User's Manual

32

EC-1

User's Manual: Hardware

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (http://www.renesas.com).

Renesas Electronics
www.renesas.com

Rev.1.50 Jan, 2022

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 be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. 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.
6. 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.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. 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.
9. 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.
10. 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.
11. 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.
12. 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.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

© 2022 Renesas Electronics Corporation. All rights reserved.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can 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 must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. 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. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. 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 V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, 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 V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

How to Use This Manual

1. Objective and Target Users

This manual was written to explain the hardware functions and electrical characteristics of this LSI to the target users, i.e. those who will be using this LSI in the design of application systems. Target users are expected to understand the fundamentals of electrical circuits, logic circuits, and microcomputers.
This manual is organized in the following items: an overview of the product, descriptions of the CPU, system control functions, and peripheral functions, electrical characteristics of the device, and usage notes.

When designing an application system that includes this LSI, take all points to note into account.
Points to note are given in their contexts and at the final part of each section, and in the section giving usage notes.

The list of revisions is a summary of major points of revision or addition for earlier versions. It does not cover all revised items. For details on the revised points, see the actual locations in the manual.

The following documents have been prepared for the EC-1. Before using any of the documents, please visit our web site to verify that you have the most up-to-date available version of the document.

Document Type	Contents	Document Title	Document No.
Data Sheet	Overview of hardware and electrical characteristics	—	—
User's manual: Hardware	Hardware specifications (pin assignments, memory maps, peripheral specifications, electrical characteristics, and timing charts) and descriptions of operation	EC-1 User's manual: Hardware	This User's manual
User's manual: Software	Please find this information from the Arm® Ltd. home page.		
Application Note	Examples of applications and sample programs	—	—
Renesas Technical Update	Preliminary report on the specifications of a product, document, etc.	—	—

2. Description of Registers

Each register description includes a bit chart, illustrating the arrangement of bits, and a table of bits, describing the meanings of the bit settings. The standard format and notation for bit charts and tables are described below.

X.X.X ... Register

Address(es): xxxx xxxxxh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	... [1:0]		...4	—	—	—	...0
Value after reset:	x	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	... 0	... Bit	0: ..... 1: Setting prohibited	R/W
b3 to b1	—	Reserved	The read value is 0. The write value should be 0.	R/W
b4	... 4	... Bit	0: 1:	R
b6, b5	... [1:0]	... Bit	0 0: ..... 0 1: ..... Settings other than above are prohibited.	R/(W)*
b7	—	Reserved	The read value is undefined. Writing to this bit has no effect.	R

(1) R/W: The bit or field is readable and writable.
R/(W): The bit or field is readable and writable. However, writing to this bit or field has some limitations. For details on the limitations, see the description or notes of respective registers.
R: The bit or field is readable. Writing to this bit or field has no effect.

(2) Reserved. Make sure to use the specified value when writing to this bit or field; otherwise, the correct operation is not guaranteed.

(3) Setting prohibited. The correct operation is not guaranteed if such a setting is performed.

3. List of Abbreviations and Acronyms

Abbreviation	Full Form
bps	bits per second
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
Hi-Z	High Impedance
I/O	Input/Output
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Non-Connect
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
UART	Universal Asynchronous Receiver/Transmitter

4. Description of the Access Size

Access size:
8 bits = Byte
16 bits = Word
32 bits = Longword

All trademarks and registered trademarks are the property of their respective owners.

Table of Contents

EC-1

R01UH0691EJ0150
Rev.1.50
Jan. 31, 2022

150 MHz, MCU with Arm® Cortex®-R4, on-chip FPU, 249 DMIPS, EtherCAT, USB 2.0 high-speed, CAN, various communications interfaces such as an SPI multi-I/O bus controller, and safety functions

Features

■ On-chip 32-bit Arm Cortex-R4 processor

- High-speed realtime control with maximum operating frequency of 150 MHz
  Capable of 249 DMIPS
- On-chip 32-bit Arm Cortex-R4 (revision r1p4)
- Tightly coupled memory (TCM) with ECC: 512 Kbytes/32 Kbytes
- Instruction cache/data cache with ECC: 8 Kbytes per cache
- High-speed interrupt
- The FPU supports addition, subtraction, multiplication, division, multiply-and-accumulate, and square-root operations at single-precision and double-precision.
- Harvard architecture with 8-stage pipeline
- Supports the memory protection unit (MPU)
- Arm CoreSight architecture, includes support for debugging through JTAG and SWD interfaces

■ Low power consumption

- Standby mode, and module stop function

■ Data transfer

- DMACAa: 16 channels × 2 units

■ Event link controller

- Module operations can be started by event signals rather than by interrupt handlers.
- Linked operation of modules is available even while the CPU is in the sleep state.

■ Reset and power supply voltage control

- Three reset sources including a pin reset
- Dual power-voltage configuration: 3.3 V (I/O unit), 1.2 V (internal)

■ Clock functions

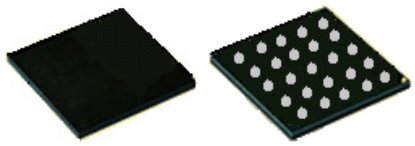
- Oscillator input frequency: 25 MHz
- CPU clock frequency: 150 MHz
- Low-speed on-chip oscillator (LOCO): 240 kHz

■ Independent watchdog timer

- Operated by a clock signal obtained by frequency-dividing the clock signal from the low-speed on-chip oscillator: Up to 120 kHz

■ Safety functions

- Register write protection, input clock oscillation stop detection, CRC, and IWDTa
- An error control module is incorporated to generate a pin signal output, interrupt, or internal reset in response to errors originating in the various modules.



PLBG0196GA-B 12x12mm, 0.8-mm pitch

■ Various communications interfaces

- EtherCAT slave controller: 2 ports
- USB 2.0 high-speed host/function : 1 channel
- CAN (compliant with ISO11898-1): 1 channel
- SCIFA with 16-byte transmission and reception FIFOs: 5 channels
- I^2C bus interface: 1 channel for transfer at up to 400 kbps
- RSPIa: 2 channels
- SPIBSC: Provides a single interface for multi-I/O compatible serial flash memory

■ 8 extended-function timers

- 16-bit CMT (6 channels), 32-bit CMTW (2 channels)

■ General-purpose I/O ports

- 5-V tolerance, open drain, input pull-up

■ Multi-function pin controller

- The locations of input/output functions for peripheral modules are selectable from among multiple pins.

