	Product Model Name	PRDNAME4	PRDNAME3	PRDNAME2	PRDNAME1
C1M-A2	R7F701275	2020 2020$_H$	2020 2035$_H$	3732 3130$_H$	3746 3752$_H$
C1M-A1	R7F701278	2020 2020$_H$	2020 2038$_H$	3732 3130$_H$	3746 3752$_H$

35.8.3.1 PRDNAMEn (n = 1 to 4) — Product Name Storage Register

This register stores the product name. The product model name is stored in 16-byte ASCII code, and PRDNAME1, PRDNAME2, PRDNAME3, and PRDNAME4 correspond to the fourth to first bytes, eighth to fifth bytes, twelfth to ninth bytes, and sixteenth to thirteenth bytes of the product model name respectively.

Access: This register can be read in 32-bit units.

Address: PRDNAME1: FFCD 00D0$_H$
PRDNAME2: FFCD 00D4$_H$
PRDNAME3: FFCD 00D8$_H$
PRDNAME4: FFCD 00DC$_H$

Value after reset: C1M-A product name Initial value (PRDNAMEn, n = 1 to 4)
See **Table 35.14, Relation between Product Name and PRDNAME Initial Value (Value after reset)**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PRDNAMEn															
Value after reset*1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRDNAMEn															
Value after reset*1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. The value is the one shown in **Table 35.14, Relation between Product Name and PRDNAME Initial Value (Value after reset)**

Table 35.15 List of Registers Related to Product Information

Bit Position	Bit Name	Function
31 to 0	PRDNAMEn[31:0]	Product Name Product model names are shown in 16-byte ASCII code. PRDNAME1[31:0]: The fourth to first bytes of the product model name PRDNAME2[31:0]: The eighth to fifth bytes of the product model name PRDNAME3[31:0]: The twelfth to ninth bytes of the product model name PRDNAME4[31:0]: The sixteenth to thirteenth bytes of the product model name

35.9 Option Bytes

The flash memory has the extended area (option bytes) to store data specified by the user for various purposes. **Table 35.16** shows the setting areas of the option bytes. In a reserved area, 1 is always read out. Also, always set 1 as a set value. Changes in settings such as initial setting of peripheral functions using option bytes become effective after release from the reset state. For details on option bytes setting and how to read out, see the *PG-FP5 Flash Memory Programmer User's Manual*, and the *Renesas Flash Programmer Flash Programming Software User's Manual*, or the *RH850/C1M-A Flash Memory User's Manuals: Hardware Interface.*

Table 35.16 Setting Areas of Option Bytes

Option Byte Area (8 bits each x 32 = 256 bits in total)	Option Byte Register	Setting-enabled Area	Initial State at the time of Shipping*1
Option Bytes 4 to 1	OPBT0	Enabled	7FFF FFFE$_H$
Option Bytes 8 to 5			FFFF FFFF$_H$
	—	Reserved	
Option Bytes 12 to 9	OPBT2	Enabled	FFFF FFFF$_H$
Option Bytes 16 to 13	OPBT3	Enabled	7FFF FFFF$_H$
Option Bytes 20 to 17	OPBT4	Enabled	7FFF FFFF$_H$
Option Bytes 24 to 21	—	Reserved	FFFF FFFF$_H$
Option Bytes 28 to 25	OPBT6	Enabled	FFFF FFFA$_H$
Option Bytes 32 to 29	—	Reserved	FFFF FFFF$_H$

Note 1. This is a value of the initial state of the shipping product, which can be changed by the option byte setting.

35.9.1 OPBT0 — Option Byte 0 Register

Access: This register can be read /written in 32-bit units only when using on-chip debug function.

Address: FFCD 0030$_H$

Value after reset: User defined (7FFF FFFE$_H$ at the time of shipping)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPWD RUN	OPWDI NT	OPWD WS1	OPWD WS0	OPWD OVF2	OPWD OVF1	OPWD OVF0	—	—	—	—	—	—	—	—	—
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AUDRE N	—	—	STMSE L1	STMSE L0
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. For the option byte areas and the initial state at the time of shipping of this register, see **Table 35.16**. Reading and setting (R/W) of the option byte areas are possible only when using FACI commands, PG-FP5 Flash Memory Programmer, and Renesas Flash Programmer Flash Programming Software.

Table 35.17 OPBT0 Register Contents

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>31</td><td>OPWDRUN</td><td>Selects the start mode of WDTA0.
0: WDTA0 software trigger start mode
1: WDTA0 default start mode</td></tr>
<tr><td>30</td><td>OPWDINT</td><td>Enables or disables 75% interrupt request WDTA0TIT of WDTA0.
0: Disables WDTA0TIT
1: Enables WDTA0TIT</td></tr>
<tr><td>29, 28</td><td>OPWDWS1,
OPWDWS0</td><td>Selects the window-open period of WDTA0.
<table>
<tr><th>OPWDWS1</th><th>OPWDWS0</th><th>Activation Area</th></tr>
<tr><td>0</td><td>0</td><td>25%</td></tr>
<tr><td>0</td><td>1</td><td>50%</td></tr>
<tr><td>1</td><td>0</td><td>75%</td></tr>
<tr><td>1</td><td>1</td><td>100%</td></tr>
</table></td></tr>
<tr><td>27 to 25</td><td>OPWDOVF2
to
OPWDOVF0</td><td>Selects the overflow interval time of WDTA0.
<table>
<tr><th>OPWDOVF2</th><th>OPWDOVF1</th><th>OPWDOVF0</th><th>Overflow Interval Time</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>2^9 / WDTATCKI</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>2^{10} / WDTATCKI</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>2^{11} / WDTATCKI</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>2^{12} / WDTATCKI</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>2^{13} / WDTATCKI</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>2^{14} / WDTATCKI</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>2^{15} / WDTATCKI</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>2^{16} / WDTATCKI</td></tr>
</table></td></tr>
<tr><td>24 to 5</td><td>—</td><td>Reserved
When read, the value after reset is returned.
When writing to these bits, write the read value.</td></tr>
<tr><td>4</td><td>AUDREN</td><td>AUDRAM Monitor Enable
0: Disables AUDRAM monitor
1: Enables AUDRAM monitor</td></tr>
<tr><td>3, 2</td><td>—</td><td>Reserved
When read, the value after reset is returned.
When writing to these bits, write the read value.</td></tr>
<tr><td>1, 0</td><td>STMSEL1,
STMSEL0</td><td>Selects the operating mode activation area.
The operating mode activation area can be selected by the combination of the values of STMSEL1 and STMSEL0 if MD0, MD1, and FLMODE pins are all 0. For details, see Section 5, Operating Modes.
<table>
<tr><th>STMSEL1</th><th>STMSEL0</th><th>Operating Mode</th><th>Activation Area</th></tr>
<tr><td>0</td><td>0</td><td>User boot mode</td><td>User area</td></tr>
<tr><td>0</td><td>1</td><td>User boot mode</td><td>User boot area</td></tr>
<tr><td>1</td><td>X</td><td>Serial programming mode</td><td>Boot area</td></tr>
</table>
X: Don't care</td></tr>
</table>

35.9.2 OPBT2 — Option Byte 2 Register

Access: This register can be read /written in 32-bit units only when using on-chip debug function.

Address: FFCD 0038$_H$

Value after reset: User defined (FFFF FFFF$_H$ at the time of shipping)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	OPJTAG1	OPJTAG0	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. For the option byte areas and the initial state at the time of shipping of this register, see **Table 35.16**. Reading and setting (R/W) of the option byte areas are possible only when using FACI commands, PG-FP5 Flash Memory Programmer, and Renesas Flash Programmer Flash Programming Software.

Table 35.18 OPBT2 Register Contents

Bit Position	Bit Name	Function
31	—	Reserved When read, the value after reset is returned. When writing to these bits, write the read value.
30, 29	OPJTAG1, OPJTAG0	Debug Interface Switching The following debug Interfaces are selected by the combination of the values (OPJTAG1 and OPJTAG0).
28 to 0	—	Reserved When read, the value after reset is returned. When writing to these bits, write the read value.

OPJTAG1	OPJTAG0	Debug interface
0	0	FLSCI3 (Programmer I/F)
0	1	LPD (4-pin)
1	0	Setting prohibited
1	1	Nexus (JTAG)

35.9.3 OPBT3 — Option Byte 3 Register

Access: This register can be read /written in 32-bit units only when using on-chip debug function.

Address: FFCD 003C$_H$

Value after reset: User defined (7FFF FFFF$_H$ at the time of shipping)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SWDT0_OPRUN	SWDT0_OPWDOVF2	SWDT0_OPWDOVF1	SWDT0_OPWDOVF0	—	—	—	—	—	SWDT0_SADU22	SWDT0_SADU21	SWDT0_SADU20	SWDT0_SADU19	SWDT0_SADU18	SWDT0_SADU17	SWDT0_SADU16
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SWDT0_SADU15	SWDT0_SADU14	SWDT0_SADU13	SWDT0_SADU12	SWDT0_SADU11	SWDT0_SADU10	SWDT0_SADU9	SWDT0_SADU8	SWDT0_SADU7	SWDT0_SADU6	SWDT0_SADU5	SWDT0_SADU4	SWDT0_SADU3	SWDT0_SADU2	SWDT0_SADU1	SWDT0_SADU0
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. For the option byte areas and the initial state at the time of shipping of this register, see **Table 35.16**. Reading and setting (R/W) of the option byte areas are possible only when using FACI commands, PG-FP5 Flash Memory Programmer, and Renesas Flash Programmer Flash Programming Software.

Table 35.19 OPBT3 Register Contents

Bit Position	Bit Name	Function
31	SWDT0_OPRUN	Enables or disables the auto-start of SWDTA0. 0: Disables the auto-start of SWDTA0 1: Enables the auto-start of SWDTA0 When SWDTA0 is not used, set this bit to 0 to disable the auto-start.
30 to 28	SWDT0_OPWDOVF2 to SWDT0_OPWDOVF0	Selects the overflow time of SWDTA0.
27 to 23	—	Reserved When read, the value after reset is returned. When writing to these bits, write the read value.
22 to 0	SWDT0_SADU22 to SWDT0_SADU0	Sets the comparison address of SWDTA0. Bit 23 to 31 of the address can be extended by 0. Any code flash area from 0000_0000$_H$ to 007F_FFFF$_H$ can be set.

SWDT0_OPWDOVF2	SWDT0_OPWDOVF1	SWDT0_OPWDOVF0	Overflow Interval Time
0	0	0	2^9 / WDTATCKI
0	0	1	2^{10} / WDTATCKI
0	1	0	2^{11} / WDTATCKI
0	1	1	2^{12} / WDTATCKI
1	0	0	2^{13} / WDTATCKI
1	0	1	2^{14} / WDTATCKI
1	1	0	2^{15} / WDTATCKI
1	1	1	2^{16} / WDTATCKI

35.9.4 OPBT4 — Option Byte 4 Register

Access: This register can be read /written in 32-bit units only when using on-chip debug function.

Address: FFCD 0040_H

Value after reset: User defined (7FFF $FFFF_H$ at the time of shipping)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SWDT1_OPRUN *2	SWDT1_OPWDOVF2 *2	SWDT1_OPWDOVF1 *2	SWDT1_OPWDOVF0 *2	—	—	—	—	—	SWDT1_SADU22 *2	SWDT1_SADU21 *2	SWDT1_SADU20 *2	SWDT1_SADU19 *2	SWDT1_SADU18 *2	SWDT1_SADU17 *2	SWDT1_SADU16 *2
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SWDT1_SADU15 *2	SWDT1_SADU14 *2	SWDT1_SADU13 *2	SWDT1_SADU12 *2	SWDT1_SADU11 *2	SWDT1_SADU10 *2	SWDT1_SADU9 *2	SWDT1_SADU8 *2	SWDT1_SADU7 *2	SWDT1_SADU6 *2	SWDT1_SADU5 *2	SWDT1_SADU4 *2	SWDT1_SADU3 *2	SWDT1_SADU2 *2	SWDT1_SADU1 *2	SWDT1_SADU0 *2
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. For the option byte areas and the initial state at the time of shipping of this register, see **Table 35.16**. Reading and setting (R/W) of the option byte areas are possible only when using FACI commands, PG-FP5 Flash Memory Programmer, and Renesas Flash Programmer Flash Programming Software.

Note 2. Only C1M-A2 has this bit; write the factory default value for C1M-A1, which does not have SWDTA1.

Table 35.20 OPBT4 Register Contents

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>31</td><td>SWDT1_OPRUN</td><td>• For C1M-A2:
Enables or disables the auto-start of SWDTA1.
0: Disables the auto-start of SWDTA1
1: Enables the auto-start of SWDTA1

• For C1M-A1:
SWDTA1 is not implemented.
Reserved
When read, the value after reset is returned.
When writing to these bits, write the read value.
When SWDTA1 is not used, set this bit to 0 to disable the auto-start.</td></tr>
<tr><td>30 to 28</td><td>SWDT1_OPWDOVF2
to
SWDT1_OPWDOVF0</td><td>• For C1M-A2:
Selects the overflow time of SWDTA1.
<table>
<tr><th>SWDT1_
OPWDOVF2</th><th>SWDT1_
OPWDOVF1</th><th>SWDT1_
OPWDOVF0</th><th>Overflow Interval Time</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>2^9 / WDTATCKI</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>2^{10} / WDTATCKI</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>2^{11} / WDTATCKI</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>2^{12} / WDTATCKI</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>2^{13} / WDTATCKI</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>2^{14} / WDTATCKI</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>2^{15} / WDTATCKI</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>2^{16} / WDTATCKI</td></tr>
</table>
• For C1M-A1:
SWDTA1 is not implemented.
Reserved
When read, the value after reset is returned.
When writing to these bits, write the read value.</td></tr>
<tr><td>27 to 23</td><td>—</td><td>Reserved
When read, the value after reset is returned.
When writing to these bits, write the read value.</td></tr>
<tr><td>22 to 0</td><td>SWDT1_SADU22
to
SWDT1_SADU0</td><td>• For C1M-A2:
Sets the comparison address of SWDTA1.
Bit 23 to 31 of the address can be extended by 0.
Any code flash area from 0000_0000_H to $007F\_FFFF_H$ can be set.

• For C1M-A1:
SWDTA1 is not implemented.
Reserved
When read, the value after reset is returned.
When writing to these bits, write the read value.</td></tr>
</table>

35.9.5 OPBT6 — Option Byte 6 Register

Access: This register can be read /written in 32-bit units only when using on-chip debug function.

Address: FFCD 0048H

Value after reset: User defined (FFFF FFFAH at the time of shipping)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	WDTCLKSEL	—	—	—	—
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	STARTUPPE	—	—	—	—	—	—	—	—	—
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1	R/W *1

Note 1. For the option byte areas and the initial state at the time of shipping of this register, see **Table 35.16**. Reading and setting (R/W) of the option byte areas are possible only when using FACI commands, PG-FP5 Flash Memory Programmer, and Renesas Flash Programmer Flash Programming Software.

Table 35.21 OPBT6 Register Contents

Bit Position	Bit Name	Function
31 to 21,	—	Reserved When read, the value after reset is returned. When writing to these bits, write the read value.
20	WDTCLKSEL	Selects the clock of the counter clock of WDTA and SWDTA (WDTATCKI). 0: LS IntOSC (240 kHz) 1: Main OSC x1/80 (250 kHz)
19 to 10	—	Reserved When read, the value after reset is returned. When writing to these bits, write the read value.
9	STARTUPPE	• For C1M-A2: Selects the boot mode of CPU. 0: After reset release, only CPU1 starts. CPU2 is in a stopped status. 1: After reset release, CPU1 and CPU2 start simultaneously. • For C1M-A1: Reserved When read, the value after reset is returned. When writing to these bits, write the read value.
8 to 0	—	Reserved When read, the value after reset is returned. When writing to these bits, write the read value.

35.10 Notes

(1) Reading areas where programming or erasure was interrupted

When programming or erasure of an area of flash memory is interrupted, the data stored in the area become undefined. To avoid undefined data that are read out becoming the source of faulty operation, take care not to fetch instructions or read data from areas where programming or erasure was interrupted.

(2) Reading the code flash memory that has been erased but not yet been programming again

Note that reading from an area of code flash memory that has been erased but not yet been programming again (i.e. that is in the non-programmed state) can lead to the detection of an ECC error and generation of the corresponding exception. Use blank checking when you need to confirm that an area is in the non-programmed state.

(3) Prohibition of additional writing

Writing to a given area twice is not possible. If you want to overwrite data in an area of flash memory after writing to the area has been completed, erase the area first.

(4) Resets during programming and erasure

In the case of an external reset during programming and erasure, wait for at least min-width of reset pulse once the operating voltage is within the range stipulated in the electrical characteristics after assertion of the reset signal before releasing the device from the reset state.

(5) Allocation of vectors for interrupts and other exceptions during programming and erasure

In the case of an interrupt or other exception during transferring an FCU firmware, or programming or erasing a code flash, mask the interrupt or exception in advance, or place an interrupt handler address table*[1] and exception handler in an instruction fetchable space other than the code flash.

Note 1. It applies in cases where the table reference method is selected as the interrupt handler address selection method. For details, see the *RH850G3MH: Software*.

(6) Abnormal termination of programming and erasure

Even if programming/erasure ends abnormally due to the generation of an external reset or instantaneous power failure, the programming/erasure state of the flash memory with undefined data cannot be verified or checked. For the area where programming or erasure ends abnormally, the blank check function cannot judge whether the area is erased successfully or not. Erase the area again to prove that the corresponding area is completely erased before using. If programming and erasure of code flash memory are not completed normally, the lock bit for the target area may be enabled (locked). In such cases, erase the block to erase the lock bit while the lock bit is in the disabled state (the area is not locked).

(7) Items prohibited during programming and erasure

Do not perform the following operations during programming and erasure of the flash memory.

- Have the operating voltage from the power supply go beyond the allowed range.
- Update the values of FHVE15 and FHVE3. Update the values of FHVE15 and FHVE3.
- Change the operating frequency of the peripheral clock.

(8) Prohibition of the various command execution of the flash memory before the completion of the clock gear up sequence

Execute the various commands in serial programming and self-programming after the completion of the clock gear up sequence. For details on the clock gear up sequence, see **Section 10, Clock Controller**.

(9) Ensuring the coherency of the instruction cache and the data buffer

After the completion of programming/erasing the code flash memory, a reset or clearance of the instruction cache and the data buffer is needed in order to ensure the coherency of the instruction cache and the data buffer. For details on the instruction cache and the data buffer, see **Section 3, CPU System**.

Section 36 Flash Security

To protect the code flash memory, data flash memory, and ID codes, this product has the security functions described in **Section 35, Flash Memory**, and functions to restrict connection of the debug interface, which are described in this section.

For detailed descriptions of the security functions in serial programming mode and how to program the flash memory, see **Section 35, Flash Memory**.

In this section, the interfaces that may be used in on-chip debugging (the Nexus and 4-pin LPD) are referred to as “debug interfaces”. In addition, ID authentication to protect the code flash memory, data flash memory, and ID codes in user boot mode is referred to as “SELF ID authentication”, and ID authentication to protect against access through the on-chip debug functions is referred to as “OCD ID authentication”.

The data length of each ID code during SELF ID and OCD ID authentication as well as the ID code in the ID authentication stage of serial programming is 128 bits. In the initial state of the product at shipment, the ID codes are FFFF_FFFF_FFFF_FFFF_FFFF_FFFF_FFFF_FFFF$_H$.

36.1 Features

36.1.1 Protection of Code Flash Memory, Data Flash Memory, and ID Codes

This microcontroller includes the following security functions in user boot mode and serial programming mode to prevent any leaking of the user program programmed in the code flash memory.

36.1.1.1 Functions Unique to User Boot Mode

SELF-ID authentication provides protections against the following operations: programming or erasure of code flash memory, rewriting to the option bytes, and rewriting or reading of ID codes.

Security States

This mode has two security states, protection locked and protection unlocked. Transition between these states can be made by changing SELF-ID authentication and the ID codes.

- Protection unlocked
  Security functions have been unlocked by SELF ID authentication so that protection against programing/erasure of code flash memory, rewriting to option bytes, and rewriting/reading of ID codes is released.
- Protection locked
  Security functions have been enabled by SELF ID authentication and protection against programing/erasure of code flash memory, rewriting to option bytes, and rewriting/reading of ID codes is in place.

36.1.1.2 Functions Unique to Serial Programming Mode

Three functions are provided as security functions unique to serial programming mode: ID authentication, prohibition of programming, erasure, and read commands, and prohibition of serial programmer connection. Parallel use of these functions is not allowed.

(1) ID Authentication
ID codes are checked for authenticity to protect the code flash memory and data flash memory. Programming, erasure, and reading of the code flash and data flash memory can proceed upon successful ID authentication.

(2) Prohibition of Programming, Erasure, and Read Commands
Issuing of programming, erasure, and read commands can be enabled or disabled individually for the code flash memory and data flash memory. All commands are permitted for both areas in the initial state of the product as shipped.

(3) Prohibition of Serial Programmer Connection
Issuing commands for programming, erasure, or reading to the code flash memory and data flash memory can be disabled in serial programming mode. Transitions to other functions are not possible once this state is set.

36.1.1.3 Common Function of User Boot Mode and Serial Programming Mode

OTP (One Time Programming)

Setting an area of the code flash memory as OTP protects it against any further programming and erasure. For details of the setting, see **Section 35, Flash Memory**.

36.1.2 Connection Restriction Function of Debug Interface

This product is provided with unauthorized access protection via debug interfaces and two security levels are available.

- Security level 1:
  Debug interfaces can be used. At this level, the on-chip debugging function is protected by OCD ID authentication. For OCD to be used, it must be unlocked by using OCD ID authentication.
- Security level 2:
  At this level, the debug interfaces cannot be used.

As described so far, restrictions on the connection of code flash and data flash memory, ID code protection, and availability of the debug interfaces differ according to the mode. **Table 36.1** summarizes security functions in each mode.

Table 36.1 Security Functions in Each Mode

Operation Mode	Code Flash and Data Flash, ID Code Protection	Restriction on Debug Interface Connection
User boot mode	• SELF ID authentication • OTP (parallel use possible)	• Security level 1 (OCD ID authentication) • Security level 2 (Debug interface connection is prohibited)
Serial programming mode	• ID authentication • Programming commands, block erasure commands, and read commands are prohibited. • Connection of serial programmers is prohibited. (The above three cannot be used in parallel.) • OTP (parallel use possible)	• No function (Debug interface connection is always prohibited.)

36.2 Security in User Boot Mode

36.2.1 SELF ID Authentication

To prevent leakage of the user program written to the code flash memory, this product has security functions to enable or disable the programming/erasure of code flash memory, rewriting to option bytes, and rewriting/reading of ID codes. The setting of enabling or disabling of these functions can be switched through SELF ID authentication with the ID code the user has set as the expected value.

36.2.2 SELF ID Authentication and Security State

Table 36.2 and **Figure 36.1** show SELF ID authentication security state and conditions for transition.

Table 36.2 Security State Set by SELF ID Authentication

State	SELF ID Authentication	Protection State of Reprogramming/Erasure of Code Flash, Rewriting to Option Bytes, Rewriting/Reading of ID Codes.
Protection unlocked	Unlocked	Not protected
Protection locked	Locked	Protected

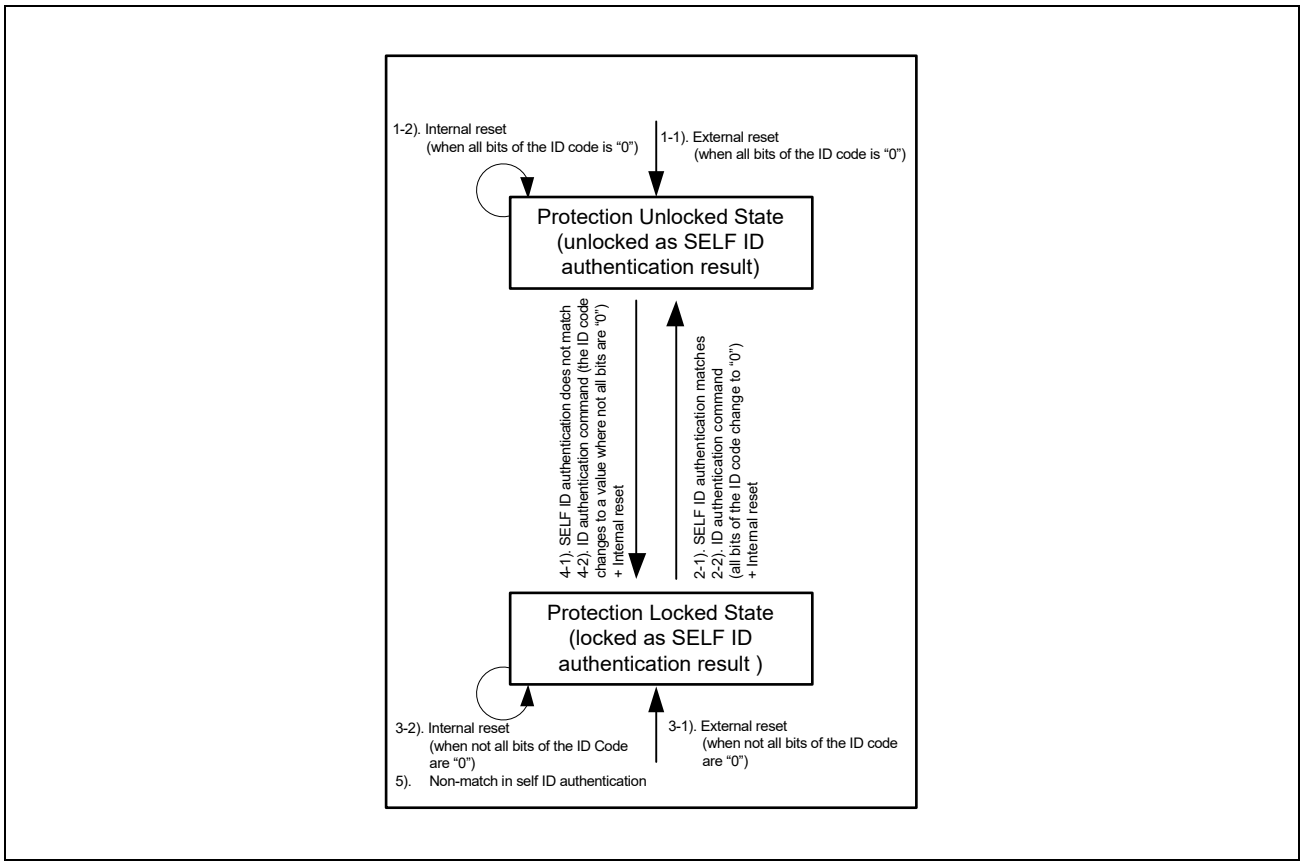


Figure 36.1 Security Setting State by SELF ID Authentication and Transition

The conditions for the transitions of security states are shown in **Figure 36.1** as follows.

- Conditions for Transitions to the Protection Unlocked State

1) Startup when protection is released

1-1) When an external reset is applied while the ID code is all 0s, startup proceeds with protection released.

1-2) When an internal reset is applied without modification of the ID code while all bits of the ID code are 0, startup proceeds with protection in place.

2) Transition from protection locked to protection released

2-1) A transition to the protection unlocked state follows a match in SELF ID authentication matches.

2-2) When an internal reset is applied after changing the ID code has been changed from other than “all bits 0” to “all bits 0”, startup proceeds with protection released.

- Conditions for Transitions to Protection Locked

3) Conditions for transitions to protection locked

3-1) When an external reset is applied while the ID code is other than “all bits 0”, startup proceeds with protection in place.

3-2) When an internal reset is applied without modification of the ID code while not all bits of the ID code are 0, startup proceeds with protection in place.

4) Transition from protection unlocked to protection locked

4-1) A transition to the protection locked state follows a non-match in SELF ID authentication.

4-2) When an internal reset is applied after the ID code has been changed from “all bits 0” to “other than all bits 0”, startup proceeds with protection in place.

5) To retain protection locked

Protection remains in place in case of a non-match in SELF ID authentication.

36.3 Security Functions in Serial Programming Mode

For details of security functions in serial programming mode, see **Section 35, Flash Memory**.

36.4 Restricting Connection with Debug Interfaces

This product is capable of restricting connection of the debug interfaces to protect against unauthorized access via the debug interfaces. This feature has two security levels.

- Security level 1: Restrict access to OCD functions by using OCD ID authentication
- Security level 2: Prohibit all connection to the debug interfaces

These security levels can be changed by bits 30 and 29 (OPJTAG1 and OPJTAG0) in option byte 2 in the extended area of the flash memory.
In this section, OPJTAG0 and OPJTAG1 are referred to as the OPJTAG bits.

36.4.1 Security Levels and State of Restricting the Connection of Debug Interfaces

Table 36.3 show each security level and the corresponding security states and **Figure 36.2** shows the conditions for transitions.

Table 36.3 Security Levels and State of Restricting Connection to the Debug Interfaces

State	Result of OCD ID Authentication	OPJTAG Bit*1	Restriction on Debug Interface Connection
Security level 1	Unlocked	Other than 00_B	No restriction
	Locked	Other than 00_B	Restriction on access via the debug interfaces is in place.
Security level 2	—	00_B	Connection to the debug interfaces is prohibited.

Note 1. For a detailed description of the OPJTAG bits, see **Section 35, Flash Memory**.

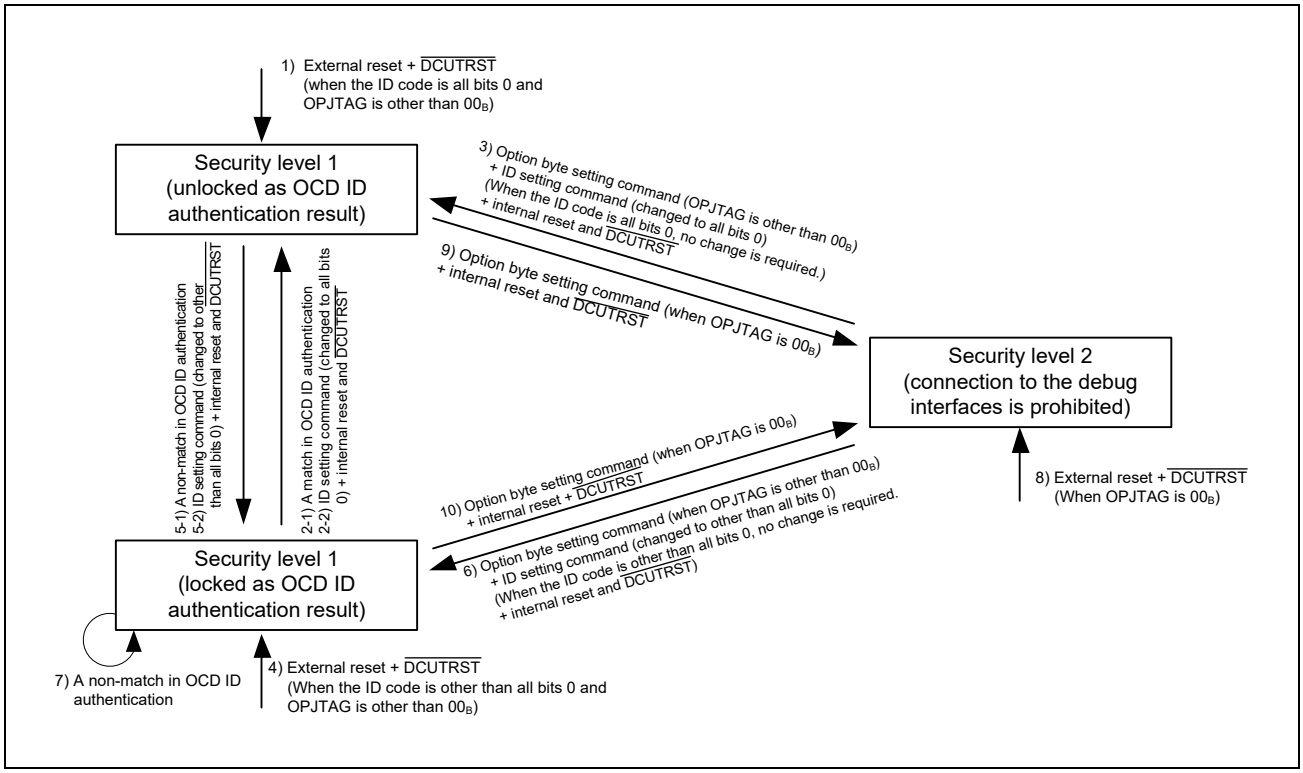


Figure 36.2 Transitions of Security Level

Conditions for transitions to each security level described in **Figure 36.2**.

- Conditions for Transitions to Security Level 1 (unlocked as OCD ID authentication result)

1) Startup in the state of security level 1 (unlocked as OCD ID authentication result)

When an external reset and a reset by $\overline{\text{DCUTRST}}$ are applied while the ID code is all 0s and while the value of the OPJTAG is other than 00_B, startup proceeds with security level 1 (unlocked as OCD ID authentication result).

2) Transition from security level 1 (locked as OCD ID authentication result) to security level 1 (unlocked as OCD ID authentication result)

2-1) Transition to security level 1 (unlocked as OCD ID authentication result) follows a match in OCD ID authentication.

2-2) When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the ID code to “all bits 0”, startup proceeds at security level 1 (unlocked as OCD ID authentication result).

3) Transition from security level 2 to security level 1 (unlocked as OCD ID authentication result)

When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the ID code to “all bits 0” and the value of the OPJTAG to other than 00_B, startup proceeds at security level 1 (unlocked as OCD ID authentication result).

- Conditions for Transitions to Security Level 1 (locked as OCD ID Authentication result)

4) Startup in the state of security level 1 (locked as OCD ID authentication result)

When an external reset or a reset by $\overline{\text{DCUTRST}}$ is applied while the ID code is “other than all bits 0” and the value of the OPJTAG is other than 00_B, startup proceeds at security level 1 (locked as OCD ID authentication result).

5) Transition from security level 1 (unlocked as OCD ID authentication result) to security level 1 (locked as OCD ID authentication result)

5-1) A transition to security level 1 (locked as OCD ID authentication result) follows a non-match in OCD ID authentication.

5-2) When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the ID code to “other than all bits 0”, startup proceeds at security level 1 (locked as OCD ID authentication result).

6) Transition from security level 2 to security level 1 (locked as OCD ID authentication result)

When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the ID code to “other than all bits 0” and the value of OPJTAG is other than 00_B, startup proceeds at security level 1 (locked as OCD ID authentication result).

7) Retaining security level 1 (locked as OCD ID authentication result)

In case of a non-match in OCD ID authentication, security level 1 (locked as OCD ID authentication result) remains in place.

- Transition to Security Level 2

  8) Startup at security level 2

     When an external reset or a reset by $\overline{\text{DCUTRST}}$ is applied while the value of OPJTAG is 00_B, startup proceeds at security level 2.

  9) Transition from security level 1 (unlocked as OCD ID authentication result) to security level 2

     When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the value of the OPJTAG to 00_B, startup proceeds at security level 2.

  10) Transition from security level 1 (locked as OCD ID authentication result) to security level 2

      When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the value of the OPJTAG to 00_B, startup proceeds at security level 2.

Section 37 RAM

37.1 List of On-Chip RAM

RH850/C1M-A includes the following RAM.

- Local RAM (CPU1): 64 Kbytes
- Local RAM (CPU2): 64 Kbytes (C1M-A2)
- Local RAM (EMU): 64 Kbytes
- Global RAM: Up to 128 Kbytes*1

Note 1. Refer to **Table 1.1, Overview of Products** for including size in each product.

37.2 Features

Access:

CPU1, CPU2, the sub-CPU in EMU3, and the DMAC can access the local RAM of the CPUs (CPU1, CPU2, sub-CPU in EMU3) and the global RAM.

For details of address map and access availability, see **Section 4, Address Space**.

ECC:

The local RAM of CPU1, CPU2, and the sub-CPU in EMU3 and the global RAM include ECC and address parity.

For details, see **Section 29, Functional Safety**.

37.3 Notes

(1) The local RAM and global RAM must be initialized with the maximum bit length of its access size before using the RAM.

If the RAM is accessed before its initialization, an ECC error may be detected. Also if initialization with the maximum bit length is not performed (e.g. if a 32-bit RAM is initialized by an 8-bit or 16-bit access), an ECC error may be detected.

(2) Buffers are included for high-speed access between the following RAMs and the CPU.

- Local RAM (CPU1), Local RAM (CPU2), Local RAM (EMU)

  After a value has been written to an address, reading the value from the same address may lead to reading from the buffer instead of the RAM.

  Ways to make sure that values are being read from the RAM are as follows.

  1. Only read the first byte to have been written after having written more than 32 bytes of data.
  2. Issue a SYNCM instruction between a write instruction and an instruction to read from the same address.

Section 38 Boundary Scan

This section contains a generic description of the boundary scan function.

RH850/C1M-A has the JTAG interface and provides the functionality of boundary scan.

38.1 Overview

Boundary scan is a test method specified in IEEE 1149.1 standard used for testing the connection between devices mounted on a printed-circuit board. The boundary scan function of RH850/C1M-A conforms to the IEEE 1149.1-2001 standard.

38.2 Features

- Five test signals (DCUTCK, DCUTDI, DCUTDO, DCUTMS, and $\overline{\text{DCUTRST}}$)
- TAP controller
- Instruction register
- Bypass register
- Boundary scan register

The JTAG interface has six commands.

- BYPASS mode
  Test mode conforming to the IEEE 1149.1
- EXTEST mode
  Test mode conforming to the IEEE 1149.1
- SAMPLE/PRELOAD mode
  Test mode conforming to the IEEE 1149.1
- CLAMP mode
  Test mode conforming to the IEEE 1149.1
- HIGHZ mode
  Test mode conforming to the IEEE 1149.1
- IDCODE mode
  Test mode conforming to the IEEE 1149.1

Figure 38.1 shows a block diagram of the JTAG interface.

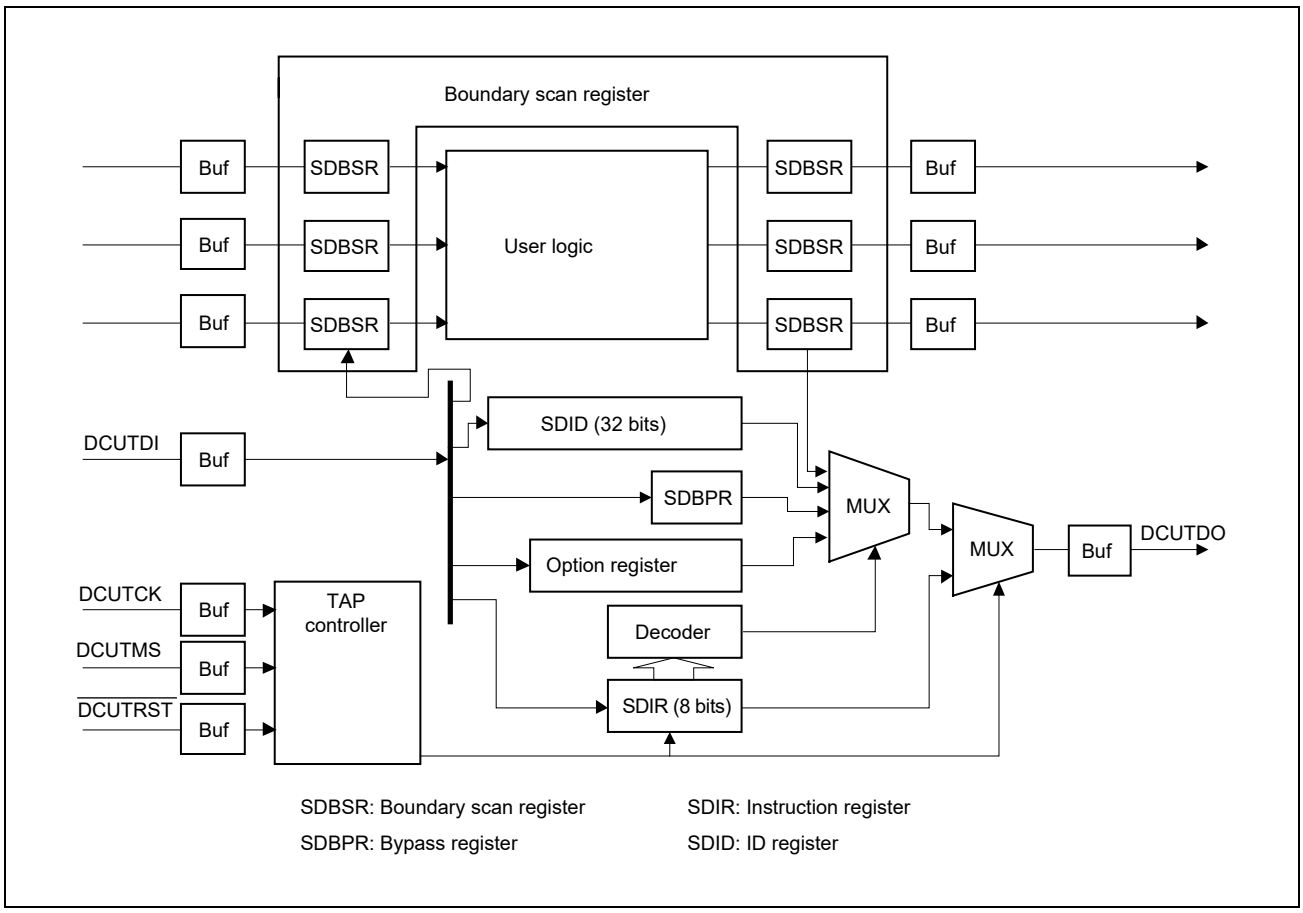


Figure 38.1 Block Diagram of JTAG Interface

38.3 Input/Output Pins

There are five JTAG control signals: DCUTCK, DCUTDI, DCUTMS, DCUTDO, and $\overline{\text{DCUTRST}}$.