■ Operating temperature range*1

- Tj = -40°C to +125°C
  Tj: Junction temperature

Note 1. When the operating temperature (junction temperature) is 110°C or higher, refer to the precautions regarding the high-temperature operation on the EC-1 (R01AN3998).

1. Overview

1.1 Outline of Specifications

This LSI circuit is a high-performance industrial MCU equipped with the Arm® Cortex®-R4 processor with FPU, and incorporating integrated peripheral functions necessary for system configuration. Table 1.1 lists the specifications in outline, and Table 1.2 gives a list of functions.

Table 1.1 Outline of Specifications (1 / 3)

Classification	Module/Function	Description
CPU	Central processing unit (Cortex-R4)	• Operating frequency 196-pin FBGA:150 MHz • 32-bit CPU Cortex-R4 designed by Arm (core revision r1p4) • Address space: 4 Gbytes • Instruction cache: 8 Kbytes (with ECC) • Data cache: 8 Kbytes (with ECC) • Tightly coupled memory (TCM) ATCM: 512 Kbytes (with ECC) BTCM: 32 Kbytes (with ECC) • Instruction set: Arm v7-R architecture, so support includes Thumb® and Thumb-2 • Data arrangement Instructions: Little endian Data: Little endian • Memory protection unit (MPU)
	FPU (Cortex-R4)	• Supports addition, subtraction, multiplication, division, multiply-and-accumulate, and square-root operations at single- and double-precision. • Registers 32-bit single-word registers: 32 bits × 32 (can be used as 16 double-word registers: 64 bits × 16)
Operating modes		• SPI boot mode (for booting up from serial flash memory)
Clock	Clock generation circuit	• The input clock can be selected from an external resonator. • Detection of input clock oscillation stopping • The following clocks are generated. CPU clock:150 MHz (fixed) System clock: 150 MHz (fixed) High-speed peripheral module clock: 150 MHz (fixed) Low-speed peripheral module clock: 75 MHz (fixed) Low-speed on-chip oscillator: 240 kHz (fixed)
Reset		RES# pin reset, error control module (ECM) reset, software reset
Low power	Low power consumption	• Standby mode (Cortex-R4) • Module stop function
Interrupt	Cortex-R4 vector interrupt controller (VIC)	• Peripheral function interrupts: 87 • External interrupts: 15 sources (NMI, IRQ0 to IRQ4, IRQ6, IRQ7, IRQ9, IRQ11 to IRQ14, ETH0_INT, ETH1_INT) • Non-maskable interrupts: 2 sources • Sixteen levels specifiable for the order of priority
Data transfer	Direct memory access controller (DMACAa)	• 2 units (16 channels for unit 0, 16 channels for unit 1) • Transfer modes: Single transfer mode and block transfer mode • Transfer size Unit 0: 1/2/4/16/32/64 bytes Unit 1: 1/2/4/16 bytes • Activation sources: External interrupts, on-chip peripheral module requests, and software requests
I/O ports	General-purpose I/O ports	• 196-pin FBGA I/O pins: 115 Input pins: 8 Pull-up/pull-down resistors: 115 5-V tolerance: 5
Event link controller (ELC)		• Event signals can be interlinked with the operation of modules. • In particular, the operation of timer modules can be started by input event signals. • Event-linked operation of signals of ports B and E is to be possible.

Table 1.1 Outline of Specifications (2 / 3)

Classification	Module/Function	Description
Multi-function pin controller (MPC)		The locations of input/output functions are selectable from among multiple pins.
Timer	Compare match timer (CMT)	• (16 bits × 2 channels) × 3 units • Select from among four counter-input clock signals for each channel (with maximum operating frequency of 75 MHz) • Event linking by the ELC (channel 1 of unit 0 only)
	Compare match timer W (CMTW)	• (32 bits × 1 channel) × 2 units • Compare-match, input-capture input, and output-comparison output are available. • Select from among four counter-input clock signals for each channel (with maximum operating frequency of 75 MHz) • Interrupt requests can be output in response to compare-match, input-capture, and output-comparison events. • Digital noise filter function for signals on the input capture pins • Event linking by the ELC
	Watchdog timer (WDTA)	• 14 bits × 1 channel • Select from among six counter-input clock signals for each channel (with maximum operating frequency of 75 MHz)
	Independent watchdog timer (IWDTa)	• 14 bits × 1 channel • Counter-input clock: Low-speed on-chip oscillator (LOCO)/2 Dedicated clock/1, dedicated clock/16, dedicated clock/32, dedicated clock/64, dedicated clock/128, dedicated clock/256 (with maximum operating frequency of 120 kHz)
Communication function	EtherCAT Slave Controller (ESC)*1	• 1 channel (2 ports) • EtherCAT Slave Controller IP core (made by Beckhoff Automation GmbH) implemented
	USB 2.0 HS host/ function module	• 1 port • Compliance with the USB 2.0 specification • Transfer rate High speed (480 Mbps), full speed (12 Mbps) • Communications buffer Incorporates 1 Kbyte of RAM for host mode Incorporates 8 Kbytes of RAM for function mode
	Serial communication interface with FIFO (SCIFA)	• 5 channels • Serial communications modes: Asynchronous, clock synchronous*2 • On-chip baud rate generator allows selection of the desired bit rate • Choice of LSB-first or MSB-first transfer • Both the transmission and reception sections are equipped with 16-byte FIFO buffers, allowing continuous transmission and reception. • Bit rate modulation
	I^2C bus interface (RIICa)	• 1 channel • Supports I^2C bus format • Supports the multi-master • Max. transfer rate: 400 kbps
	CAN module (RSCAN)	• 1 channel • Compliance with the ISO11898-1 specification (standard frame and extended frame) • Message buffers Max. 64 × 1 channel of receive message buffers 16 transmit message buffers per channel • Max. transfer rate: 1 Mbps

Table 1.1 Outline of Specifications (3 / 3)