Table 38.1 shows the pin configuration.

Table 38.1 Pin Configuration

Pin Name	Description
DCUTCK	Serial data input/output clock pin Data is supplied via the data input pin (DCUTDI) and is output from the data output pin (DCUTDO) in synchronization with this clock signal.
DCUTMS	Mode select input pin Changing the level of this signal in synchronization with DCUTCK changes the state of the TAP controller. For the protocol, refer to **Figure 38.2, TAP Controller State Transition Diagram**.
$\overline{\text{DCUTRST}}$	Reset input pin A low-level input of this signal, which is accepted asynchronously with DCUTCK, resets the JTAG interface. Even if the JTAG interface is not used, $\overline{\text{DCUTRST}}$ must be held low for the specified time at a power on.
DCUTDI	Serial data input pin Changing the state of this pin in synchronization with DCUTCK allows data to be sent to the JTAG interface.
DCUTDO	Serial data output pin Reading the state of this pin in synchronization with DCUTCK allows data to be read from the JTAG interface.

38.4 Register Descriptions

The JTAG interface has the following registers. None of the registers can be accessed by the CPU.

- SDIR: Instruction register
- SDID: ID register
- SDBPR: Bypass register
- SDBSR: Boundary scan register

Table 38.2 Register Configuration

Register Name	Symbol	Access Size	Initial Value*1
Instruction register	SDIR	8	55_H
ID register	SDID	32	1832 $B447_H$ (C1M-A2) 1832 $D447_H$ (C1M-A1)
Bypass register	SDBPR	1	Undefined
Boundary scan register	SDBSR	—	Undefined

Note 1. Registers are initialized when $\overline{\text{DCUTRST}}$ pin is 0 or when TAP is in the Test-Logic-Reset state.

Commands can be serially transferred from the serial data input pin (DCUTDI) and input to the instruction register (SDIR). The bypass register (SDBPR) is a 1-bit register, to which DCUTDI and DCUTDO are connected in BYPASS mode, CLAMP mode, and HIGHZ mode. The boundary scan register (SDBSR) is connected DCUTDI and DCUTDO in SAMPLE/PRELOAD mode and EXTEST mode. The ID code register (SDID) is a 32-bit register, from which the ID code is output via DCUTDO in IDCODE mode.

Table 38.3 shows the serial transfer types possible with the JTAG interface registers.

Table 38.3 Serial Transfer Types Possible with JTAG Interface Registers

Register	Serial Input	Serial Output
SDIR	Possible	Impossible *1
SDBPR	Possible	Possible
SDBSR	Possible	Possible
SDID	Impossible	Possible

Note 1. A fixed value is read out.

38.4.1 Instruction Register (SDIR)

SDIR is an 8-bit register that holds a boundary scan command. SDIR is initialized by asserting $\overline{\text{DCUTRST}}$ or in the TAP Test-Logic-Reset state. Operation is not guaranteed when any reserved command is set in this register.

Table 38.4 Boundary Scan Commands

IR Code								Description
0	0	0	0	0	0	0	0	JTAG EXTEST
0	1	0	0	0	0	0	0	JTAG SAMPLE/PRELOAD
1	1	0	1	0	0	0	0	JTAG CLAMP
1	0	0	0	0	0	0	0	JTAG HIGHZ
0	1	0	1	0	1	0	1	JTAG IDCODE (value after reset)
1	1	1	1	1	1	1	1	JTAG BYPASS
Other than above								Reserved

38.4.2 SDID — ID Register

SDID is a 32-bit register that indicates an LSI-specific ID.

SDID can be read via the JTAG interface pin when the IDCODE command is set but cannot be written to.

For the read values, refer to **Table 38.2, Register Configuration**.

38.4.3 SDBPR — Bypass Register

SDBPR is a 1-bit register. When SDIR is set to BYPASS mode, SDBPR is connected between DCUTDI and DCUTDO. The value after reset is undefined. SDBPR is not initialized by a power-on reset or asserted $\overline{\text{DCUTRST}}$.

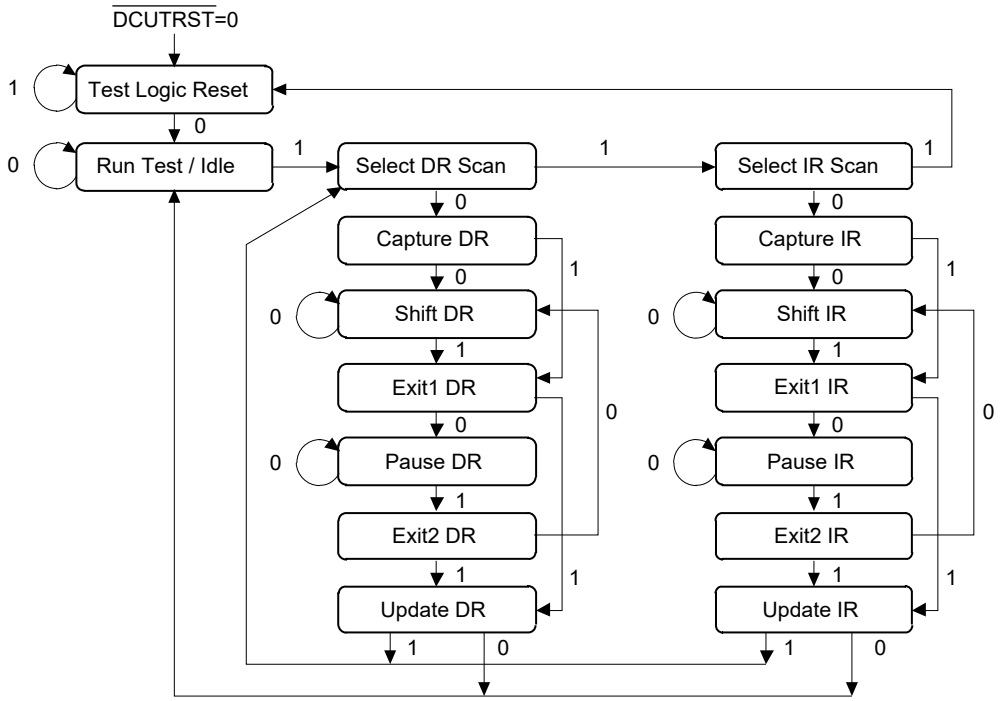
38.4.4 SDBSR — Boundary Scan Register

SDBSR is a shift register located on the PADs for controlling the external I/O pins. When SDIR is set to SAMPLE/PRELOAD or EXTEST mode, SDBSR is connected between DCUTDI and DCUTDO. The value after reset is undefined. SDBSR is not initialized by a power-on reset or asserted $\overline{\text{DCUTRST}}$.

38.5 Operation

38.5.1 TAP Controller

Figure 38.2 shows the TAP controller internal states.



Note: Transition is made according to the DCUTMS value at the rising edge of DCUTCK. The DCUTDI value is sampled at the rising edge of DCUTCK and is shifted at the falling edge. DCUTDO is in the high-impedance state in the states other than Shift-DR and Shift-IR. Asserting $\overline{\text{DCUTRST}}$ causes transition to Test-Logic-Reset state asynchronously with DCUTCK.

Figure 38.2 TAP Controller State Transition Diagram

38.5.2 Supported Commands

38.5.2.1 BYPASS

The BYPASS command is a standard command indispensable to bypass register operation. This command shortens the shift path to achieve high-speed serial data transfer of other LSIs on the printed-circuit board. During execution of this command, the test circuit has no effect on the system circuit.

38.5.2.2 SAMPLE/PRELOAD

The SAMPLE/PRELOAD command is used to input the value to the boundary scan register from the internal circuits of this LSI; to output the value from the scan path; and to load data onto the scan path. During execution of this command, the level of the input pin of this LSI is sent to the internal circuits as is, and the value of the internal circuits is output to the outside via the output pin as is. Executing this command has no effect on the system circuit of this LSI.

The SAMPLE operation allows taking in the snapshots of the value to be transferred to the internal circuits from the input pin or the value to be transferred to the output pin from the internal circuits to the boundary scan register and allows reading the snapshots from the scan path. Snapshots can be taken in without preventing the normal operation of this LSI.

The PRELOAD operation allows setting the value after reset to the parallel output latch of the boundary scan register from the scan path prior to the EXTEST command. If the EXTEST command is executed without PRELOAD operation, the undefined value is output from the output pin until the first scan sequence is completed (transfer to the output latch) because the parallel output latch value is always output to the output pin with the EXTEST command.

38.5.2.3 EXTEST

The EXTEST command is used to test the external circuits when this LSI is mounted on the printed-circuit board. When this command is executed, the output pin is used to output the test data (previously set with the SAMPLE/PRELOAD command) from the boundary scan register to the printed-circuit board; whereas the input pin is used to take in the test result from the printed-circuit board to the boundary scan register. When the EXTEST command is executed N times for testing, the test data for the Nth execution is scanned in at the (N - 1)th scan-out.

If the data is loaded onto the boundary scan register of the output pin in the Capture-DR state of this command, it is not used for testing the external circuits (replaced through shift operation).

38.5.2.4 CLAMP

When the CLAMP command is selected, the output pin outputs the boundary scan register value that has been previously set with the SAMPLE/PRELOAD command. While the CLAMP command is selected, the boundary scan register retains the previous state regardless of the TAP controller state.

The bypass register is connected between DCUTDI and DCUTDO and operates in the same manner as when the BYPASS command is selected.

38.5.2.5 HIGHZ

When the HIGHZ command is selected, all the output pins go to the high-impedance state. While the HIGHZ command is selected, the boundary scan register retains the previous state regardless of the TAP controller state.

The bypass register is connected between DCUTDI and DCUTDO and operates in the same manner as when the BYPASS command is selected.

38.5.2.6 IDCODE

The IDCODE command sets the JTAG interface pins to IDCODE mode, which is defined by the JTAG standard. When the JTAG interface is initialized (by asserting $\overline{\text{DCUTRST}}$ or placing TAP in the Test-Logic-Reset state), IDCODE mode is set.

38.5.3 Notes

There are restrictions related to the JTAG interface as follows.

- The power supply/GND pins are not subjected to boundary scan.
- The reference voltage pins of the A/D converter (A0VREFH, A1VREFH, and A2VREFH) are not subjected to boundary scan.
- The NC pins are not subjected to boundary scan.
- **Table 38.5** lists the pins not subjected to boundary scan.

Table 38.5 Pins Not Subjected to Boundary Scan

Type	Pins
Analog input	ADCC0I00, ADCC0I01, ADCC0I02, ADCC0I03, ADCC0I10, ADCC0I11, ADCC0I12, ADCC0I13, ADCC0I20, ADCC0I21, ADCC0I22, ADCC0I23, ADCC0I30, ADCC0I31, ADCC0I32, ADCC0I33, ADCC1I00, ADCC1I01, ADCC1I02, ADCC1I03, ADCC1I10, ADCC1I11, ADCC1I12, ADCC1I13, ADCC1I20, ADCC1I21, ADCC1I22, ADCC1I23, ADCC1I30, ADCC1I31, ADCC1I32, ADCC1I33, ADCC2I00, ADCC2I01, ADCC2I02, ADCC2I03, ADCC2I10, ADCC2I11, ADCC2I12, ADCC2I13, ADCC2I20, ADCC2I21, ADCC2I22, ADCC2I23, ADCC2I30, ADCC2I31, ADCC2I32, ADCC2I33, RDC3A0COM, RDC3A1COM, RDC3A0RSO, RDC3A1RSO, RDC3A0S1, RDC3A0S2, RDC3A0S3, RDC3A0S4, RDC3A1S1, RDC3A1S2, RDC3A1S3, RDC3A1S4
General-purpose input	P7_8
Debug system	AUDCK, $\overline{\text{DCURDY}}$, DCUDCK, DCUTDI, DCUTDO, DCUTMS, $\overline{\text{DCUTRST}}$
Mode setting	MD0, MD1, FLMODE
Clock, reset	X1, X2, $\overline{\text{RESET}}$
Error output	ERROROUT_M

- The HIGHZ command is invalid for the pulled-down pins.

38.6 Usage Notes

1. Once a command is set, it is not modified until another command is issued again. To continuously issue the same commands, insert a command that has no effect on chip operation (such as BYPASS mode) between the desired commands.
2. To start the system in boundary scan mode, negate $\overline{\text{DCUTRST}}$ while $\overline{\text{RESET}}$ is high.
3. For the maximum clock frequency that can be input to DCUTCK, refer to **Section 39, Electrical Characteristics**.
4. If the number of serially transferred bits exceeds the number of bits of the register connected between DCUTDI and DCUTDO, the data that is input from DCUTDI is output from DCUTDO after the serial data equal to the number of register bits are output.
5. If the serial transfer sequence is corrupted, be sure to reset $\overline{\text{DCUTRST}}$. Here, start the transfer over again regardless of the point of transfer corruption.
6. Data is output via DCUTDO at the falling edge of DCUTCK.
7. To facilitate debugging, route $\overline{\text{DCUTRST}}$ on the board in such a way that patterns can be easily cut.

Section 39 Electrical Characteristics

39.1 Absolute Maximum Ratings

Table 39.1 lists absolute maximum ratings.

Table 39.1 Absolute Maximum Ratings

Item		Symbol	Rated Value	Unit	Note
Power voltage*1	SYSVCC, VCC	VCC	– 0.3 to + 6.5	V	
	VDD	VDD	– 0.3 to + 1.8	V	
Input voltage	SYSVCC power pin	Vin	– 0.3 to SYSVCC + 0.3	V	See **Table 39.2** for intended pins
	VCC power pin	Vin	– 0.3 to VCC + 0.3	V	
Analog power voltage		A0VCC, A1VCC, A2VCC	– 0.3 to + 6.5	V	
		RVCC	– 0.3 to + 6.5	V	
Analog reference voltage		A0VREFH	– 0.3 to A0VCC + 0.3	V	
		A1VREFH	– 0.3 to A1VCC + 0.3	V	
		A2VREFH	– 0.3 to A2VCC + 0.3	V	
Analog input voltage		V_{AIN}	– 0.3 to A0VCC + 0.3 – 0.3 to A1VCC + 0.3 – 0.3 to A2VCC + 0.3	V	
		V_{RIN}	– 0.3 to RVCC + 0.3	V	
VSS differential voltage (Condition: Between any two of VSS, A0VSS, A1VSS, A2VSS, and RVSS)			– 0.1 to + 0.1	V	
Maximum input current (per pin)	Digital input pin	Imax	– 25 to + 25	mA	Only 1 pin simultaneously
	Analog input pin	Imax	– 25 to + 25	mA	
Junction temperature*1		Tj	– 40 to + 150	ºC	
Storage temperature		Tstg	– 55 to + 150	ºC	After installation

Note 1. Cumulative hours of operation of this LSI with Tj in the range from 125°C to 150°C must be kept within 3000

NOTE

Using this LSI without observing these absolute maximum ratings may result in permanent breakdown of the LSI.

This product is used in combination of multiple power voltages simultaneously in some cases. Use this LSI conforming to power pin connections, conditions for combination of power voltages to be applied, voltages that can be applied to pins, and output voltage conditions, which are specified in the manual. Using this LSI with unspecified power connection or voltage may result in permanent breakdown of the LSI or damage to the system that contains this LSI.

Input voltage, analog reference voltage and analog input voltage must not exceed 6.5 V.

39.2 DC Characteristics

39.2.1 Relationship between Power Name and Pin

Table 39.2 shows the relationship between power name and pin.

Table 39.2 Relationship between Power Name and Pin

Pin Name (Initial Value)	Circuit Power Name	I/O	Input Buffer Type	Note
Px_x	VCC	I/O	Schmitt B	Variable driving ability
P7_8	SYSVCC	I	Schmitt A	
ADCC0Ixx	A0VCC	I*1	Analog (ADC)	
ADCC1Ixx	A1VCC	I	Analog (ADC)	
ADCC2Ixx	A2VCC	I	Analog (ADC)	
RDC3AnSx*2	RVCC	I	Analog (RDC)	
RDC3AnRSO*2	RVCC	IO	—	
RDC3AnCOM*2	RVCC	IO	—	
$\overline{\text{RESET}}$	SYSVCC	I	Schmitt A	
FLMODE	SYSVCC	I	Schmitt A	
MD0	SYSVCC	I	Schmitt A	
MD1	SYSVCC	I	Schmitt A	
$\overline{\text{ERROROUT\_M}}$	VCC	O	—	
X1	VCC	I	CMOS	
X2	VCC	O	—	
$\overline{\text{AUDRST}}$	VCC	I	Schmitt A	
AUDCK	VCC	I	CMOS	
AUDSYNC	VCC	I	CMOS	
AUDATAx	VCC	I/O	CMOS	
$\overline{\text{DCUTRST}}$ / $\overline{\text{LPDTRST}}$	VCC	I	Schmitt A	
DCUTDO/LPDO	VCC	O	—	
DCUTMS	VCC	I	CMOS	
DCUTCK/LPDCLK	VCC	I	CMOS	
DCUTDI/LPDI	VCC	I	CMOS	
$\overline{\text{DCURDY}}$ /LPDCLKOUT	VCC	O	—	

Note 1. Some pins also serve as RDC3A pins. Following pins are I/O for both functions.
ADCC1I00/RDC3A0SINMNT, ADCC1I01/RDC3A0COSMNT, ADCC0I12/RDC3A1SINMNT, ADCC0I13/RDC3A1COSMNT

Note 2. C1M-A2: n = 0-1, C1M-A1: n = 0

NOTE

A through current may develop within the pin when an intermediate potential is applied in a DC manner, even if the input buffer is of the Schmitt buffer type (Schmitt A, B).

39.2.2 Recommended Operating Conditions

Table 39.3 Recommended Operating Conditions

Symbol	Min.	Typ.	Max.	Unit	Note
SYSVCC	4.5	5.0	5.5	V	—
VCC	4.5	5.0	5.5		—
VDD	1.15	1.25	1.35		—
A0VCC, A1VCC,A2VCC*1	4.5	5.0	5.5		—
RVCC*1	4.5	5.0	5.5		—
A0VREFH, A1VREFH, A2VREFH*2	4.5	5.0	5.5		—

Note: VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V must be kept.

Note 1. A0VCC, A1VCC, A2VCC, and RVCC must be connected to the same electric potential.

Note 2. Do not exceed A0VCC, A1VCC, or A2VCC.

39.2.3 Input Voltage Characteristics

Table 39.4 DC Characteristics (Input Voltage)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = – 40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Schmitt trigger input voltage (Buffer type A)	SYSVCC VCC	V_T^+ (V_{IH})	SYSVCC × 0.75 VCC × 0.75	—	SYSVCC + 0.3 VCC + 0.3	V	**Table 39.2** (Item of Schmitt A input buffer type)
		V_T^- (V_{IL})	–0.3	—	SYSVCC × 0.25 VCC × 0.25	V	
		V_{HS}	SYSVCC × 0.2 VCC × 0.2	—	—	V	
Schmitt trigger input voltage (Buffer type B)	VCC	V_T^+ (V_{IH})	VCC × 0.7	—	VCC + 0.3	V	**Table 39.2** (Item of Schmitt B input buffer type)
		V_T^- (V_{IL})	–0.3	—	VCC × 0.42	V	
		V_{HS}	VCC × 0.082	—	—	V	
CMOS input voltage	VCC	V_{IH}	VCC × 0.7	—	VCC + 0.3	V	**Table 39.2** (Item of CMOS input buffer type)
		V_{IL}	–0.3	—	VCC × 0.2	V	

39.2.4 Input Leak Current Characteristics

Table 39.5 DC Characteristics (Input Leak Current)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input leak current	Other than A/D port and R/D port*1	I lin I	—	—	1	µA	Vin = 0 V to SYSVCC Vin = 0 V to VCC
	A/D port	I lin I	—	—	0.1	µA	Vin = 0 V to A0VCC, A1VCC, A2VCC
	R/D port	I lin I	—	—	0.3	µA	Vin = 0 V to RVCC, and stopping RDC3A

Note 1. X1 pin is not intended. Pull-up/pull-down pins are also not intended.