Classification	Module/Function	Description
Communication function	Serial peripheral interface (RSPIa)	• 2 channels • RSPI transfer facility Using the MOSI (master out slave in), MISO (master in slave out), SSL (slave select), and RSPCK (RSPI clock) signals enables serial transfer through SPI operation (four lines) or clock-synchronous operation (three lines) Capable of handling serial transfer as a master or slave • Data formats Switching between MSB first and LSB first The number of bits in each transfer can be changed to any number of bits from 8 to 16, or 20, 24, or 32 bits. 128-bit buffers for transmission and reception Up to four frames can be transmitted or received in a single transfer operation (with each frame having up to 32 bits) • Buffered structure Double buffers for both transmission and reception • RSPCK can be stopped automatically with the receive buffer full for master reception • Event linking by the ELC
	SPI multi I/O bus controller (SPIBSC)	• 1 channel • One serial flash memory with multiple I/O bus sizes (single/dual/quad) can be connected. • External address space read mode (built-in read cache) • SPI operating mode • Clock polarity and clock phase can be selected. • Maximum transfer rate: 300 Mbps (for quad)
Safety	Register write protection function	Protects important registers from being overwritten for in case a program runs out of control.
	CRC calculator (CRC)	• CRC code generation for arbitrary amounts of data in 8-, 16-, or 32-bit units • Select any of four generating polynomials: $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ (32-Ethernet), $X^{16} + X^{12} + X^5 + 1$ (16-CCITT), $X^8 + X^4 + X^3 + X^2 + 1$ (8-SAEJ1850), $X^8 + X^5 + X^3 + X^2 + X + 1$ (8-0x2F)
	Input clock oscillation stop function	Input clock oscillation stop detection: Available
	Clock monitor circuit (CLMA)	Monitors the abnormal output clock frequency from the PLL circuit or low-speed on-chip oscillator.
	Error control module (ECM)	• Generates an interrupt, internal reset, or error output for the error signal input from each module. • Time-out function • The error control is duplicated in the master and the checker.
Power supply voltage		VDD = PLLVDD0 = PLLVDD1 = DVDD_USB = 1.14 to 1.26 V VCCQ33 = VDD33_USB = 3.0 to 3.6 V
Operating temperature*3		Tj = -40 to +125°C
Package		196-pin FBGA: 12 × 12 mm, 0.8-mm pitch PLBG0196GA-B
Debugging interface		• CoreSight architecture designed by Arm • Debugging function by the JTAG/SWD interface, and trace function by the trace port/ SWV interface

Note 1. EtherCAT is a registered trademark of Beckhoff Automation GmbH, Germany.
Note 2. Channels 3 and 4 are used only in asynchronous mode.
Note 3. When the operating temperature (junction temperature) is 110°C or higher, refer to the precautions regarding the high-temperature operation on the EC-1(R01AN3998).

Table 1.2 List of Functions

Module/Function		EC-1 196 Pins
Interrupt	External interrupt	NMI, IRQ0 to 4, IRQ6, IRQ7, IRQ9, IRQ11 to IRQ14, ETH0_INT, ETH1_INT
DMA	DMA controller (DMACAa)	ch0 to ch31
Timers	Compare match timer (CMT)	ch0 to ch5
	Compare match timer W (CMTW)	ch0, ch1
	Watchdog timer (WDTA)	ch0
	Independent watchdog timer (IWDTa)	Available
Communication function	EtherCAT slave controller (ESC)	2 ports
	USB 2.0 HS host/function module (USB)	ch0
	Serial communications interface with FIFO (SCIFA)	ch0 to ch4*1
	I^2C bus interface (RIICa)	ch1
	Serial peripheral interface (RSPIa)	ch0, ch1
	CAN module (RSCAN)	ch1
	SPI multi I/O bus controller (SPIBSC)	ch0
CRC calculator (CRC)		Available
Clock monitor circuit (CLMA)		Available
Event link controller (ELC)		Available

Note 1. Channels 3 and 4 are used only in asynchronous mode.

1.2 List of Products

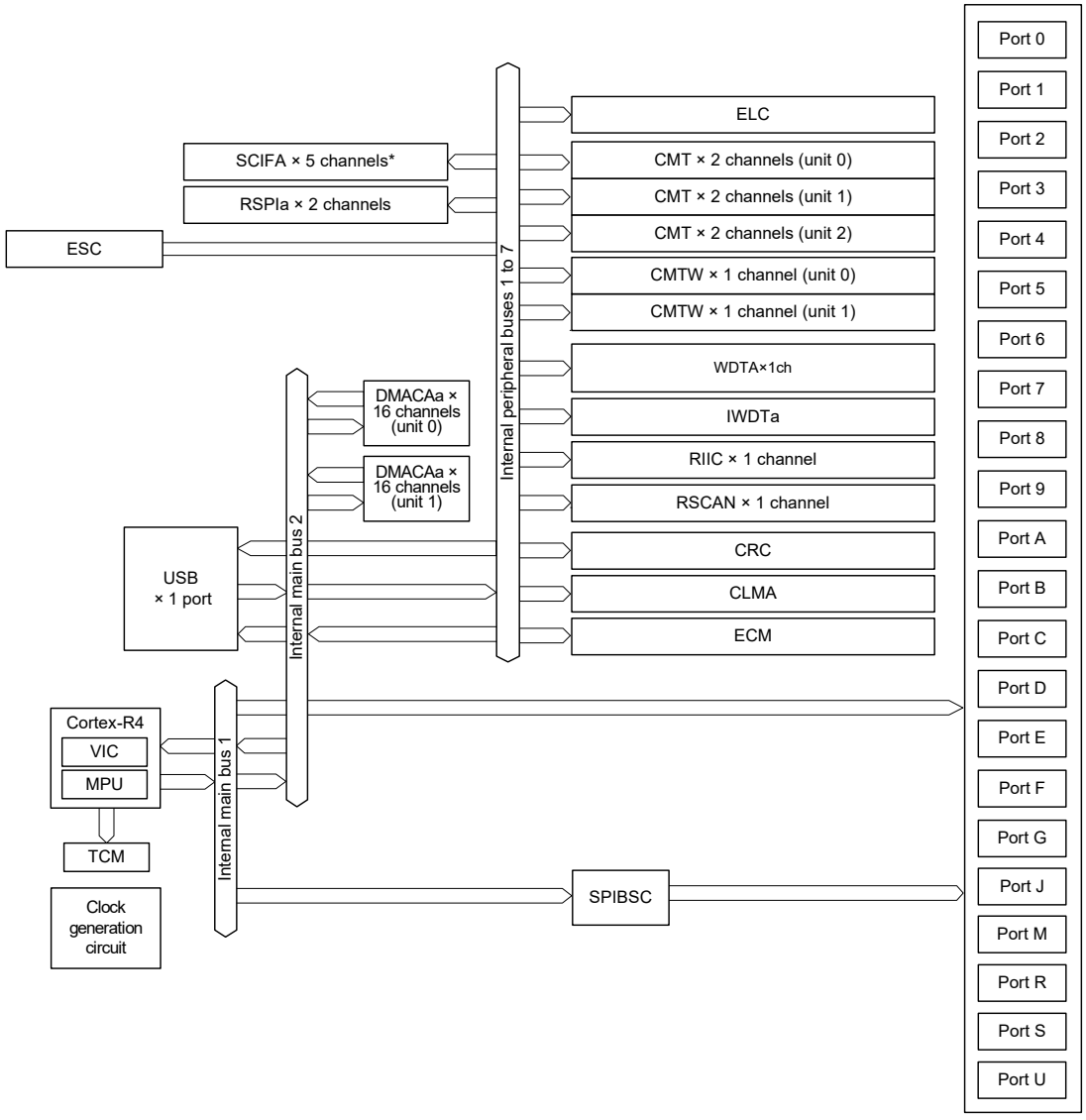
Table 1.3 is a list of products.