39.2.5 Pull-Up/Pull-Down MOS Current Characteristics

Table 39.6 DC Characteristics (Pull-Up/pull-Down MOS Current)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input pull-up MOS current	DCUTMS, DCUTCK, DCUTDI	I_{pu}	—	—	350	µA	Vin = 0 V VCC = 4.5 to 5.5 V
	AUDCK, $\overline{\text{AUDSYNC}}$, AUDATA3-0*1		—	—	350	µA	Vin = 0 V VCC = 4.5 to 5.5 V
	General port		—	—	350	µA	Vin = 0 V, VCC = 4.5 to 5.5 V
Input pull-down MOS current	$\overline{\text{RESET}}$	I_{pd}	25	160	350	µA	Vin = SYSVCC = 4.5 to 5.5 V
	$\overline{\text{DCUTRST}}$		—	—	350	µA	Vin = SYSVCC = 4.5 to 5.5 V
	FLMODE, MD0, MD1		15	—	350	µA	Vin = SYSVCC = 4.5 to 5.5 V
	$\overline{\text{AUDRST}}$		—	—	350	µA	Vin = VCC = 4.5 to 5.5 V
	General port		—	—	350	µA	Vin = VCC = 4.5 to 5.5 V

Note 1. The pull-up of AUDATA3-0 is valid not only in input but also in output.

39.2.6 Output Voltage Characteristics

Table 39.7 DC Characteristics (Output Voltage)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Output high level voltage	VCC power supply system pins	V_{OH}	VCC – 0.5	—	—	V	I_{OH} = 200 µA VCC = 4.5 to 5.5 V
			VCC – 1.0	—	—	V	I_{OH} = 1 mA VCC = 4.5 to 5.5 V
Output low level voltage	VCC power supply system pins	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA VCC = 4.5 to 5.5 V
			—	—	1.2	V	I_{OL} = 4 mA VCC = 4.5 to 5.5 V

39.2.7 Allowable Output Current

Table 39.8 DC Characteristics (Allowable Output Current)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Output low-level allowable current (per pin)	IOL	—	—	4.0	mA	
Output low-level allowable current (total)	ΣIOL	—	—	80.0	mA	
Output high-level allowable current (per pin)	IOH	—	—	2.0	mA	
Output high-level allowable current (total)	ΣIOH	—	—	25.0	mA	

This item affects the calorific value and Tj of the chip. In addition to these restrictions, you also need to take thermal design into consideration.

39.2.8 Injection Current

Table 39.9 DC Characteristics (Injection Current)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit
DC injection current (per pin)	Logic pin	IIC	–2.0	—	2.0	mA
	Analog pin*1		–3.0	—	3.0	mA
DC injection current (total)		Σ|IIC|	–50.0	—	50.0	mA

This item affects the calorific value and Tj of the chip. In addition to these restrictions, you also need to take thermal design into consideration.

Note 1. The objects are ADCCn pins. However, the following pins are excluded:
ADCC1I00, ADCC1I01, ADCC0I12, ADCC0I13

39.2.9 Input Capacitance

Table 39.10 DC Characteristics (Input Capacitance)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input capacitance	All pins	Cin	—	10	20	pF	Vin = 0 V, f = 1 MHz Tj = 25ºC

39.2.10 Supply Current Characteristics

Table 39.11 DC Characteristics (Supply Current: C1M-A2)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Core supply current	Normal operation	I_{dd}	—	—	730	mA	
(VDD power supply)	Current during reset	I_{ddrst}	—	—	335	mA	
VCC power supply current	Normal operation (excluding erasure of code flash)	I_{CC}	—	—	25	mA	Excluding I/O current
	Erasure of code flash	$I_{cc\_cferase}$	—	—	60	mA	Excluding I/O current, *1
	Current during reset	I_{ccrst}	—	—	15	mA	
System supply current	Normal operation	I_{SYS}	—	—	7	mA	Including PLL current
(SYSVCC power supply)	Current during reset	I_{sysrst}	—	—	3	mA	
Analog power supply current (A0VCC, A1VCC, A2VCC power supply)		I_{AVCC}	—	—	30	mA	
Analog power supply current (RVCC power supply)		I_{RVCC}	—	—	20	mA	
ADC reference power supply current (A0VREFH, A1VREFH, A2VREFH)		I_{AVREF}	—	—	0.5	mA	

Note 1. Large fluctuations in current occur during erasure. The average current is 15 mA.

CAUTIONS

1. When the A/D converter is not used or it is in the standby state, do not open the A0VCC pin, A1VCC pin, A2VCC pin, A0VREFH pin, A1VREFH pin, A2VREFH pin, A0VSS pin, A1VSS pin, and A2VSS pin.
2. Supply current values are those measured when VIHmin = VCC – 0.5 V and VIL = 0.5 V with no load applied to all output pins.

Table 39.12 DC Characteristics (Supply Current: C1M-A1)

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Core supply current	Normal operation	I_{dd}	—	—	540	mA	
(VDD power supply)	Current during reset	I_{ddrst}	—	—	335	mA	
VCC power supply current	Normal operation (excluding erasure of code flash)	I_{CC}	—	—	25	mA	Excluding I/O current
	Erasure of code flash	$I_{cc\_cferase}$	—	—	60	mA	Excluding I/O current, *1
	Current during reset	I_{ccrst}	—	—	15	mA	
System supply current	Normal operation	I_{SYS}	—	—	7	mA	Including PLL current
(SYSVCC power supply)	Current during reset	I_{sysrst}	—	—	3	mA	
Analog power supply current (A0VCC, A1VCC, A2VCC power supply)		I_{AVCC}	—	—	30	mA	
Analog power supply current (RVCC power supply)		I_{RVCC}	—	—	10	mA	
ADC reference power supply current (A0VREFH, A1VREFH, A2VREFH)		I_{AVREF}	—	—	0.5	mA	

Note 1. Large fluctuations in current occur during erasure. The average current is 15 mA.

CAUTIONS

1. When the A/D converter is not used or it is in the standby state, do not open the A0VCC pin, A1VCC pin, A2VCC pin, A0VREFH pin, A1VREFH pin, A2VREFH pin, A0VSS pin, A1VSS pin, and A2VSS pin.
2. Supply current values are those measured when VIHmin = VCC – 0.5 V and VIL = 0.5 V with no load applied to all output pins.

39.3 AC Characteristics

Unless otherwise described, the following timing conditions are applied.

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

- In the port control register, conditions where all output pins of the module used in the same channel are set to the same driving ability are applied to output pins whose driving ability is selectable. Unless otherwise specified, all driving ability settings are included.

- Unless otherwise described, AC measurement conditions described in the figure below are applied.

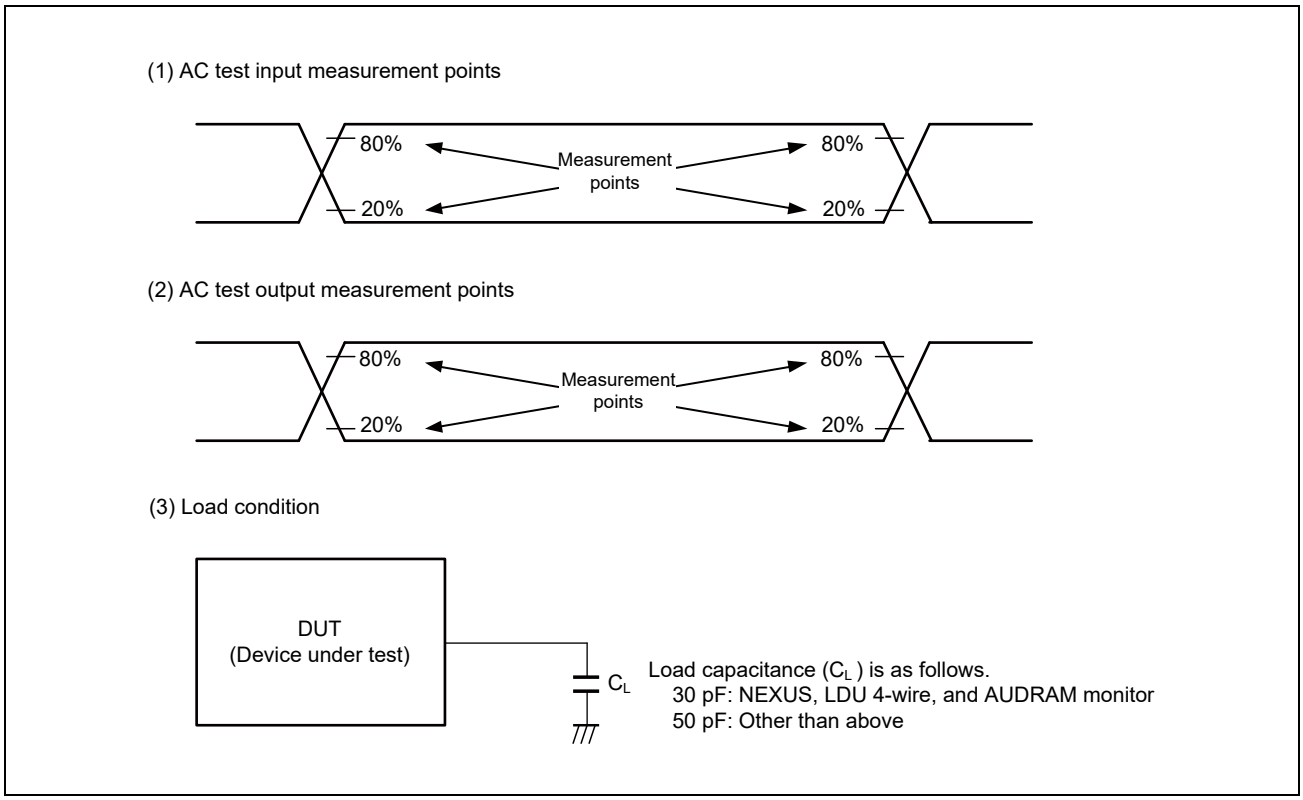


Figure 39.1 AC Measurement Conditions

39.3.1 Power On/Off Timings

Table 39.13 Power On/Off Timings

Item	Symbol	Min.	Max.	Unit	Note
Pin reset L time at power-on	tRESW1	10	—	ms	*1
Pin reset L time at power-off	tRESW2	0	—	µs	*2
PLL1 lock-in time	tPLL1L0	—	1	ms	*3

Note 1. tRESW1 is the reset time required for the supply of internal clock signals to become stable after all power voltages are turned on.

Note 2. tRESW2 is the time from assertion of the reset signal until all of the power voltages have dropped below the lower-limit voltages.

Note 3. tPLL1L0 is the time required for PLL1 to lock in after MOSC (main oscillator) oscillation has become stable.

CAUTIONS

- The states of I/O pins are not reset during the reset noise cancellation interval (max. 1.2 µs) following assertion of the reset signal while power is being turned off.
- If power is disconnected during programming or erasure of flash memory, data in the area of the flash memory that was being programmed or erased are not guaranteed.

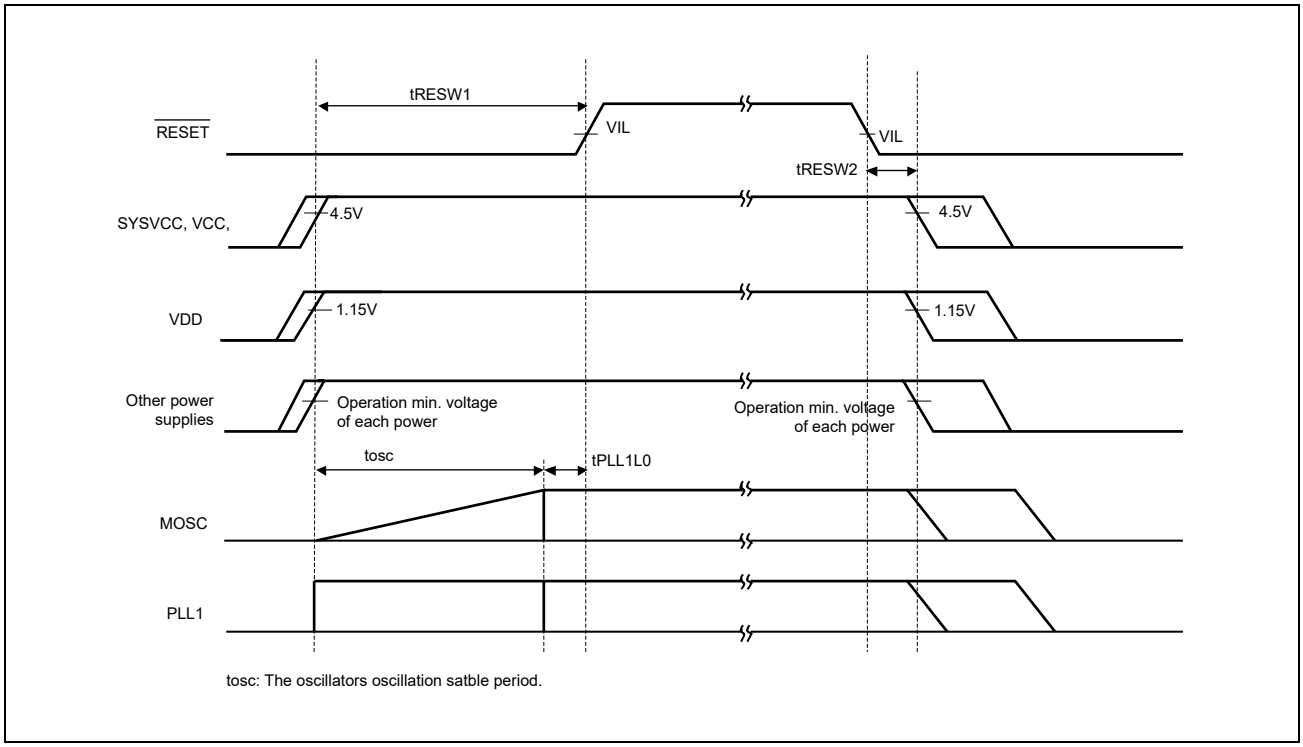


Figure 39.2 Power On/Off Timings

39.3.2 Clock Timing

39.3.2.1 Spread Spectrum Clock Generator

Table 39.14 SSCG Timing

Item	Symbol	Min.	Typ.	Max.	Unit
Modulation frequency*1	f_{mod}	20	—	100	kHz
Frequency dithering range*1	f_{dit}	4.1	—	—	%
Frequency stabilization time (OFF → ON)		—	—	1.6	ms

Note 1. The modulation method is applied only to down spread.

39.3.2.2 Oscillation Frequency Accuracy of the On-Chip Oscillator

Table 39.15 Oscillation Frequency Accuracy of the On-Chip Oscillator

Item	Symbol	Min.	Typ.	Max.	Unit
CLK_LIOSC oscillation frequency	fLIOSC	160	240	360	kHz

39.3.3 Output Slew Rate

VCC power supply pins

Table 39.16 Selection of Driving Ability = High

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Output rising time/falling time slew rate	tR, tF	CL = 25 pF	—	4	6	ns
		CL = 50 pF	—	6	12	ns
		CL = 75 pF	—	8	16	ns
		CL = 100 pF	—	10	20	ns

Table 39.17 Selection of Driving Ability = Mid

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Output rising time/falling time slew rate	tR, tF	CL = 25 pF	—	8	15	ns
		CL = 50 pF	—	15	30	ns
		CL = 75 pF	—	23	45	ns
		CL = 100 pF	—	30	60	ns

Table 39.18 Selection of Driving Ability = Low

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Output rising time/falling time slew rate	tR, tF	CL = 25 pF	—	25	50	ns
		CL = 50 pF	—	50	100	ns
		CL = 75 pF	—	70	120	ns
		CL = 100 pF	—	85	150	ns

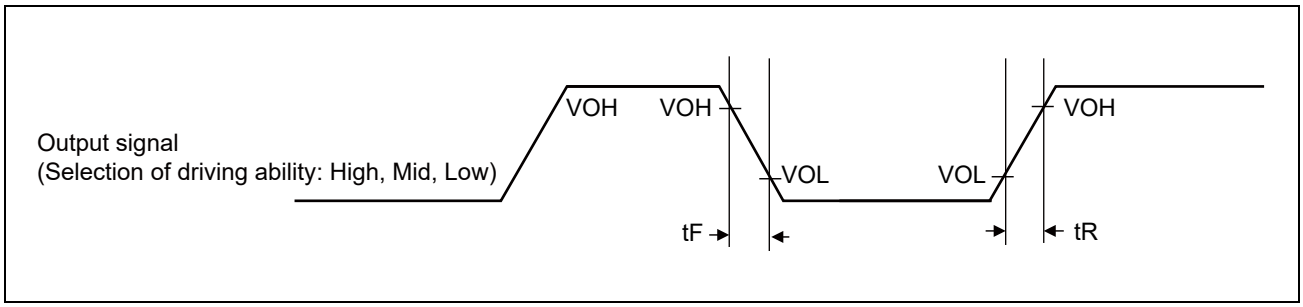


Figure 39.3 Output Signal Timing

39.3.4 Control Signal Timing

Table 39.19 Control Signals

Item	Symbol	Min.	Typ.	Max.	Unit
Reset pulse width*1	tRESW3	1.5	—	—	µs
Reset noise cancel width	tRESNCW	0.2	0.4	1.2	µs
IRQ pulse width*2	tIRQ	50	—	—	ns
Operating mode setup time	tMDS	1	—	—	ms
Operating mode hold time	tMDH	1	—	—	ms

Note 1. The reset pulse width must be equal to or more than the minimum tRESW3 value.
Any reset whose pulse width is less than the minimum value of the reset noise cancellation width will not be accepted.

Note 2. In case noise removal is disabled by DNF.

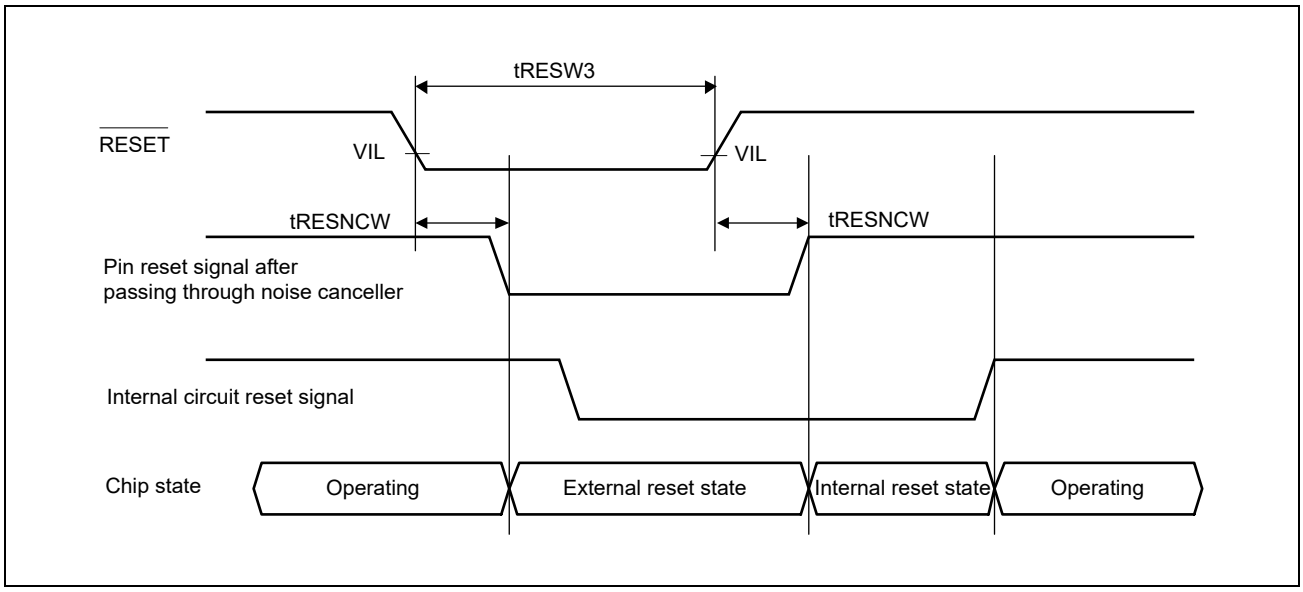


Figure 39.4 Reset Timing

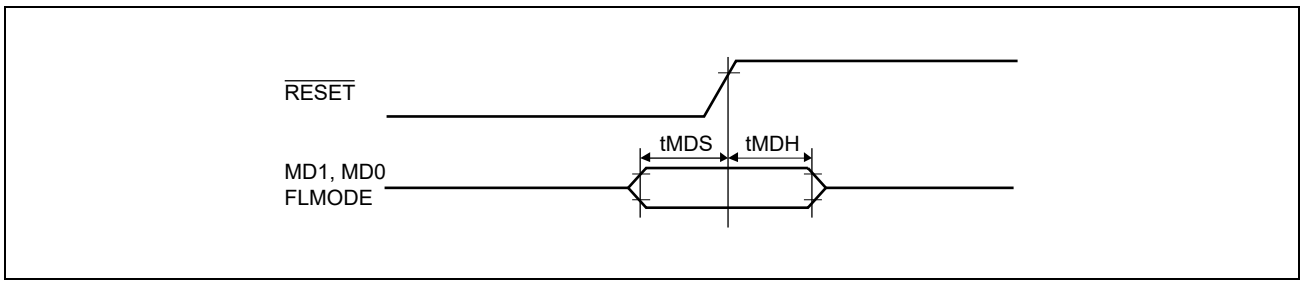


Figure 39.5 Control Signal Timing

39.3.5 CSIH Timing

39.3.5.1 Master Mode

Table 39.20 CSIH Timing in Master Mode

Conditions: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Max.	Unit
CSIHnSC cycle	tKCYM		100	—	ns
CSIHnSC output high-level width	tKWHM		(tKCYM/2) – 20	—	ns
CSIHnSC output low-level width	tKWLM		(tKCYM/2) – 20	—	ns
CSIHnSI input setup time	tSSIM		18	—	ns
CSIHnSI input hold time	tHSIM		10	—	ns
CSIHnSO output delay time	tDSOM		—	10	ns
CSIHnSO output hold time (vs. CSIHnSC)	tHSOM		tKWHM – 10	—	ns
CSIHnRYI setup time	tSRYI	HSE = 1	(2 × tPAck) + 30	—	ns
CSIHnCSSx inactive level width	tWSCSB	*1	(CSidle + 0.5) × tKCYM – 20	—	ns
		Other than above	CSidle × tKCYM – 20	—	ns
CSIHnCSSx setup time	tSSCSB0	DAP = 0	CSsetup × tKCYM – 10	—	ns
	tSSCSB1	DAP = 1	(CSsetup + 0.5) × tKCYM – 10	—	ns
CSIHnCSSx hold time	tHSCSB0	SIT = 0	CShold × tKCYM – 10	—	ns
	tHSCSB1	SIT = 1	(CShold + 0.5) × tKCYM – 10	—	ns

Note 1. When the serial clock level is changed during communication and when the idle time is set to 0.5 transmission clock periods.

Note: tPAck is the operating clock cycle of CSIH (80 MHz SSCG)

n = 0 to 2, x = 0 to 3

CSsetup : CSIHnCFGx.CSIHnSPx3-0 set value

CShold : CSIHnCFGx.CSIHnHDx3-0 set value

CSidle : CSIHnCFGx.CSIHnIDx2-0 set value

DAP : CSIHnCFGx.CSIHnDAPx bit

SIT : CSIHnCTL1.CSIHnSIT bit

HSE : CSIHnCTL1.CSIHnHSE bit

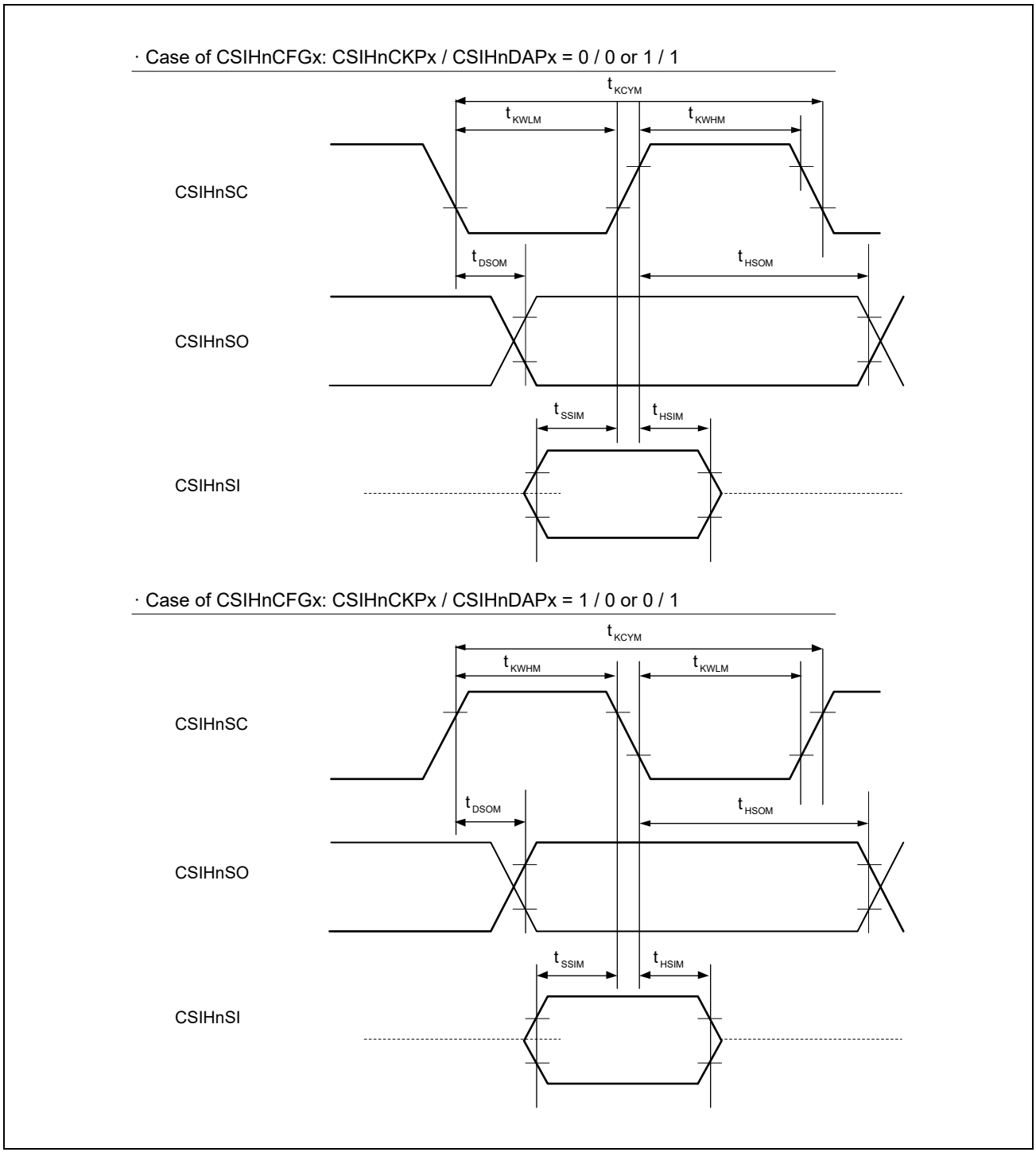


Figure 39.6 CSIH Timing (Master Mode) (1/4)

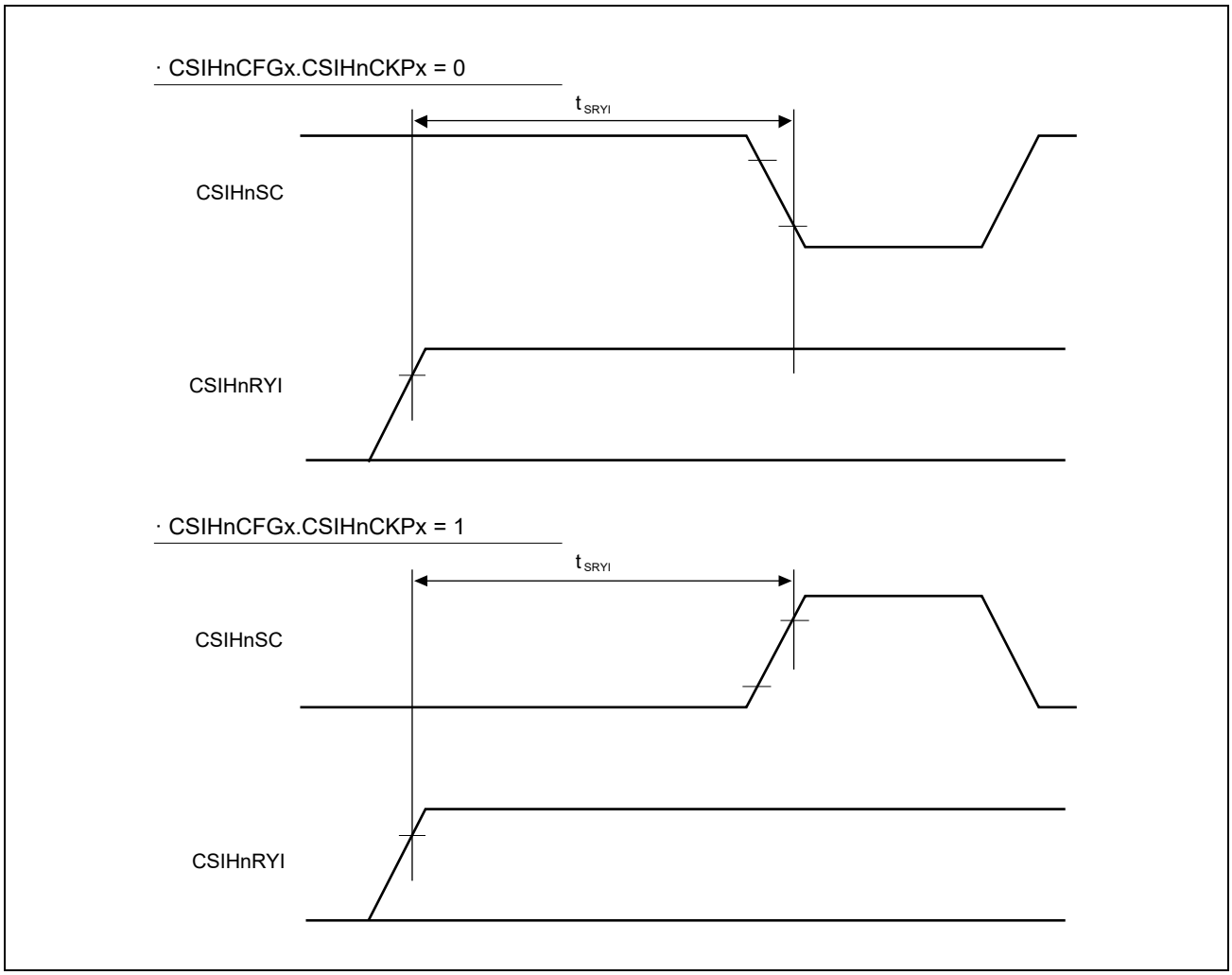


Figure 39.6 CSIH Timing (Master Mode) (2/4)

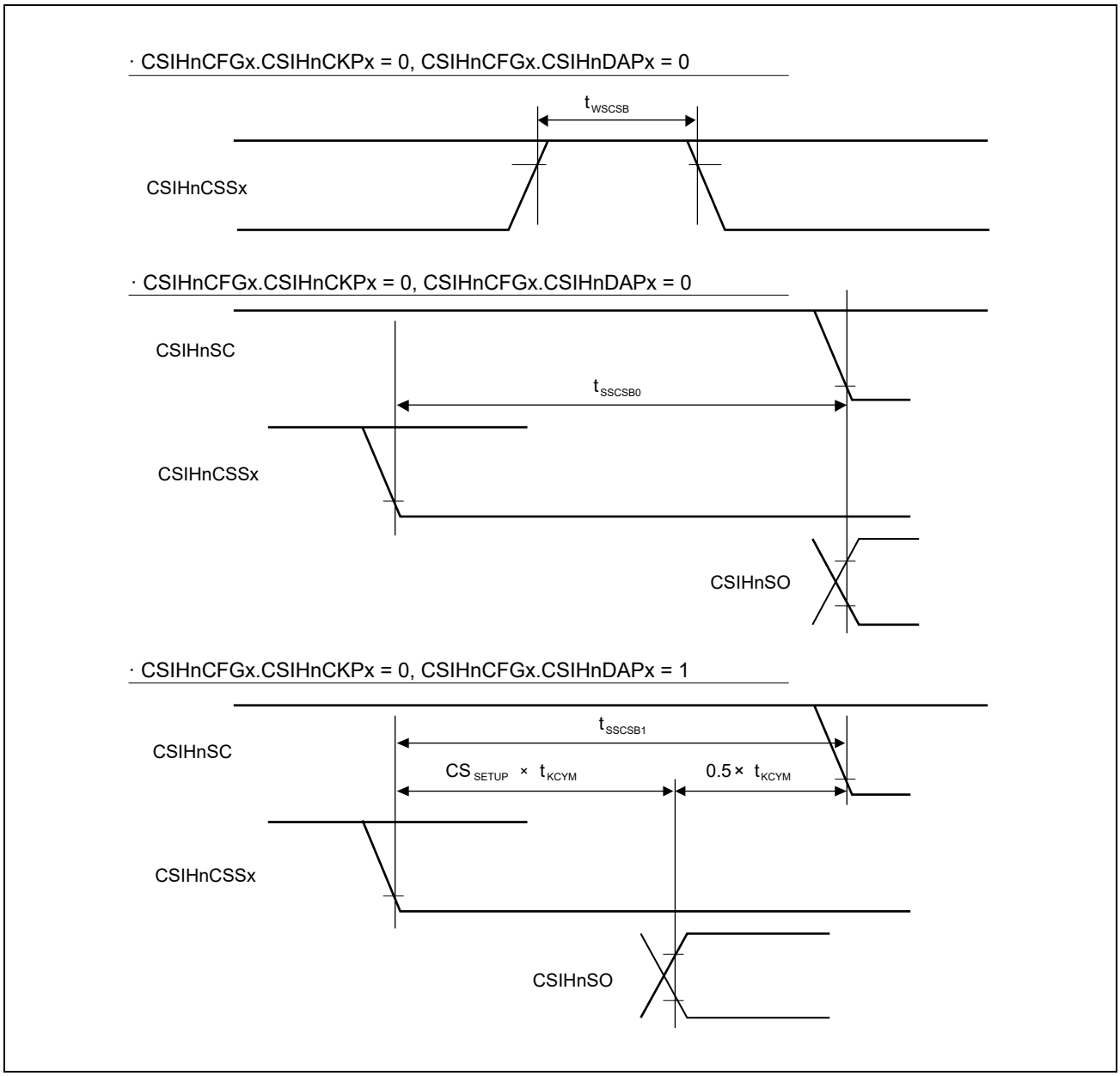


Figure 39.6 CSIH Timing (Master Mode) (3/4)

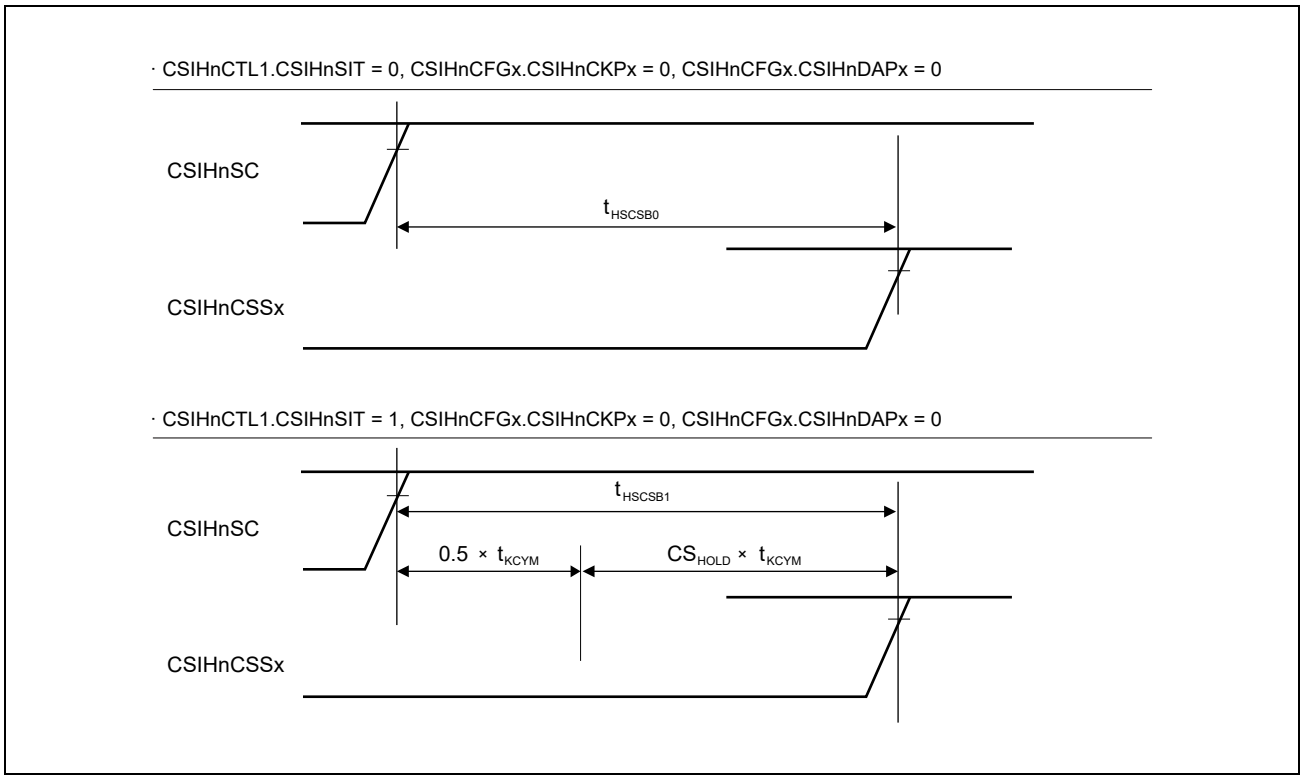


Figure 39.6 CSIH Timing (Master Mode) (4/4)

39.3.5.2 Slave Mode

Table 39.21 CSIH Timing in Slave Mode

Conditions: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Max.	Unit
CSIHnSC cycle	tKCYS		200	—	ns
CSIHnSC input high-level width	tKWHS		(tKCYS/2) – 30	—	ns
CSIHnSC input low-level width	tKWLS		(tKCYS/2) – 30	—	ns
CSIHnSI input setup time	tSSIS		15	—	ns
CSIHnSI input hold time	tHSIS		tPAck + 15	—	ns
CSIHnSO output delay time	tDSOS		—	30	ns
CSIHnSO output hold time (vs. CSIHnSC)	tHSOS		tKWHS	—	ns
CSIHnRYO output delay time	tSRYO		—	30	ns
$\overline{\text{CSIHnSSI}}$ setup time	tSSSIS		0.5 × tKCYS	—	ns
$\overline{\text{CSIHnSSI}}$ hold time	tHSSIS		tPAck + 30	—	ns
Slave output release time	tREL		—	100	ns

Note: tPAck is the operating clock cycle of CSIH (80 MHz SSCG).