Table 1.3 List of Products

Part No.	Package	CPU	Operating Frequency (max.)
R9A06G043GBG	196 pins (PLBG0196GA-B)	Cortex-R4	150 MHz

1.3 Block Diagram

Figure 1.1 shows a block diagram of a 196-pin device.



ESC:	EtherCAT slave controller	CMT:	Compare match timer
DMACAa:	DMA controller	CMTW:	Compare match timer W
SPIBSC:	SPI multi I/O bus controller	RIICa:	I^2C bus interface
WDTA:	Watchdog timer	RSCAN:	CAN module
IWDTa:	Independent watchdog timer	CRC:	CRC (cyclic redundancy check) calculator
SCIFA:	Serial communication interface with FIFO	CLMA:	Clock monitor circuit
RSPIa:	Serial peripheral interface	ECM:	Error control module
USB:	USB 2.0 HS host/function module		
VIC:	Vector interrupt controller		
MPU:	Memory protection unit		
ELC:	Event link controller		

Note: Channels 3 and 4 are used only in asynchronous mode.

Figure 1.1 Block Diagram

1.4 Pin Functions

Table 1.4 lists the pin functions.

Table 1.4 Pin Functions (1 / 3)

Classifications	Pin Name	I/O	Description
Power supply	VDD	Input	Power supply pin. Connect this pin to the system power supply.
	VSS	Input	Ground pin. Connect this pin to the system power supply (0 V).
	VCCQ33	Input	Power supply pin for I/O pin
	PLLVDD0, PLLVDD1	Input	Power supply pins for the on-chip PLL oscillator
	PLLVSS0, PLLVSS1	Input	Ground pins for the on-chip PLL oscillator. Connect these pins to the system power supply (0 V).
Clock	XTAL	Output	Connected to a crystal resonator.
	EXTAL	Input	
	CLKOUT25M0, CLKOUT25M1	Output	Output the external clock for EtherCAT PHY.
System control	RES#	Input	Reset signal input pin. This LSI enters the reset state when this signal goes low.
	BSCANP	Input	Inputs the boundary scan enable signal. Boundary scan is enabled when this pin goes high. When not used, it should be driven low.
	ERROROUT#	Output	Outputs the error signal from the error control module (ECM).
	RSTOUT#	Output	Outputs the reset signal externally.
Debugging interface	TRST#	Input	Test reset pin for on-chip emulator
	TMS	I/O	Test mode select pin for on-chip emulator
	TDI	Input	Test data input pin for on-chip emulator
	TDO	Output	Test data output pin for on-chip emulator
	TCK	Input	Test clock pin for on-chip emulator
	TRACECLK	Output	Outputs the clock for synchronization with the trace data.
	TRACECTL	Output	Outputs the enable signal for trace control.
	TRACEDATA0 to TRACEDATA7	Output	Output the trace data.
Interrupt	NMI	Input	Inputs the non-maskable interrupt request signal.
	IRQ0 to IRQ4, IRQ6, IRQ7, IRQ9, IRQ11 to IRQ14	Input	Input the external interrupt request signal.
	ETH0_INT, ETH1_INT	Input	Input the EtherCAT PHY interrupt request signal.
Compare match timer W (CMTW)	TIC0 to TIC3	Input	CMTW input capture input pins
	TOC0 to TOC3	Output	CMTW output compare output pins
Serial communication interface with FIFO (SCIFA)	SCK0 to SCK2	I/O	Clock I/O pins
	RXD0 to RXD4	Input	Input the receive data.
	TXD0 to TXD4	Output	Output the transmit data.
	CTS0# to CTS2#	I/O	Hardware flow control input (transmission enable signal)/general output
	RTS0# to RTS2#	Output	Hardware flow control output (transmission request signal)/general output
I^2C bus interface (RIICa)	SCL1	I/O	Clock I/O pin. The bus can be directly driven by the N-channel open drain.
	SDA1	I/O	Data I/O pin. The bus can be directly driven by the N-channel open drain.

Table 1.4 Pin Functions (2 / 3)

Classifications	Pin Name	I/O	Description
EtherCAT slave controller (ESC)	ETH0_TXC, ETH1_TXC	Input	Transmission clock input pins
	ETH0_TXEN, ETH1_TXEN	Output	Output the transmission enable signal.
	ETH0_TXD0 to 3, ETH1_TXD0 to 3	Output	Output the transmission data signal.
	ETH0_RXC, ETH1_RXC	Input	Receive clock input pins
	ETH0_RXDV, ETH1_RXDV	Input	Input the receive data enable signal.
	ETH0_RXER, ETH1_RXER	Input	Input the receive data error signal.
	ETH0_RXD0 to 3, ETH1_RXD0 to 3	Input	Input the receive data signal.
	ETH_MDC	Output	Output the management interface clock.
	ETH_MDIO	I/O	Management data signal I/O pin
	PHYLINK0, PHYLINK1	Input	Input the PHY Link signal.
	PHYRESETOUT#	Output	Output the PHY RESET signal
	CATLEDRUN	Output	Outputs the EtherCAT RUN LED signal.
	CATIRQ	Output	Outputs the EtherCAT IRQ signal.
	CATLEDSTER	Output	Outputs the EtherCAT Dual-color state LED signal.
	CATLEDERR	Output	Outputs the EtherCAT error LED signal.
	CATLINKACT0, CATLINKACT1	Output	Output the EtherCAT link/activity LED signal.
	CATSYNC0, CATSYNC1	Output	Output the EtherCAT SYNC signal.
	CATLATCH0	Input	Input the EtherCAT LATCH signal.
	CATLATCH1	Input	Input the EtherCAT LATCH signal.
	CATI2CCLK	Output	Outputs the EtherCAT EEPROM I^2C clock signal.
	CATI2CDATA	I/O	Inputs/outputs the EtherCAT EEPROM I^2C data signal.
USB 2.0 host/function module	VDD33_USB	Input	Power supply input pin for USB
	VSS_USB	Input	Ground input pin for USB
	DVDD_USB	Input	Digital power supply input pin for USB
	USB_RREF	Input	Reference current input pin for USB. Connect this pin to the VSS_USB pin via 200Ω (±1%).
	USB_DP	I/O	USB bus D+ data I/O pin
	USB_DM	I/O	USB bus D- data I/O pin
	USB_VBUSEN	Output	Outputs the VBUS power enable signal for USB.
	USB_OVRCUR	Input	Inputs the overcurrent signal for USB.
	USB_VBUSIN	Input	USB cable connection/disconnection detection input pin
CAN module (RSCAN)	CRXD1	Input	Receive data input pin
	CTXD1	Output	Transmit data output pin
Serial peripheral interface (RSPIa)	RSPCK0, RSPCK1	I/O	Clock I/O pins
	MOSI0, MOSI1	I/O	Master transmit data I/O pins
	MISO0, MISO1	I/O	Slave transmit data I/O pins
	SSL00, SSL10	I/O	Slave select signal I/O pins
	SSL01, SSL02, SSL03, SSL11	Output	Slave select signal output pins

Table 1.4 Pin Functions (3 / 3)

Classifications	Pin Name	I/O	Description
SPI multi I/O bus controller (SPIBSC)	SPBCLK	Output	Clock output pin
	SPBSSL	Output	Slave select signal output pin
	SPBMO/SPBIO0	I/O	Master transmit data/data 0 I/O pin
	SPBMI/SPBIO1	I/O	Master input data/data 1 I/O pin
	SPBIO2, SPBIO3	I/O	Data 2, data 3 I/O pins
I/O ports	P00	I/O	1-bit I/O pin
	P10, P12, P16, P17	I/O	4-bit I/O pins
	P20 to P23, P25 to P27	I/O	7-bit I/O pins
	P33 to P35	I/O	3-bit I/O pins
	P40, P42, P44	I/O	3-bit I/O pins
	P50 to P54, P56	I/O	6-bit I/O pins
	P60 to P66	I/O	7-bit I/O pins
	P70 to P77	I/O	8-bit I/O pins
	P80 to P87	I/O	8-bit I/O pins
	P90 to P97	I/O	8-bit I/O pins
	PA0 to PA7	I/O	8-bit I/O pins
	PB0 to PB7	I/O	8-bit I/O pins
	PC0 to PC7	Input	8-bit input pins
	PD5 to PD7	I/O	3-bit I/O pins
	PF5 to PF7	I/O	3-bit I/O pins
	PG2 to PG6	I/O	5-bit I/O pins
	PJ0 to PJ7	I/O	8-bit I/O pins
	PM1 to PM7	I/O	7-bit I/O pins
	PR1	I/O	1-bit I/O pin
	PS0 to PS7	I/O	8-bit I/O pins
	PU7	I/O	1-bit I/O pin
Others	IC0	Input	Connect to VSS via a resistor (pull-down)

1.5 Pin Assignments

Figure 1.2 show the pin arrangement. Table 1.5 show the pin assignments. Table 1.6 show the lists of pin functions.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
A	VSS	PC2	PJ2	PF7	PB5	PB2	PC1	PF5	PD6	P56	P51	IC0	IC0	VSS	A
B	PJ7	PJ4	PJ3	PJ1	PB6	PB1	PB7	P87	PD5	P53	IC0	IC0	P16	P97	B
C	P83	PJ5	PC3	PJ0	PB4	PB0	PF6	PD7	P54	VCCQ33	VCCQ33	P17	P96	P95	C
D	P84	P81	PJ6	VCCQ33	PB3	PC0	P86	P52	P50	VSS	VSS	PA7	P94	P12	D
E	TRST#	P85	P82	P80	VCCQ33	VDD	VDD	VDD	VDD	VCCQ33	VCCQ33	P90	P92	P93	E
F	P34	P33	ERROROUT#	P35	PLLVDD1	VDD	VSS	VSS	VSS	VDD	P91	PA4	PA5	PA6	F
G	PC4	PC5	TCK	TMS	PLLVSS1	VSS	VSS	VSS	VSS	VDD	PA3	P77	PA2	PA1	G
H	VCCQ33	BSCANP	PU7	IC0	PLLVDD0	VSS	VSS	VSS	VSS	VDD	P74	P75	P76	PA0	H
J	EXTAL	VSS	PM1	RES#	PLLVSS0	VDD	VSS	VSS	VSS	VDD	PE7	P71	P72	P73	J
K	XTAL	VSS	PM4	RSTOUT#	VDD33_USB	VDD	VDD	VDD	VDD	VCCQ33	PE1	PE5	PE6	P70	K
L	VSS	PM3	USB_REF	P62	VCCQ33	PG2	VDD	PR1	P27	VDD	VCCQ33	PS6	PE3	PE4	L
M	PM2	PM6	VSS_USB	P60	P64	PG3	PG5	P21	P26	P44	PS0	P00	PS7	PE2	M
N	PM7	PM5	DVDD_USB	P61	P63	PG4	PG6	P22	P20	P42	PS1	PS3	PS5	PE0	N
P	VSS_USB	USB_DM	USB_DP	P66	P65	PC6	PC7	P23	P25	P40	P10	PS2	PS4	VSS	P
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	

Figure 1.2 Pin Arrangement (196-Pin FBGA) (Top View)

Table 1.5 Pin Assignments (196-Pin FBGA) (1 / 5)

Pin Number	Pin Name
A1	VSS
A2	PC2 / ETH0_TXC
A3	PJ2 / ETH0_TXD1
A4	PF7 / IRQ7
A5	PB5 / ETH_MDIO
A6	PB2 / ETH1_RXC
A7	PC1 / ETH1_RXD3
A8	PF5 / ETH1_TXEN
A9	PD6 / ETH1_TXD2
A10	P56
A11	P51 / PHYLINK1
A12	IC0
A13	IC0
A14	VSS
B1	PJ7 / ETH0_RXD3
B2	PJ4 / ETH0_RXD0
B3	PJ3 / ETH0_TXD0
B4	PJ1 / ETH0_TXD2
B5	PB6 / ETH_MDC
B6	PB1 / ETH1_RXER
B7	PB7 / ETH1_RXD1
B8	P87 / ETH1_TXC
B9	PD5 / ETH1_TXD3
B10	P53 / ETH1_INT
B11	IC0
B12	IC0
B13	P16
B14	P97 / IRQ7
C1	P83 / IRQ11 / CATLINKACT0 / TXD4
C2	PJ5 / ETH0_RXD1
C3	PC3 / ETH0_RXC
C4	PJ0 / ETH0_TXD3
C5	PB4 / ETH0_RXER / CATSYNC0 / CATLATCH0 / RXD3
C6	PB0 / ETH1_RXDV
C7	PF6 / ETH1_RXD0
C8	PD7 / ETH1_TXD1
C9	P54 / CLKOUT25M1
C10	VCCQ33
C11	VCCQ33
C12	P17 / PHYRESETOUT#
C13	P96
C14	P95 / IRQ13 / CTS2#
D1	P84 / CATLINKACT1 / RXD4
D2	P81 / ETH0_RXER