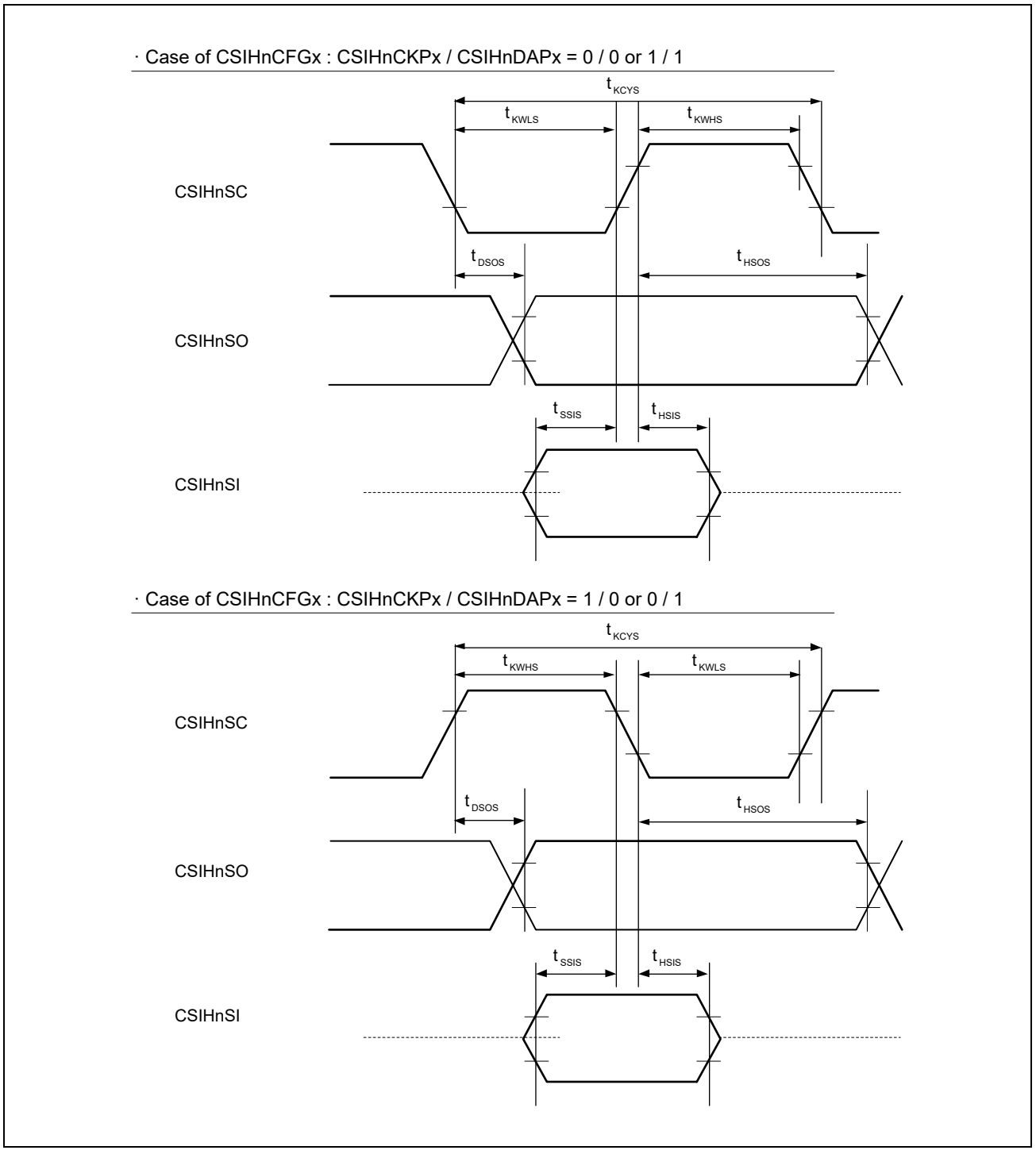


Figure 39.7 CSIH Timing (Slave Mode) (1/3)

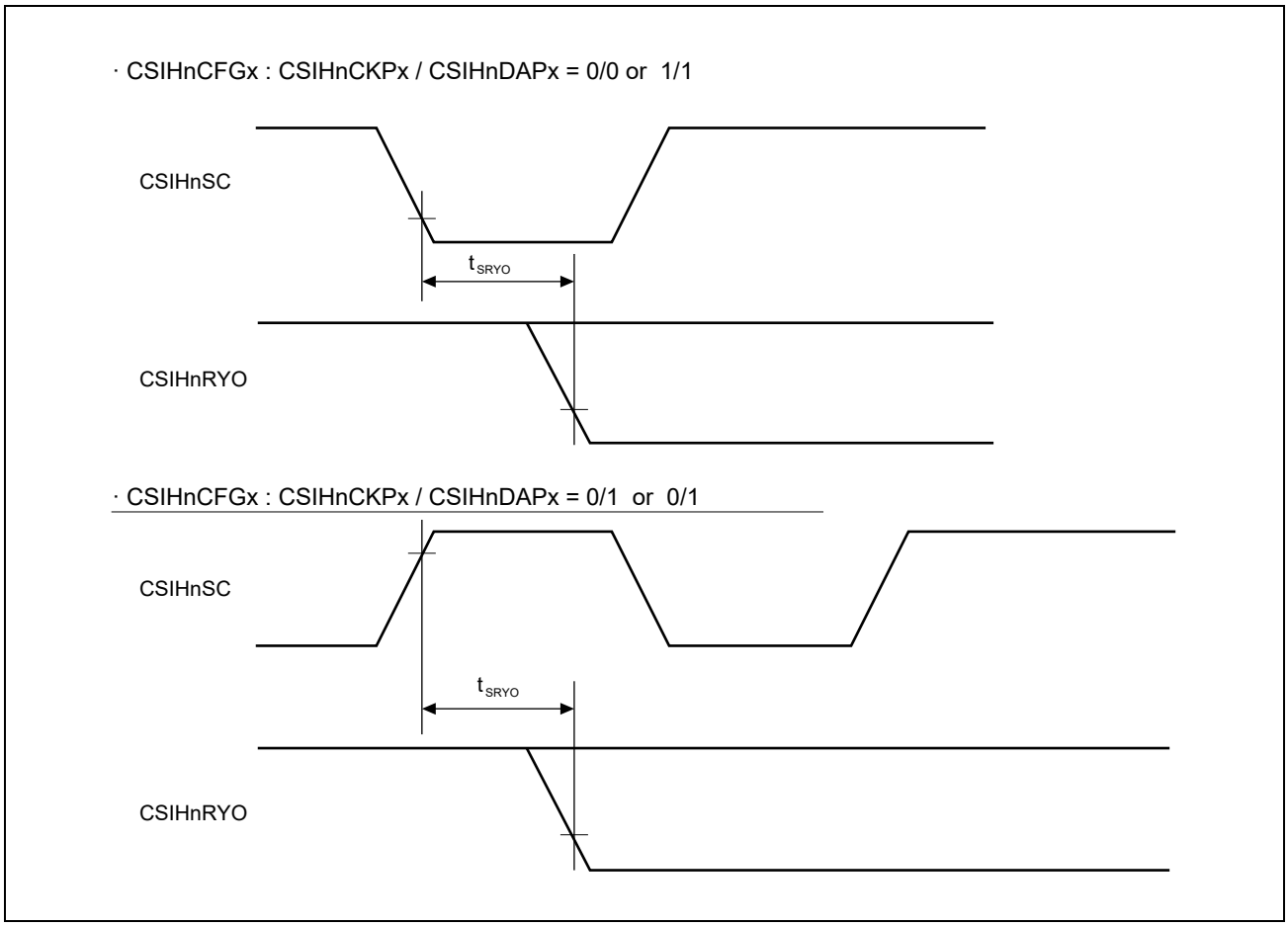


Figure 39.7 CSIH Timing (Slave Mode) (2/3)

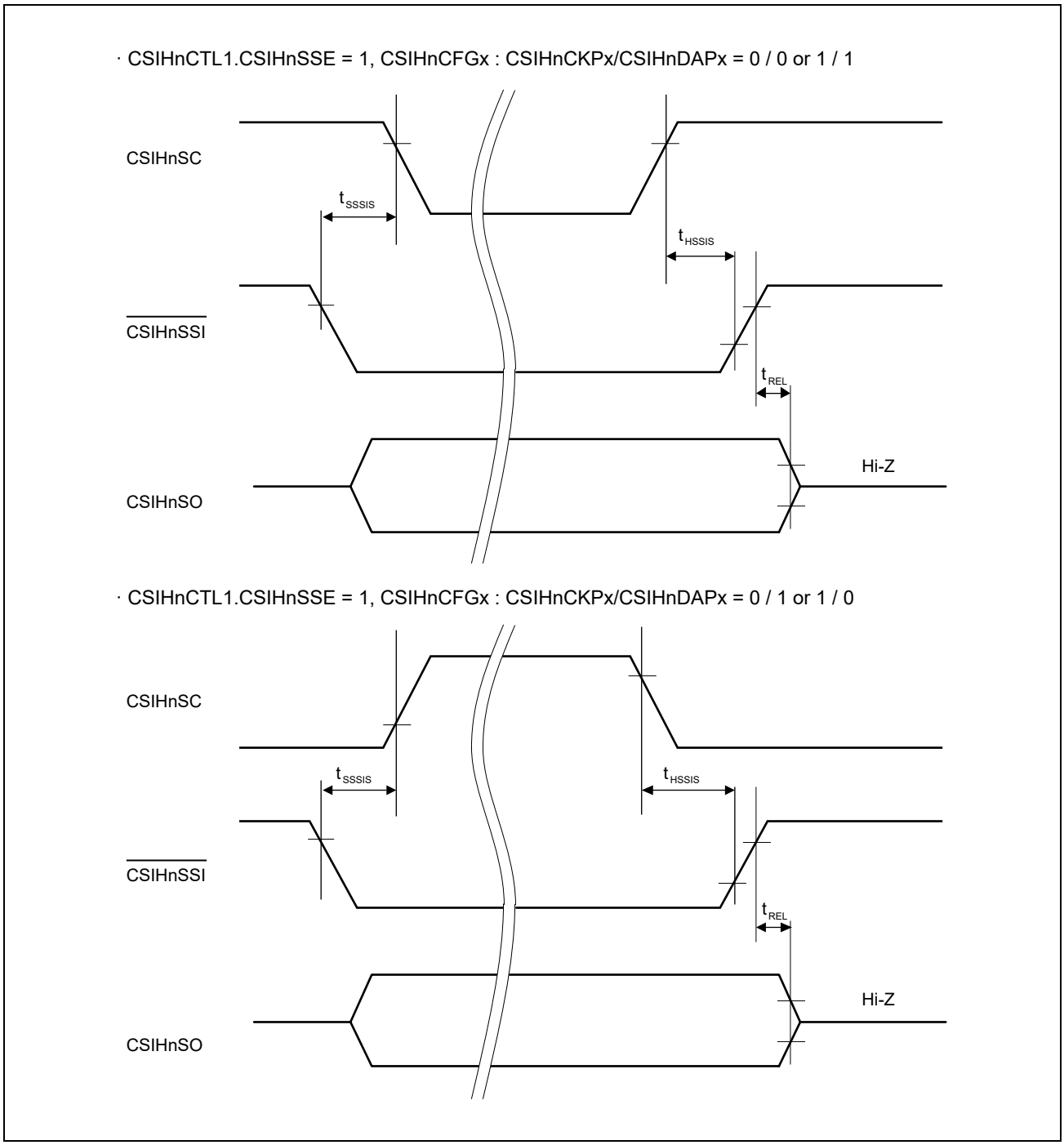


Figure 39.7 CSIH Timing (Slave Mode) (3/3)

39.3.6 SCI/FLSCI Timing

Table 39.22 SCI3 Timing (Master Mode)

Conditions: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Max.	Unit
Output clock cycle	tScyc	Asynchronous	8 × tPck	—	ns
		Clock synchronous	8 × tPck	—	ns
Output clock pulse width	tSCKW		0.4 × tScyc	0.6 × tScyc	ns
Transmit data delay time	tTXD	Clock synchronous	—	40	ns
Receive data setup time	tRXS	Clock synchronous	2 × tPck	—	ns
Receive data hold time	tRXH	Clock synchronous	2 × tPck	—	ns

Note: tPck is the operating clock cycle of SCI (40 MHz clean clock).

CAUTION

FLSCI does not support master mode.

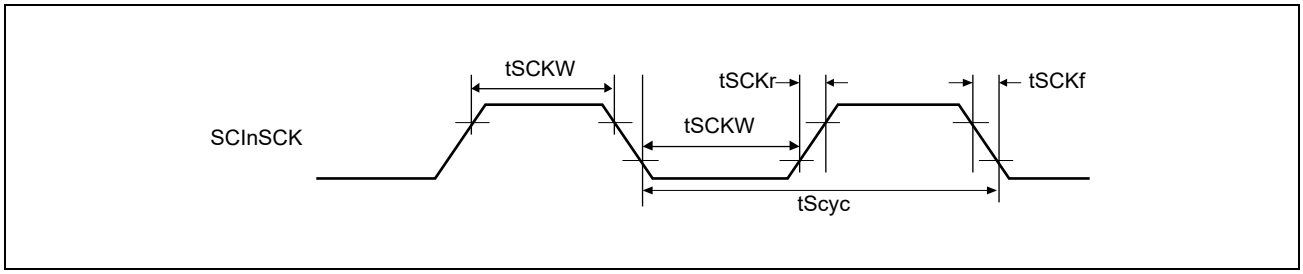


Figure 39.8 SCI Clock Input/Output Timing

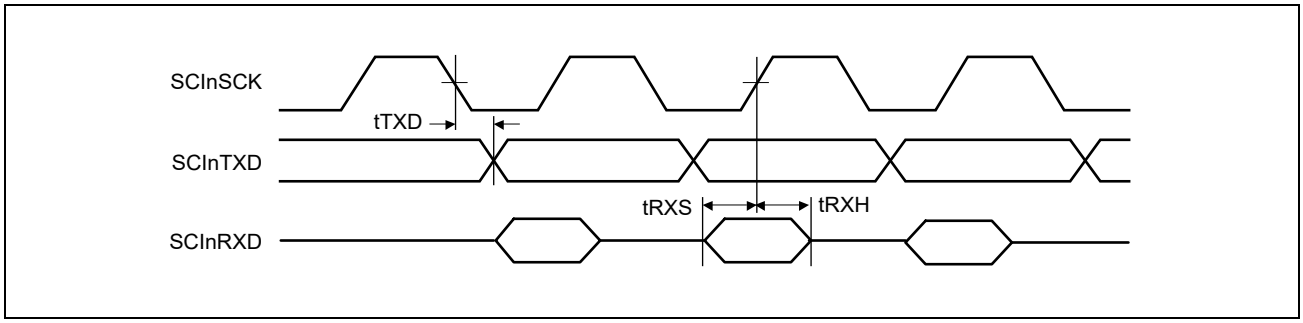


Figure 39.9 SCI Input/Output Timing, Clock Synchronous Mode

Table 39.23 SCI3 Timing (Slave Mode)

Conditions: CL = 50 pF, selection of driving ability = High

Item	Symbol	Min.	Typ.	Max.	Unit
Input clock cycle	tScyc	12 × tPck	—	—	ns
Input clock pulse width	tSCKW	0.4 × tScyc	—	0.6 × tScyc	ns
Input clock rising time	tSCKr	—	—	20	ns
Input clock falling time	tSCKf	—	—	20	ns
Transmit data delay time*1	tTXD	2 × tPck	—	50 + 3 × tPck	ns
Receive data setup time	tRXS	2 × tPck	—	—	ns
Receive data hold time	tRXH	2 × tPck	—	—	ns

Note 1. Applies to data other than “non-continuous transfer mode Data0 (1st bit).” The transmission of “non-continuous mode Data0 (1st bit)” is initiated at the same time TDRE is set to 0.

Note: tPck is the operating clock cycle of SCI (40 MHz clean clock).
Asynchronous clock input mode is not supported.

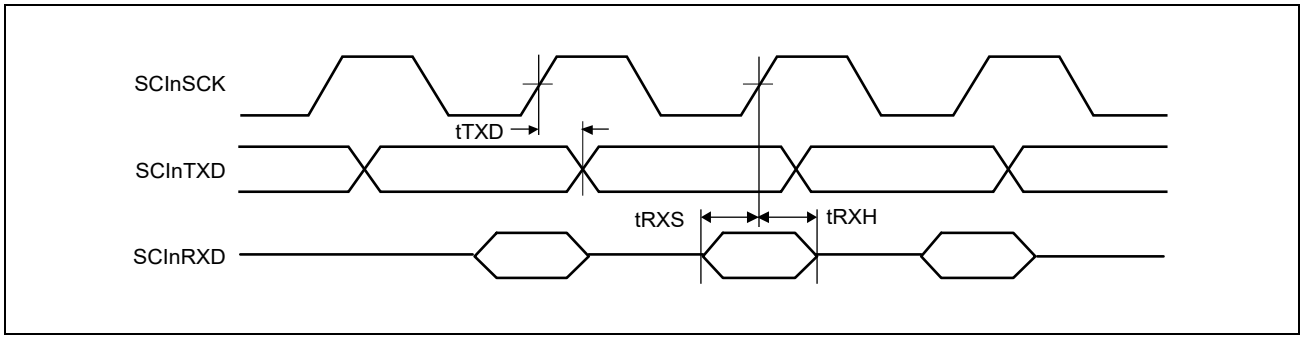


Figure 39.10 SCI Input/Output Timing, Clock Synchronous Mode (in Slavel Mode)

39.3.7 RS-CANFD Timing

Table 39.24 RS-CANFD Timing

Conditions: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition.	Min.	Typ.	Max.	Unit
Transfer rate			—	—	5	Mbps
Internal delay time	t_{NODE}		—	—	50	ns

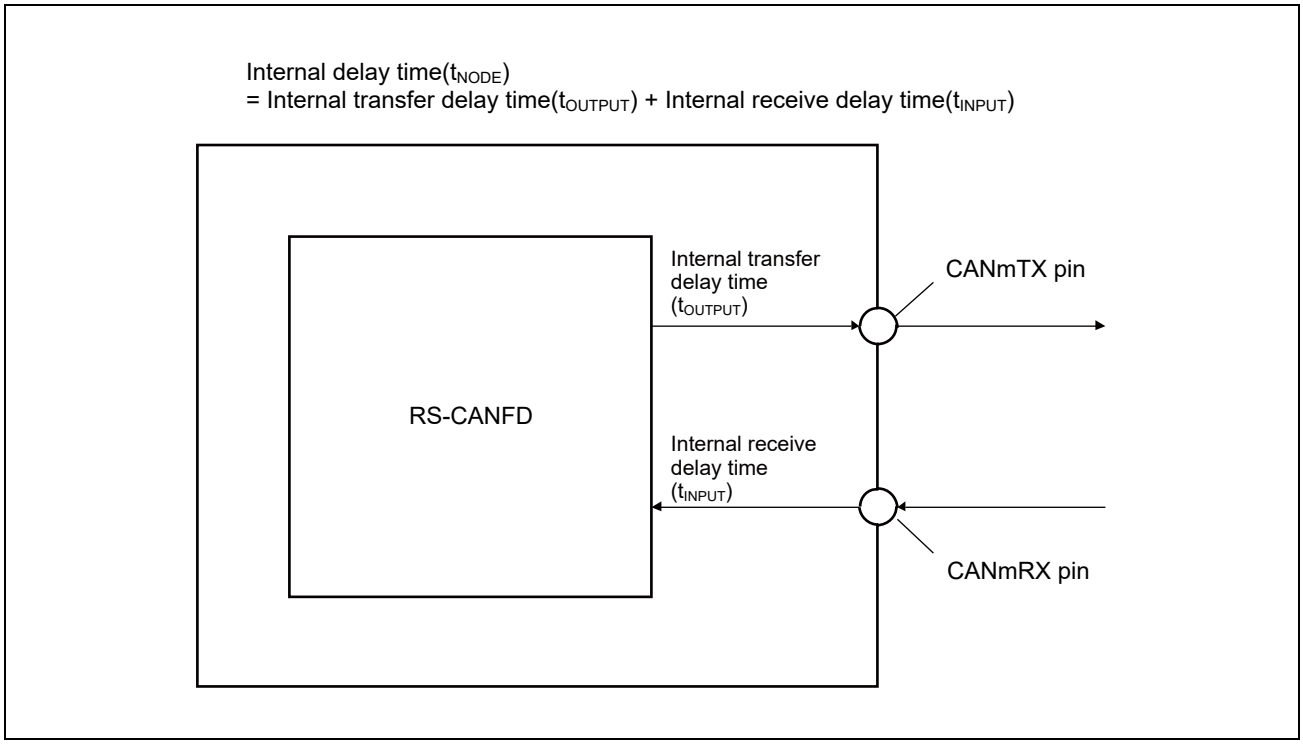


Figure 39.11 RS-CANFD Timing

Definition of internal delay time of RS-CANFD

Internal delay time (tNODE) = tOUTPUT + tINPUT

39.3.8 RLIN3 Timing

Table 39.25 RLIN3 Timing

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Transfer rate		LIN function	—	—	20	kbps
		UART function	—	—	1.5	Mbps

39.3.9 Motor Control Signals Timing

Table 39.26 Motor Control Signals Timing

Item	Symbol	Condition	Min.	Max.	Unit
Input high-level width	tTIH	ENCAnE0-1, ENCAnEC*1, TAPAnESO	1.5 × tPck	—	ns
Input low-level width	tTIL	ENCAnE0-1, ENCAnEC*1, TAPAnESO	1.5 × tPck	—	ns

Note: In case noise removal is disabled by DNF

Note 1. When used as hall sensor inputs (TSG3nPTSI0-I2).

Note: tPck is the operating clock cycle of TSG3 (80 MHz clean clock).