Table 1.5 Pin Assignments (196-Pin FBGA) (2 / 5)

Pin Number	Pin Name
D3	PJ6 / ETH0_RXD2
D4	VCCQ33
D5	PB3 / IRQ3 / PHYRESETOUT# / TXD3 / CTXD1
D6	PC0 / ETH1_RXD2
D7	P86 / ETH1_TXD0
D8	P52 / ETH0_INT
D9	P50 / PHYLINK0
D10	VSS
D11	VSS
D12	PA7 / IRQ7 / RTS2#
D13	P94 / IRQ4 / RTS2#
D14	P12
E1	TRST#
E2	P85 / CLKOUT25M0
E3	P82 / ETH0_TXEN
E4	P80 / ETH0_RXDV
E5	VCCQ33
E6	VDD
E7	VDD
E8	VDD
E9	VDD
E10	VCCQ33
E11	VCCQ33
E12	P90 / TXD4
E13	P92 / TOC3 / RXD2
E14	P93 / TIC3 / SCK2
F1	P34 / TDI
F2	P33 / TDO
F3	ERROROUT#
F4	P35 / NMI
F5	PLLVDD1
F6	VDD
F7	VSS
F8	VSS
F9	VSS
F10	VDD
F11	P91 / TXD2
F12	PA4 / ETH1_INT / RXD2
F13	PA5 / ETH0_INT / TXD2
F14	PA6 / IRQ6 / CTS2#
G1	PC4 / CATI2CCLK
G2	PC5 / CATI2CDATA
G3	TCK
G4	TMS
G5	PLLVSS1

Table 1.5 Pin Assignments (196-Pin FBGA) (3 / 5)

Pin Number	Pin Name
G6	VSS
G7	VSS
G8	VSS
G9	VSS
G10	VDD
G11	PA3 / SCK2
G12	P77 / RSPCK0 / TRACEDATA5
G13	PA2 / SSL02
G14	PA1 / MISO0 / TRACEDATA7
H1	VCCQ33
H2	BSCANP
H3	PU7 / CATIRQ
H4	IC0
H5	PLLVDD0
H6	VSS
H7	VSS
H8	VSS
H9	VSS
H10	VDD
H11	P74 / CTS1# / SSL03 / TRACEDATA2
H12	P75 / IRQ13 / SSL00 / TRACEDATA3
H13	P76 / SSL01 / TRACEDATA4
H14	PA0 / MOSI0 / TRACEDATA6
J1	EXTAL
J2	VSS
J3	PM1 / CATLEDERR
J4	RES#
J5	PLLVSS0
J6	VDD
J7	VSS
J8	VSS
J9	VSS
J10	VDD
J11	PE7 / SCK1 / RSPCK0 / TRACEDATA7
J12	P71 / TOC2 / SCK1 / TRACECTL
J13	P72 / TIC2 / TXD1 / TRACEDATA0
J14	P73 / IRQ3 / RXD1 / TRACEDATA1
K1	XTAL
K2	VSS
K3	PM4 / CATLEDRUN
K4	RSTOUT#
K5	VDD33_USB
K6	VDD
K7	VDD
K8	VDD

Table 1.5 Pin Assignments (196-Pin FBGA) (4 / 5)

Pin Number	Pin Name
K9	VDD
K10	VCCQ33
K11	PE1 / SSL03 / TRACEDATA1
K12	PE5 / TXD1 / MOSI0 / TRACEDATA5
K13	PE6 / IRQ6 / RXD1 / MISO0 / TRACEDATA6
K14	P70 / IRQ0 / RTS1# / USB_OVRCUR / TRACECLK
L1	VSS
L2	PM3 / CATSYNC0 / CATLATCH0
L3	USB_RREF
L4	P62 / SPBCLK
L5	VCCQ33
L6	PG2 / TOC0 / RSPCK1
L7	VDD
L8	PR1 / IRQ9 / CTS1#
L9	P27 / RTS0#
L10	VDD
L11	VCCQ33
L12	PS6 / IRQ14 / RXD2
L13	PE3 / IRQ3 / CTS1# / SSL01 / TRACEDATA3
L14	PE4 / RTS1# / SSL00 / TRACEDATA4
M1	PM2 / CATSYNC1 / CATLATCH1
M2	PM6 / IRQ6 / CATLINKACT0
M3	VSS_USB
M4	P60 / SPBSSL
M5	P64 / SPBMI/SPBIO1
M6	PG3 / TIC1 / MISO1
M7	PG5 / SSL10
M8	P21 / IRQ1 / CTS0#
M9	P26
M10	P44 / IRQ12 / CTS0#
M11	PS0
M12	P00 / TRACECTL
M13	PS7 / TXD2
M14	PE2 / IRQ2 / SSL02 / TRACEDATA2
N1	PM7 / CATLINKACT1
N2	PM5 / CATLEDSTER
N3	DVDD_USB
N4	P61 / SPBIO3 / CTXD1
N5	P63 / SPBMO/SPBIO0
N6	PG4 / TOC1 / MOSI1
N7	PG6 / SSL11
N8	P22 / IRQ2 / SCK0
N9	P20
N10	P42 / RXD0
N11	PS1 / IRQ1

Table 1.5 Pin Assignments (196-Pin FBGA) (5 / 5)

Pin Number	Pin Name
N12	PS3
N13	PS5
N14	PE0 / TRACEDATA0
P1	VSS_USB
P2	USB_DM
P3	USB_DP
P4	P66 / IRQ14 / CTXD1 / USB_VBUSEN
P5	P65 / SPBIO2
P6	PC6 / SCL1 / USB_VBUSIN
P7	PC7 / TIC0 / SDA1 / CRXD1
P8	P23 / TXD0
P9	P25
P10	P40 / TXD0
P11	P10 / IRQ0 / TRACECLK
P12	PS2
P13	PS4
P14	VSS

Table 1.6 List of Pin and Pin Functions (196-Pin FBGA) (1 / 5)

Pin Number	Power Supply Clock System Control	I/O Port	Timer (CMTW)	Communication (ESC)	Communication (SPIBSC)	Communication (SCIFA, RSPIa, RIICa, RSCAN, USB)	Interrupt	Others (Connect to VSS via a resistor)
A1	VSS							
A2		PC2		ETH0_TXC				
A3		PJ2		ETH0_TXD1				
A4		PF7					IRQ7	
A5		PB5		ETH_MDIO				
A6		PB2		ETH1_RXC				
A7		PC1		ETH1_RXD3				
A8		PF5		ETH1_TXEN				
A9		PD6		ETH1_TXD2				
A10		P56						
A11		P51		PHYLINK1				
A12								IC0
A13								IC0
A14	VSS							
B1		PJ7		ETH0_RXD3				
B2		PJ4		ETH0_RXD0				
B3		PJ3		ETH0_TXD0				
B4		PJ1		ETH0_TXD2				
B5		PB6		ETH_MDC				
B6		PB1		ETH1_RXER				
B7		PB7		ETH1_RXD1				
B8		P87		ETH1_TXC				
B9		PD5		ETH1_TXD3				
B10		P53		ETH1_INT				
B11								IC0
B12								IC0
B13		P16						
B14		P97					IRQ7	
C1		P83		CATLINKACT0		TXD4	IRQ11	
C2		PJ5		ETH0_RXD1				
C3		PC3		ETH0_RXC				
C4		PJ0		ETH0_TXD3				
C5		PB4		ETH0_RXER / CATSYNC0 / CATLATCH0		RXD3		
C6		PB0		ETH1_RXDV				
C7		PF6		ETH1_RXD0				
C8		PD7		ETH1_TXD1				
C9		P54		CLKOUT25M1				
C10	VCCQ33							
C11	VCCQ33							

Table 1.6 List of Pin and Pin Functions (196-Pin FBGA) (2 / 5)

Pin Number	Power Supply Clock System Control	I/O Port	Timer (CMTW)	Communication (ESC)	Communication (SPIBSC)	Communication (SCIFA, RSPIa, RIICa, RSCAN, USB)	Interrupt	Others (Connect to VSS via a resistor)
C12		P17		PHYRESETOUT#				
C13		P96						
C14		P95				CTS2#	IRQ13	
D1		P84		CATLINKACT1		RXD4		
D2		P81		ETH0_RXER				
D3		PJ6		ETH0_RXD2				
D4	VCCQ33							
D5		PB3		PHYRESETOUT#		TXD3 / CTXD1	IRQ3	
D6		PC0		ETH1_RXD2				
D7		P86		ETH1_TXD0				
D8		P52		ETH0_INT				
D9		P50		PHYLINK0				
D10	VSS							
D11	VSS							
D12		PA7				RTS2#	IRQ7	
D13		P94				RTS2#	IRQ4	
D14		P12						
E1	TRST#							
E2		P85		CLKOUT25M0				
E3		P82		ETH0_TXEN				
E4		P80		ETH0_RXDV				
E5	VCCQ33							
E6	VDD							
E7	VDD							
E8	VDD							
E9	VDD							
E10	VCCQ33							
E11	VCCQ33							
E12		P90				TXD4		
E13		P92	TOC3			RXD2		
E14		P93	TIC3			SCK2		
F1	TDI	P34						
F2	TDO	P33						
F3	ERROROUT#							
F4		P35					NMI	
F5	PLLVDD1							
F6	VDD							
F7	VSS							
F8	VSS							
F9	VSS							
F10	VDD							

Table 1.6 List of Pin and Pin Functions (196-Pin FBGA) (3 / 5)

Pin Number	Power Supply Clock System Control	I/O Port	Timer (CMTW)	Communication (ESC)	Communication (SPIBSC)	Communication (SCIFA, RSPIa, RIICa, RSCAN, USB)	Interrupt	Others (Connect to VSS via a resistor)
F11		P91				TXD2		
F12		PA4		ETH1_INT		RXD2		
F13		PA5		ETH0_INT		TXD2		
F14		PA6				CTS2#	IRQ6	
G1		PC4		CATI2CCLK				
G2		PC5		CATI2CDATA				
G3	TCK							
G4	TMS							
G5	PLLVSS1							
G6	VSS							
G7	VSS							
G8	VSS							
G9	VSS							
G10	VDD							
G11		PA3				SCK2		
G12	TRACEDATA5	P77				RSPCK0		
G13		PA2				SSL02		
G14	TRACEDATA7	PA1				MISO0		
H1	VCCQ33							
H2	BSCANP							
H3		PU7		CATIRQ				
H4								IC0
H5	PLLVDD0							
H6	VSS							
H7	VSS							
H8	VSS							
H9	VSS							
H10	VDD							
H11	TRACEDATA2	P74				CTS1# / SSL03		
H12	TRACEDATA3	P75				SSL00	IRQ13	
H13	TRACEDATA4	P76				SSL01		
H14	TRACEDATA6	PA0				MOSI0		
J1	EXTAL							
J2	VSS							
J3		PM1		CATLEDERR				
J4	RES#							
J5	PLLVSS0							
J6	VDD							
J7	VSS							
J8	VSS							
J9	VSS							

Table 1.6 List of Pin and Pin Functions (196-Pin FBGA) (4 / 5)

Pin Number	Power Supply Clock System Control	I/O Port	Timer (CMTW)	Communication (ESC)	Communication (SPIBSC)	Communication (SCIFA, RSPIa, RIICa, RSCAN, USB)	Interrupt	Others (Connect to VSS via a resistor)
J10	VDD							
J11	TRACEDATA7	PE7				SCK1 / RSPCK0		
J12	TRACECTL	P71	TOC2			SCK1		
J13	TRACEDATA0	P72	TIC2			TXD1		
J14	TRACEDATA1	P73				RXD1	IRQ3	
K1	XTAL							
K2	VSS							
K3		PM4		CATLEDRUN				
K4	RSTOUT#							
K5	VDD33_USB							
K6	VDD							
K7	VDD							
K8	VDD							
K9	VDD							
K10	VCCQ33							
K11	TRACEDATA1	PE1				SSL03		
K12	TRACEDATA5	PE5				TXD1 / MOSI0		
K13	TRACEDATA6	PE6				RXD1 / MISO0	IRQ6	
K14	TRACECLK	P70				RTS1# / USB_OVRCUR	IRQ0	
L1	VSS							
L2		PM3		CATSYNC0 / CATLATCH0				
L3	USB_RREF							
L4		P62			SPBCLK			
L5	VCCQ33							
L6		PG2	TOC0			RSPCK1		
L7	VDD							
L8		PR1				CTS1#	IRQ9	
L9		P27				RTS0#		
L10	VDD							
L11	VCCQ33							
L12		PS6				RXD2	IRQ14	
L13	TRACEDATA3	PE3				CTS1# / SSL01	IRQ3	
L14	TRACEDATA4	PE4				RTS1# / SSL00		
M1		PM2		CATSYNC1 / CATLATCH1				
M2		PM6		CATLINKACT0			IRQ6	