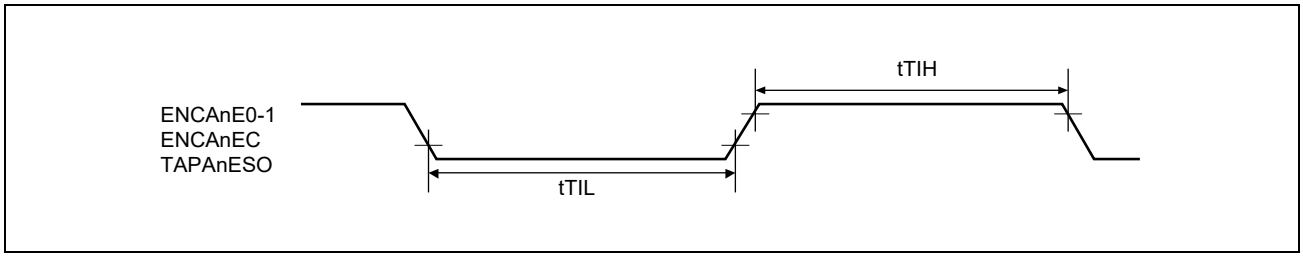


Figure 39.12 Motor Control Signals Timing

39.3.10 Timer Timing

Table 39.27 Timer Timing

Item	Symbol	Condition	Min.	Max.	Unit
Input high-level width	tTIH	TAUDnI0-15, TAUJnI0-3, ENCAnI0-1, ENCAnE0-1, ENCAnEC	1.5 × tPck	—	ns
Input low-level width	tTIL	TAUDnI0-15, TAUJnI0-3, ENCAnI0-1, ENCAnE0-1, ENCAnEC	1.5 × tPck	—	ns

Note: In case noise removal is disabled by DNF.

Note: tPck is the operating clock cycle of the timer (80 MHz clean clock).

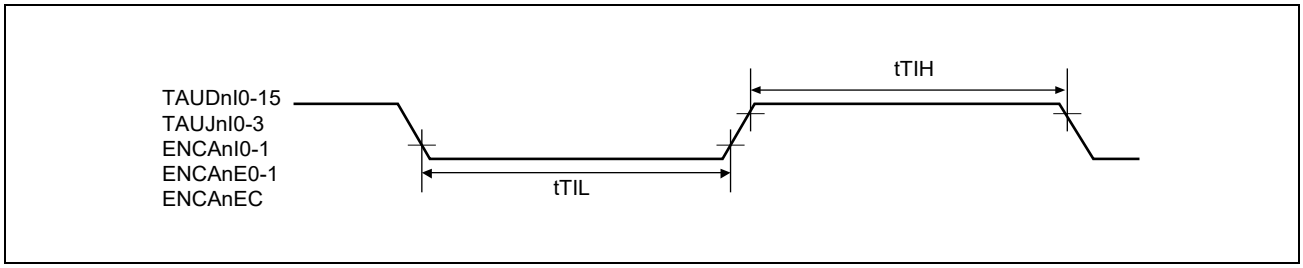


Figure 39.13 Timer Timing

39.3.11 JTAG/NEXUS Timing

Table 39.28 JTAG/NEXUS Timing

Conditions: CL = 30 pF

Item	Symbol	Condition	Min.	Max.	Unit
DCUTCK cycle time	tTCKW		50	—	ns
DCUTCK high-level width	tTCKWH		21	—	ns
DCUTCK low-level width	tTCKWL		21	—	ns
DCUTMS, DCUTDI setup time (vs. DCUTCK ↑)	tTISU		12	—	ns
DCUTMS, DCUTDI hold time (vs. DCUTCK ↑)	tTIH		12	—	ns
DCUTDO output delay time (vs. DCUTCK ↓)	tTDOD		—	tTCKW-20	ns
$\overline{\text{DCURDY}}$ output delay time (vs. DCUTCK ↓)	tRDYD		—	tTCKW-20	ns
$\overline{\text{DCUTRST}}$ low-level width	tTRSTWL		1200	—	ns
$\overline{\text{DCUTRST}}$ /DCUTCK/DCUTMS/DCUTDI input rising time	tTIR		—	12	ns
$\overline{\text{DCUTRST}}$ /DCUTCK/DCUTMS/DCUTDI input falling time	tTIF		—	12	ns

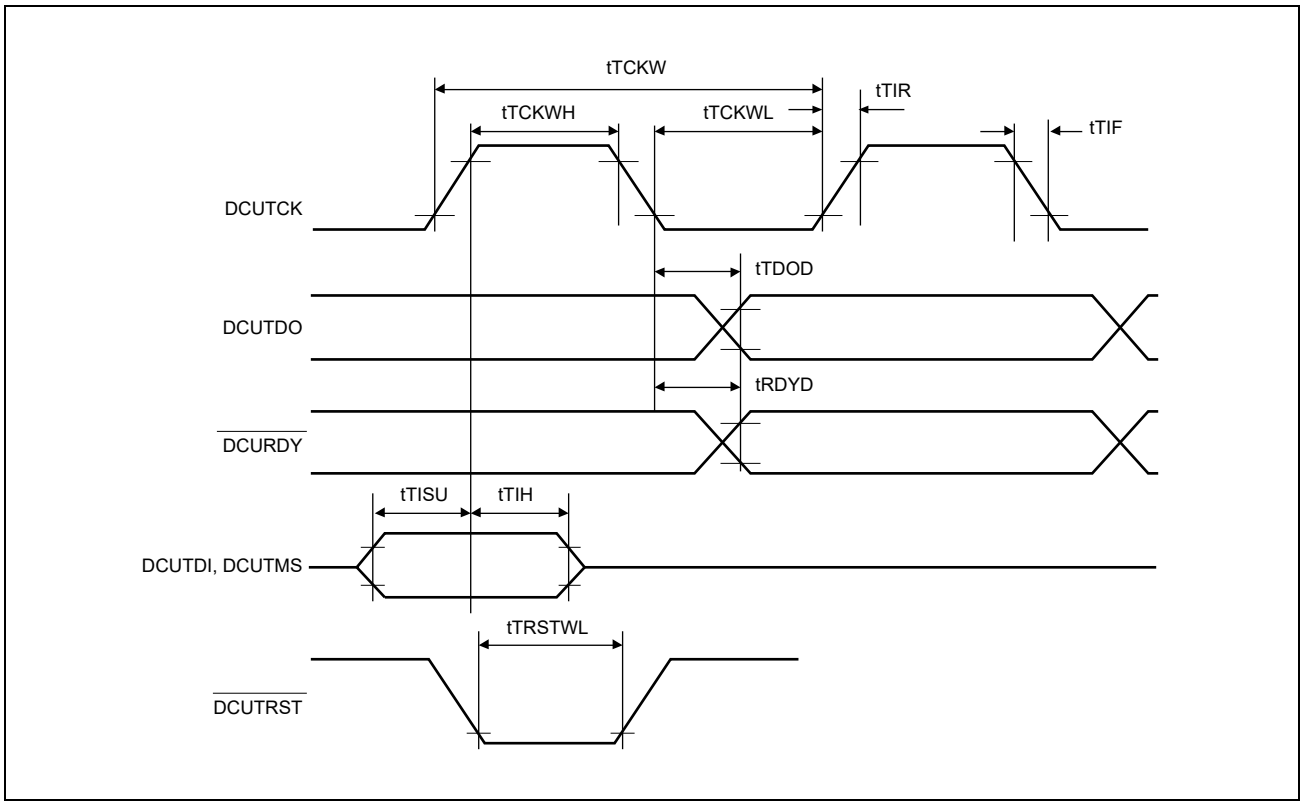


Figure 39.14 JTAG/NEXUS Timing

39.3.12 LPD (4-pin) Timing

Table 39.29 LPD (4-pin) Timing

Conditions: Tj = –40ºC to 150ºC, CL = 30 pF

Item	Symbol	Condition	Min.	Max.	Unit
LPDCLK cycle	tLPDCKW		25	—	ns
LPDCLK high-level width	tLPDCKWH		4.5	—	ns
LPDCLK low-level width	tLPDCKWL		4.5	—	ns
LPDCLK input rising time	tLPDCKR		—	8	ns
LPDCLK input falling time	tLPDCKF		—	8	ns
LPDI setup time (to LPDCLK ↑)	tLPDSU		2	—	ns
LPDI hold time (from LPDCLK ↑)	tLPDH		2	—	ns
LPDCLKOUT cycle time	tLPDCKOW		25	—	ns
LPDCLKOUT high-level width	tLPDCKOWH		4.5	—	ns
LPDCLKOUT low-level width	tLPDCKOWL		4.5	—	ns
LPDCLKOUT rising time	tLPDCKOR		—	8	ns
LPDCLKOUT falling time	tLPDCKOF		—	8	ns
LPDO output delay (from LPDCLKOUT ↑)	tLPDOD		0	12	ns

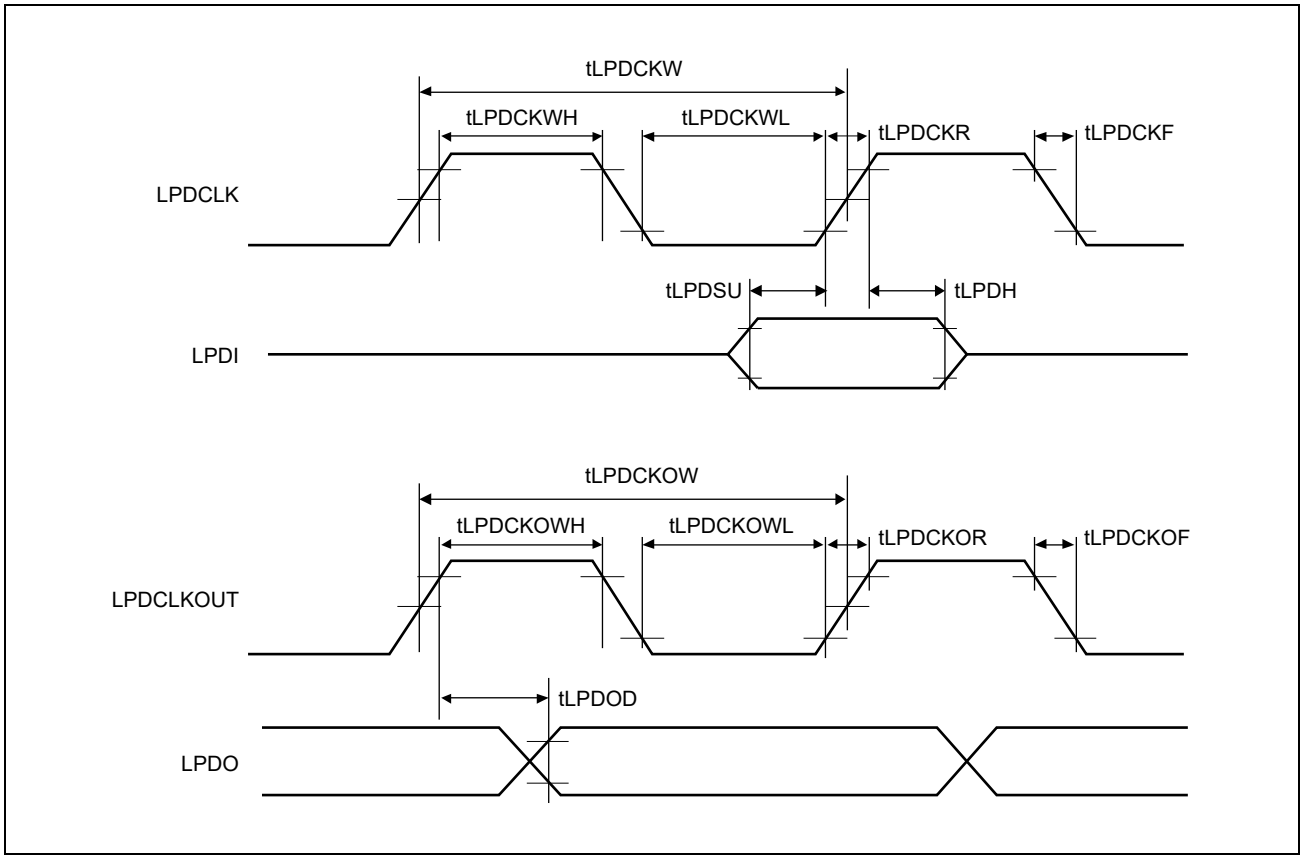


Figure 39.15 LDU 4-Wire Timing

39.3.13 AUD RAM Monitor

Table 39.30 AUD RAM Monitor Timing

Conditions: Tj = – 40ºC to 150ºC, CL = 30 pF

Item	Symbol	Min.	Max.	Unit
AUDCK cycle time (monitor mode)	tAUCKMcyc	50	—	ns
AUDCK high-level width (monitor mode)	tAUCKMH	0.4 × tAUCKMcyc	—	ns
AUDCK low-level width (monitor mode)	tAUCKML	0.4 × tAUCKMcyc	—	ns
$\overline{\text{AUDRST}}$ setup time (monitor mode, vs. AUDCK↑)	tAURSTMS	30	—	ns
$\overline{\text{AUDRST}}$ input pulse width (monitor mode)	tAURSTMW	5 × tAUCKMcyc	—	ns
Monitor data output delay time (to AUDCK ↑)	tAUDTMD	—	35	ns
Monitor data input setup time (to AUDCK ↑)	tAUDTMS	15	—	ns
Monitor data input hold time (from AUDCK ↑)	tAUDTMH	5	—	ns
$\overline{\text{AUDSYNC}}$ input setup time (vs. AUDCK ↑)	tAUDSYS	15	—	ns
$\overline{\text{AUDSYNC}}$ input hold time (vs. AUDCK ↑)	tAUDSYH	5	—	ns

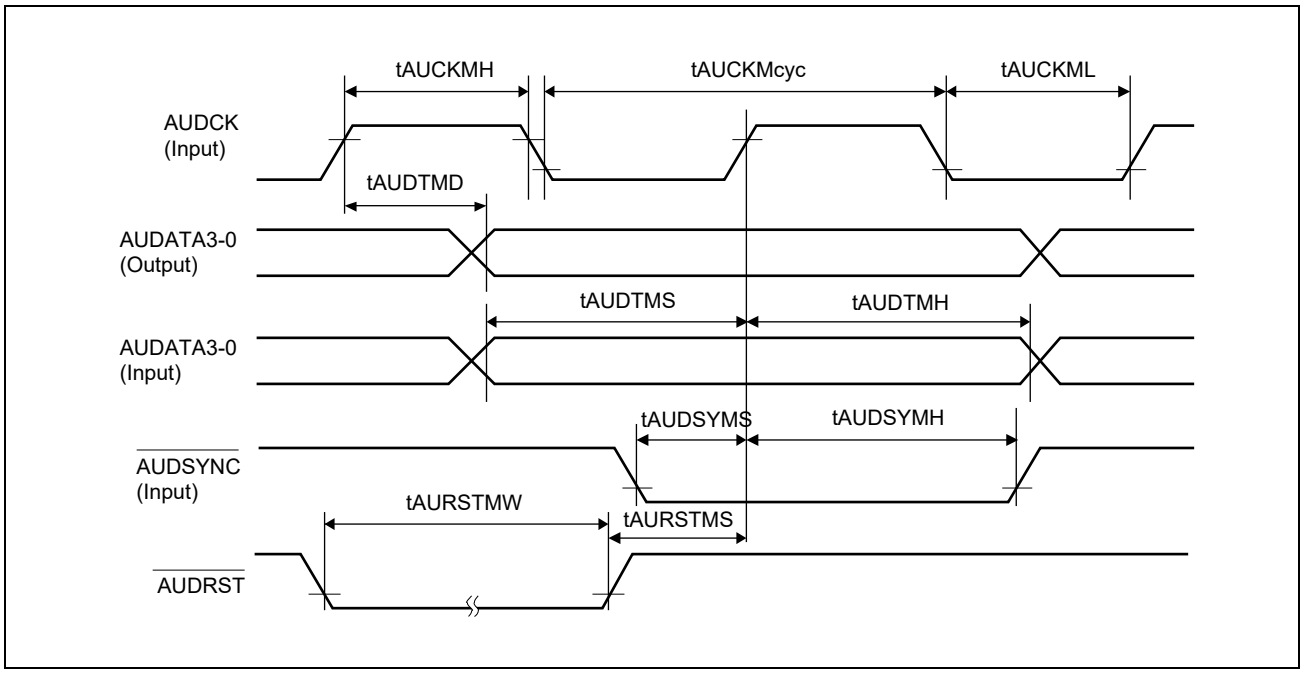


Figure 39.16 AUD RAM Monitor Timing

39.4 A/D Converter Characteristics

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Table 39.31 A/D Converter Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Digital resolution	—	—	—	12	—	bit
A/D conversion time*1	—	—	—	1	—	µs
Integral nonlinear error	—	Using T&H amplifier	—	—	±4	LSB
Offset error	—	Using T&H amplifier	—	—	±7.5	LSB
Full-scale error	—	Using T&H amplifier	—	—	±7.5	LSB
Quantization error	—	—	—	—	±0.5	LSB
Absolute error	—	—	—	—	±8.0	LSB
Self-diagnosis absolute error	—	In A/D converter self-diagnosis	—	—	±8.0	LSB
	—	In pin level self-diagnosis	—	—	±80	LSB
Analog input capacitance	—	A/D conversion standby	—	—	10	pF
	—	In sampling	—	—	20	pF
Allowable analog signal source impedance	—	—	—	—	3	kΩ
Channel T&H hold time*2	—	—	—	—	10	µs
T&H sampling time	—	—	—	—	0.45	µs
Pull-up resistance for A/D disconnection detection	—	AnVCC = 4.5V to 5.5V, ADCCnIpq = AnVSS	10	20	40	kΩ
Pull-down resistance for A/D disconnection detection	—	AnVCC = 4.5V to 5.5V, ADCCnIpq = AnVCC	10	20	40	kΩ
Input voltage range	—	Not used T&H amplifier	0	—	A0VREFH A1VREFH A2VREFH	V
	—	Using T&H amplifier	0.2	—	A0VREFH-0.2 A1VREFH-0.2 A2VREFH-0.2	V

Note 1. Conversion time for a channel, and not include T&H time.

Note 2. When the T&H circuit is in use, A/D conversion must be performed within the maximum time.

- Errors in the External Circuit of the A/D Converter

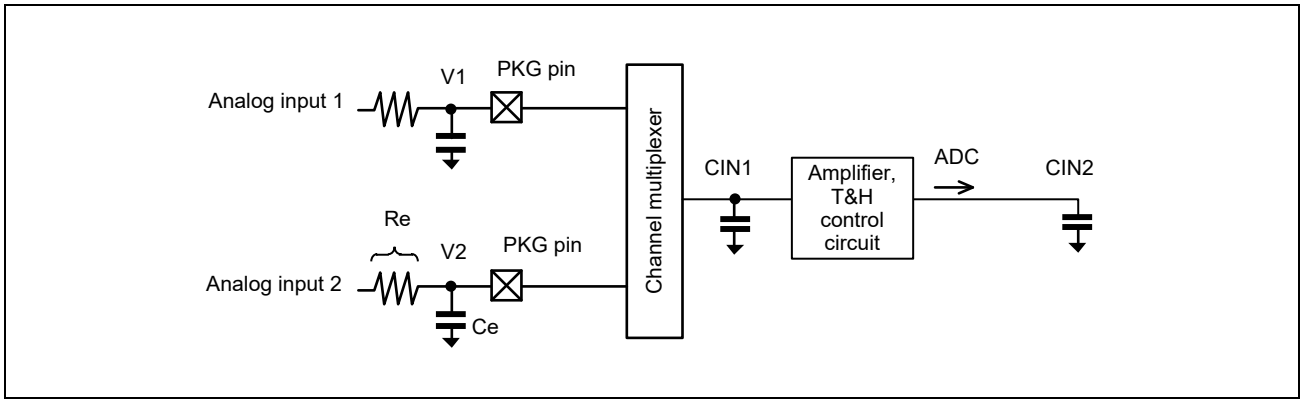
  A formula for errors in sampled values due to the external circuit of the A/D converter is given below.
  These errors will depend on the input circuit and conversion cycle. The formula given below for the errors is simplified for the calculation of sampling errors based on internal stray capacitance, amplifier offset, resistance of the signal source, and conversion cycle. This formula can also be used to calculate the effects of the signal source resistance and conversion cycle on these errors.

  The formula gives the error of analog input 2 as shown in the figure below when A/D conversion is performed in the order analog input 1 then 2.

$$Sampling\ error\ (LSB) = \left[\left(\frac{|V2 - V1| \times CIN1}{Ce + CIN1}| + \frac{|Vvfaerr| \times CIN2}{Ce + CIN2}\right) \times \frac{1}{1 - e^{-T1/(Re \times Ce)}} + \left(\frac{1}{T1} \times C1 \times V3 \times Re\right)\right] \times \frac{4096}{Vavrefh}$$

Table 39.32 Parameters of C1M-A

Item	Symbol	Reference	Unit
Common capacitance of the final stage of channel multiplexer	CIN1	1.6	pF
Common capacitance of the final stage of the amplifier and T&H control circuit	CIN2	10	pF
External capacitor on analog input pin	Ce	Depends on user board	uF
Signal source impedance	Re		kΩ
Conversion cycle of conversion pins	T1		ms
AnVREFH voltage (n = 0 to 2)	Vavrefh		V
Potential difference between V1 and V2	V2-V1	5	V
Offset voltage of amplifier and T&H control circuit	Vvfaerr	50	mV
Parasitic capacitance in channel multiplexer	C1	2	pF
|AnVCC voltage /2.5 – measured pin voltage| (n = 0 to 2)	V3	Depends on user board	V



- Values for conversion error calculated by using this formula do not include error (absolute error, etc.) specified in the A/D converter characteristics.
- This formula is a desktop formula and theoretical. When the signal source has an extremely high resistance or when the conversion cycle is too short, calculated and measured values may differ.
  Actual error depends on the capacitor, resistor, capacitance and resistance of board wiring, so please evaluate and verify the error on the user board is no greater than the value produced by this formula (Re < 1.5 MΩ and T1 ≥ 10 µs, or 1.5 MΩ ≤ Re ≤ 2 MΩ and T1 ≥ 512 µs).

39.5 R/D Converter Characteristics

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

39.5.1 RDC Conversion Performance

Table 39.33 RDC Conversion Performance (1/2)

Item	Condition		Min.	Typ.	Max.	Unit
Resolution*1			—	—	16	bit
Conversion accuracy*2	Absolute error in electrical angle signal while operation is stopped (12-bit resolution)	Angle conversion mode 0	—	—	±4	LSB
		Angle conversion mode 1	—	—	±4	
Settling time (Response for input of electric angle of 180°)	Settling range within ±8 LSB	Band 800 Hz	—	53	—	ms
		Band 1500 Hz	—	31	—	
		Band 1000 Hz	—	43	—	
		Band 500 Hz	—	85	—	
		Band 200 Hz	—	211	—	
		Automatic adjustment	—	1.9	—	
Maximum angular velocity*3 (The values in parentheses are applied to range setting as automatic adjustment)	16-bit resolution		15000 (7500)	—	—	min^{-1}
	14-bit resolution		60000 (30000)	—	—	
	13-bit resolution		120000 (60000)	—	—	
	12-bit resolution		240000 (120000)	—	—	
	11-bit resolution		480000 (240000)	—	—	
	10-bit resolution		960000 (480000)	—	—	
Maximum angular acceleration Following angular acceleration range (electrical angle)	Band 800 Hz		—	146000	—	rad/s^2
	Band 1500 Hz		—	513000	—	
	Band 1000 Hz		—	183000	—	
	Band 500 Hz		—	46000	—	
	Band 200 Hz		—	5000	—	
	Automatic adjustment		—	3000000	—	

Table 39.33 RDC Conversion Performance (2/2)

Item	Condition		Min.	Typ.	Max.	Unit
Response delay*4	Electrical angle output response delay in fixed angular velocity	Angle conversion mode 0	–0.2	—	0.20	°/10000 min^{-1}
		Angle conversion mode 1	–0.2	—	0.20	
BIST determination time*5	Obtaining the sum of squares in amplitude abnormality detection BIST (L side)		—	—	1	ms
	Obtaining the sum of squares in amplitude abnormality detection BIST (H side)		—	—	1	ms
	ADBIST		—	—	32	µs
	Angle conversion BIST (angle determination threshold is within ±8 LSB)		—	—	10	ms
	Resolver signal error detection BIST		—	—	0.5	ms
	Resolver signal cut off detection BIST		—	—	1	ms
	Conversion error BIST		—	—	10	ms
	Power short error BIST		—	—	80	µs
	Ground short error BIST		—	—	80	µs
BIST recovery time*6	All kinds of BIST		—	—	10	ms

Note 1. The resolution is changed by the setting of the maximum angular velocity select bit in the RDC3An control gain select register 1. The angle can be read with maximum 16-bit width by register access.

Note 2. It is the ability, when the waveform of analog input to RDC is ideal sign wave. RDC conversion result will defer from resolver machine angle with distortion or slippage of analog input signal or power supply voltage.

Note 3. Following angular velocity range (resolver electrical angle). It is changed by the setting of the maximum angular velocity select bit in the RDC3An control gain select register 1.

Note 4. PHI angle output from RDC is added accuracy error through analog circuit to this value. The read of PHI angle output register value with bus access needs access time. However, it does not need access time using PHI compare signal.

Note 5. It is the time to stabilize BIST determination.

Note 6. Recovery time from BIST operation to normal operation.
When the excitation frequency is under 9 kHz, BIST recovery time is maximum 15 ms.
When the conversion error determination time is set to over 10 ms, BIST recovery time is also set to over the setting value (10 ms).

39.5.2 RDC Analog Pin

Table 39.34 RDC Analog Pin Characteristics

Signal	Symbol	Item	Min.	Typ.	Max.	Unit
Signal source output*1 for resolver excitation power supply	RSO	Frequency	5	—	40	kHz
		Output voltage*2	0.38 × RVCC	0.4 × RVCC	0.42 × RVCC	VP-P
		Load impedance	10	—	—	kΩ
		Output switching*3	–40	±0	+20	%
Common voltage output for resolver excitation power supply	COM	Output voltage	0.475 × RVCC	0.5 × RVCC	0.525 × RVCC	V
		Load impedance	10	—	—	kΩ
Resolver excitation signal external input	R1E, R2E	Frequency*4, *10	5	—	40	kHz
		Input voltage range	0	—	RVCC	V
		Input voltage differential amplitude	2	—	—	VP-P
		Input impedance*11	32	40	48	kΩ
Resolver signal input	S1, S2, S3, S4	Frequency*10	5	—	40	kHz
		Input voltage range*5	—	—	—	V
		Input impedance*6	16.2	21	25.8	kΩ
		Input impedance switching*7	-40	±0	+40	%
Resolver signal monitor output	COSMNT, SINMNT	Frequency*8	5	—	40	kHz
		Output voltage*9	0.36 × RVCC	—	0.64 × RVCC	VP-P
		Load impedance	100	—	—	kΩ

Note 1. Pseudo sign wave output, 7-bit D/A output.

Note 2. The center of amplitude is COM voltage. Described values are default (±0%) of output adjustment.

Note 3. The output voltage can be adjusted in four steps of -40, -20, ±0, and +20% by adjust function.

Note 4. When the excitation frequency is over 22 kHz with resolver excitation signal external input (RDC3AnREF.EXIO = 0_B), set RDC3AnDIAG1.CVEDS = 1_B, and do not set using RD conversion error detection circuit (for high-speed rotation).

Note 5. Depends on external circuit.
Input voltage must be adjusted for COSMNT, SINMNT = 0.36 × RVCC to 0.64 × RVCC (VP-P).

Note 6. Input impedance with on-chip feed-back resistor. This is the default value (±0%).

Note 7. Can be adjusted from -40 to +40% in step of 10% by adjust function.

Note 8. Same as the frequency of resolver signal input.

Note 9. Must be adjusted within this range to keep angle conversion resolution.

Note 10. The phase error of the excitation component of the resolver excitation signal external input and the resolver signal input must be within 45º.

Note 11. When RDC3AnREF.EXIO = 0_B (i.e. the external excitation signal input is set), this impedance pulls the RSO (R1E) and COM (R2E) pins down to RVSS.

39.5.3 Error Detect Characteristics

Table 39.35 Error Detect Characteristics

Error Detect Item		Set Threshold (Default Setting)	Detect Time
Resolver signal error monitor output amplitude voltage*1	Set register RDC3AnDIAG0.EXCETH[7:0]	0.102 × (RVCC±5%) [Vp-p]	220 [µs] (typ.), 2 [ms] (max.)
Breaking of resolver signal (Direct current bias supply method) VSINMNT-VCOM or VCOSMNT-VCOM*2	The register used for setting threshold when DC resolver is selected RDC3AnDIAG0.SGBDTH[7:0]	COM + 0.35 × (RVCC±5%) [VDC]	10 [ms] (max.)
	The register used for setting threshold when VR resolver is selected RDC3AnDIAG0.SGBTH[7:0]	COM + 0.08 × (RVCC±5%) [VDC]	
R/D conversion error (over control declination) Recognition level for internal control declination (ε)*3	High side	$00CA8_H$	*4
	Low side	$1FFF3_H$	
Resolver signal power short error	Pins S1, S2, S3, and S4	0.9 × (RVCC±5%) [VDC]	0.08 [ms] (max.)
	Pins RSO and COM	0.8 × (RVCC±5%) [VDC]	
Resolver signal ground short error	Pins S1, S2, S3, and S4	0.1 × (RVCC±5%) [VDC]	0.08 [ms] (max.)
	Pins RSO and COM	0.2 × (RVCC±5%) [VDC]	
Sum square amplitude error Integral value of sum square of monitor output amplitude voltage (sin, cos) with an interval within the excitation period	High side	0.8 × (RVCC±5%) [Vp-p]	*5
	Low side	0.2 × (RVCC±5%) [Vp-p]	

Note 1. Determined as an error, when both of SINMNT and COSMNT become under threshold.

Note 2. Determined as an error, when DC level changed over threshold.

Note 3. Determined as over, when control declination becomes over high side threshold, or under low side threshold.

Note 4. Determined as an error, when control declination recognition rate becomes over 50% as average of period set by the EDPS [1:0] bits in the RDC3AnDIAG1 register (default is about 7.4 ms). It might not be detected, when the error continuous period is shorter than detect period.

Note 5. Dependent on the excitation frequency and amplitude error excitation period count settings.

39.6 Code Flash Characteristics

Table 39.36 Code Flash Basic Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Number of programming times*1	CWRT	Retained for 20 years *2	1000	—	—	Times
Programming temperature	TPRG	Tj	–40	—	+150	ºC
Reading temperature	TREAD	Tj	–40	—	+150	ºC

Note 1. The number of programming times is the number of erasure of each block. If the number of programming times is n (n = 1000), each block can be erased n times. For example, if 256-byte data is written 128 times to different addresses of a 32KB block and then the block is erased, the number of programming times is counted as 1. Note, however, that writing data to the same address multiple times for one erasure is not allowed (overwrite is prohibited).

Note 2. This is the case when the average Ta = 85°C. This retained period is from when the erasure of the code flash memory has been normally completed.

Table 39.37 Code Flash Programming Characteristics

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item	Condition	Block size	Min.	Typ.	Max.	Unit
Programming time	Number of programming times < 100	256 B	—	0.4*1	6*1	ms
		32 KB	—	80	360	ms
	Number of programming times ≥ 100	256 B	—	0.5*1	7.2*1	ms
		32 KB	—	96	432	ms
Erasing time*1	Number of programming times < 100	8 KB	—	39	120	ms
		32 KB		141	480	ms
	Number of programming times ≥ 100	8 KB	—	47	144	ms
		32 KB	—	169	576	ms

Note 1. Only the hardware processing time is included. Software overhead is not included.

39.7 Data Flash Characteristics

Table 39.38 Data Flash Basic Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Number of programming times*1	CWRT	Retained for 20 years*2	125000	—	—	Times
		Retained for 3 years*2	250000	—	—	Times
Programming temperature	TPRG	Tj	–40	—	+150	ºC
Reading temperature	TREAD	Tj	–40	—	+150	ºC

Note 1. The number of programming times is the number of erasure of each block. If the number of programming times is n (n = 125000), each block can be erased n times. For example, if 4 byte data is written 16 times to different addresses of a 64KB block and then the bock is erased, the number of programming times is counted as 1. Note, however, that writing data to the same address multiple times for one erasure is not allowed (overwrite is prohibited).

Note 2. This is the case when the average Ta = 85°C. This retained period is from when the erasure of the data flash memory has been normally completed.

Table 39.39 Data Flash Programming Characteristics

Conditions: SYSVCC = VCC = 4.5 V to 5.5 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC, A2VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC,
A2VREFH = 4.5 V to A2VCC, RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = A2VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item	Block size	Min.	Typ.	Max.	Unit
Programming time*1	4 B	—	0.16	1.7	ms
Erasing time*1	64 B	—	1.7	10	ms
Blank checking time*1	4 B	—	—	30	µs
	64 B	—	—	100	µs

Note 1. Only the hardware processing time is included. Software overhead is not included.

39.8 Thermal Characteristics

39.8.1 Parameters

Table 39.40 Thermal Resistance of RH850/C1M-A

Package	Parameter	Estimate	Unit	Note
FPBGA1717-252	θja	20.6	ºC/ W	JESD51-9 compliant (4 layers)
	Ψjb	14.0	ºC/ W	JESD51-9 compliant (4 layers)
	Tb_inc	6.7	ºC/ W	JESD51-9 compliant (4 layers)
	Ψjt	0.22	ºC/ W	JESD51-9 compliant (4 layers)
LQFP2424-176	θja	30.0	ºC/ W	JESD51-7 compliant (4 layers)
	Ψjb	22.8	ºC/ W	JESD51-7 compliant (4 layers)
	Tb_inc	7.3	ºC/ W	JESD51-7 compliant (4 layers)
	Ψjt	0.34	ºC/ W	JESD51-7 compliant (4 layers)

Note: Thermal resistance and thermal characteristics parameters will change according to the usage environment.

39.8.2 Assumed Board

Table 39.41 JESD51-9 Compliant (4 layers)

	Board Size (mm)		
Package	X	Y	Area (mm^2)
L board	101.6	114.3	11612.88
Remaining copper rates		Thickness of conductors	
50-95-95-50%		70-35-35-70 µm	

Table 39.42 JESD51-7 Compliant (4 layers)

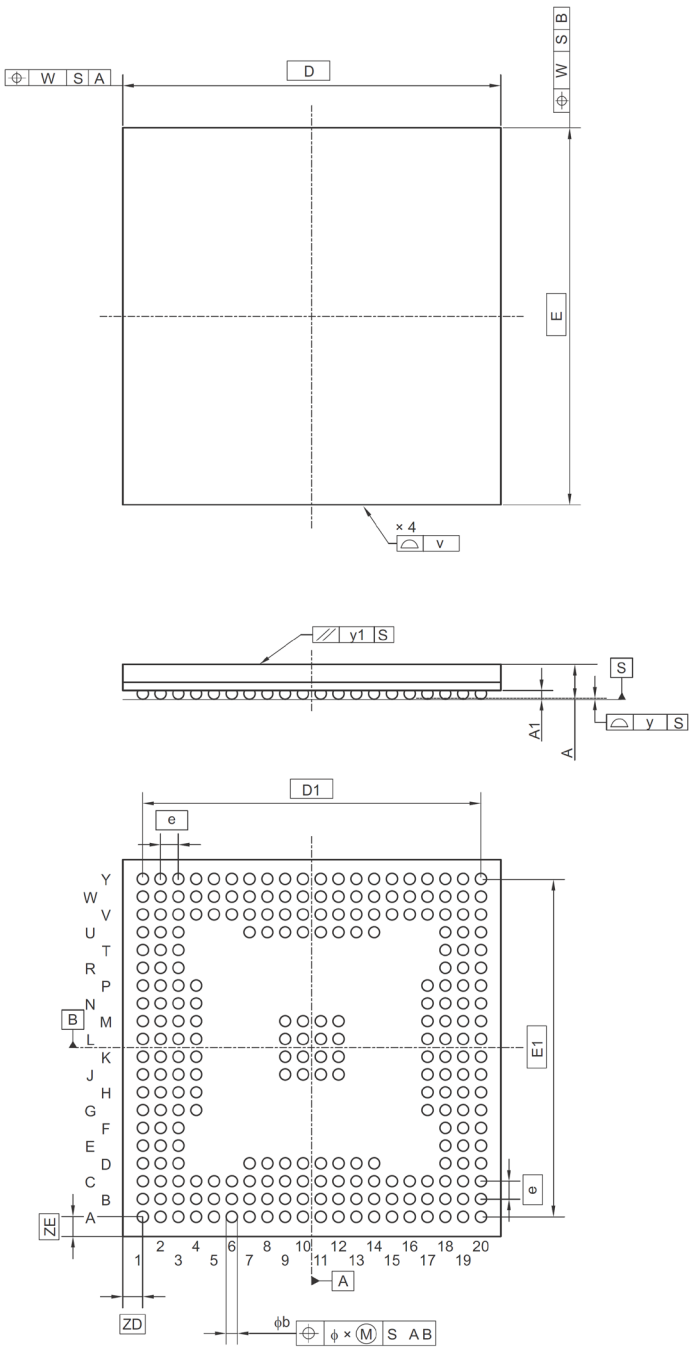
	Board Size (mm)		
Package	X	Y	Area (mm^2)
L board	76.2	114.3	8709.66
Remaining copper rates		Thickness of conductors	
50-95-95-50%		70-35-35-70 µm	

Appendix A Package Dimensions

- BGA252

JEITA Package Code	RENESAS Code	Previous Code	MASS (Typ) [g]
P-FBGA252-17x17-0.80	PRBG0252GB-A	—	0.90

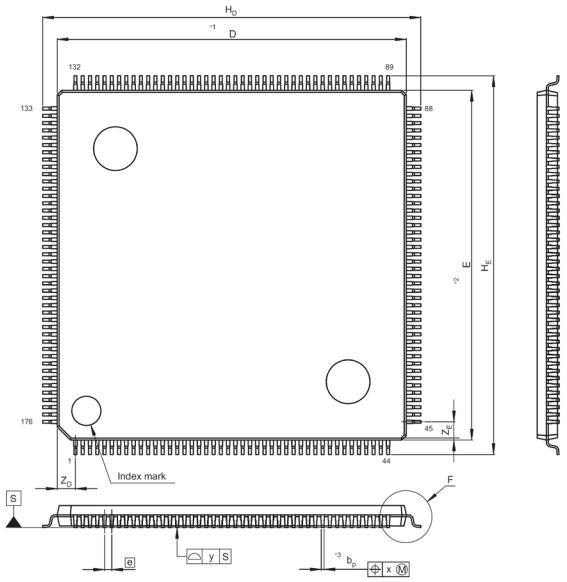
Unit: mm

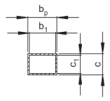


Reference Symbol	Dimensions in millimeters		
	Min	Nom	Max
D	—	17.00	—
D1	—	15.20	—
E	—	17.00	—
E1	—	15.20	—
v	—	—	0.15
w	—	—	0.20
e	—	0.80	—
A	—	1.58	2.00
A1	0.30	0.35	0.40
b	0.49	0.54	0.59
x	—	—	0.08
y	—	—	0.10
y1	—	—	0.20
ZD	—	0.90	—
ZE	—	0.90	—

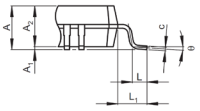
- QFP176

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LFQFP176-24x24-0.50	PLQP0176KB-A	176P6Q-A/FP-176E/FP-176EV	1.8g





Terminal cross section



Detail F

NOTE)
1. DIMENSIONS "*1" AND "*2" DO NOT INCLUDE MOLD FLASH.
2. DIMENSION "*3" DOES NOT INCLUDE TRIM OFFSET.

Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	23.9	24.0	24.1
E	23.9	24.0	24.1
A_2	—	1.4	—
H_D	25.8	26.0	26.2
H_E	25.8	26.0	26.2
A	—	—	1.7
A_1	0.05	0.1	0.15
b_p	0.15	0.20	0.25
b_1	—	0.18	—
c	0.09	0.145	0.20
c_1	—	0.125	—
θ	0°	—	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.10
Z_D	—	1.25	—
Z_E	—	1.25	—
L	0.35	0.5	0.65
L_1	—	1.0	—

RH850/C1M-A1, RH850/C1M-A2 User's Manual: Hardware

Publication Date: Rev.0.50 Apr 26, 2016
Rev.1.20 May 31, 2018

Published by: Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700, Fax: +44-1628-651-804

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709 Quantum Plaza, No.27 ZhichunLu, Haidian District, Beijing, 100191 P. R. China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, 200333 P. R. China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338

Colophon 5.0

RH850/C1M-A1, RH850/C1M-A2

Renesas Electronics Corporation

R01UH0607EJ0120