Table 1.6 List of Pin and Pin Functions (196-Pin FBGA) (5 / 5)

Pin Number	Power Supply Clock System Control	I/O Port	Timer (CMTW)	Communication (ESC)	Communication (SPIBSC)	Communication (SCIFA, RSPIa, RIICa, RSCAN, USB)	Interrupt	Others (Connect to VSS via a resistor)
M3	VSS_USB							
M4		P60			SPBSSL			
M5		P64			SPBMI/SPBIO1			
M6		PG3	TIC1			MISO1		
M7		PG5				SSL10		
M8		P21				CTS0#	IRQ1	
M9		P26						
M10		P44				CTS0#	IRQ12	
M11		PS0						
M12	TRACECTL	P00						
M13		PS7				TXD2		
M14	TRACEDATA2	PE2				SSL02	IRQ2	
N1		PM7		CATLINKACT1				
N2		PM5		CATLEDSTER				
N3	DVDD_USB							
N4		P61		CTXD1	SPBIO3			
N5		P63			SPBMO/SPBIO0			
N6		PG4	TOC1			MOSI1		
N7		PG6				SSL11		
N8		P22				SCK0	IRQ2	
N9		P20						
N10		P42				RXD0		
N11		PS1					IRQ1	
N12		PS3						
N13		PS5						
N14	TRACEDATA0	PE0						
P1	VSS_USB							
P2	USB_DM							
P3	USB_DP							
P4		P66		CTXD1		USB_VBUSEN	IRQ14	
P5		P65			SPBIO2			
P6		PC6		USB_VBUSIN		SCL1		
P7		PC7	TIC0			SDA1 / CRXD1		
P8		P23				TXD0		
P9		P25						
P10		P40				TXD0		
P11	TRACECLK	P10					IRQ0	
P12		PS2						
P13		PS4						
P14	VSS							

2. CPU

These LSI products include a Cortex-R4 CPU. The revision of the module is r1p4.

2.1 Overview

Table 2.1 Specifications of CPU

Item		Specification
Cortex-R4 (r1p4)	Minimum instruction execution time	One clock per instruction
	Address space	4 Gbytes
	Instruction cache size	8 Kbytes (with ECC)
	Data cache size	8 Kbytes (with ECC)
	Tightly coupled memory (TCM) size	ATCM: 512 Kbytes (with ECC) BTCM: 32 Kbytes (with ECC)
	Instruction set	Arm v7-R architecture supporting Thumb®/Thumb-2
	Data arrangement	Instruction: Little endian Data: Little endian
	Memory protection	Memory protection unit (MPU)
	FPU	• Supports addition, subtraction, multiplication, division, product-sum operation, and square-root operation in single-precision and double-precision • 32-bit single word register: 32 registers Can also be used as 16 64-bit double word registers

For details, refer to the following documents supplied by Arm.

- Arm Architecture Reference Manual Arm v7-A and Arm v7-R edition Issue C
- Arm v7-M Architecture Reference Manual

2.2 Configuration Information

Table 2.2 lists the configuration information for the Cortex-R4 of this LSI.

Table 2.2 Setting Values for Cortex-R4 Configuration Signals

Item		Setting Value
Endian	CFGEE	0
	CFGIE	0
Interrupt	CFGNMFI	1
Exception vector	TEINIT	0
	VINITHI	1
TCM configuration	INITRAMA	1
	INITRAMB	1
	LOCZRAMA	1
	CFGATCMSZ[3:0]	Ah
	CFGBTCMSZ[3:0]	6h
	ENTCM1IF	0
	SLBTCMSB	1 (don't care)
ECC, etc.	PARECCENRAM[2:0]	000b
	ERRENRAM[2:0]	000b
	RMWENRAM[1:0]	00b
	PARLVRAM	0 (don't care)

2.3 Restrictions on CPU

For details on restrictions on Cortex-R4 mounted on this LSI, refer to the information provided at the website of Arm.

2.4 Register Descriptions

2.4.1 ATCM Wait Control Register (SYTATCMWAIT)

SYTATCMWAIT is a register that controls ATCM access wait.
This register can be protected by the register write protection function. When writing to this register, cancel the write protection of bit 3 in the protection register (PRCR). For details, see section 11, Register Write Protection Function.

Address(es): A00B 0800h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ATCMWAIT[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	ATCMWAIT[1:0]	ATCM Wait Setting*1	b1 b0 0 0: Setting prohibited 0 1: 1-wait 1 0: 0-wait 1 1: Setting prohibited	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. If the ATCMWAIT[1:0] setting changes, operation cannot be guaranteed when the bus maser such as CPU accesses ACTM (including instruction fetch). To prevent fetch access from CPU, these bits should be handled by programs allocated in memory areas other than ATCM.

ATCMWAIT[1:0] Bits (ATCM Wait Setting)

These bits specify the number of memory access waits for ATCM.
Setting these bits to 0-wait enables high-speed access without any wait time.

3. Operating Modes

3.1 Overview

This LSI chip is intended for booting up from an external serial flash memory. The SPI boot mode is available for the serial flash memory that supports the operating mode. In SPI boot mode, the user program stored in the corresponding external serial flash memory is booted up and then runs.

3.2 Type of Operating Mode

One operating mode is available with the method of connecting to an external serial flash memory. The type of the operating mode is given in Table 3.1. For details, see section 3.4, Operating Mode Descriptions.

Table 3.1 Type of Operating Mode

Operating Mode	Description
SPI boot mode (Serial flash)	Boots a program from a serial flash memory connected to the SPI multi-I/O bus space.

3.3 Hardware Used in SPI Boot Mode

Table 3.2 describes hardware used in SPI boot mode.
“Required Pins” indicates pins required to this mode. The functions for these pins are automatically configured at boot.

Table 3.2 Hardware Used in SPI Boot Mode

Operating Mode	Peripheral Module	Required Pins
SPI boot mode (Serial flash)	SPI Multi-I/O bus controller (SPIBSC)	SPBCLK, SPBSSL SPBMO, SPBMI

3.4 Operating Mode Descriptions

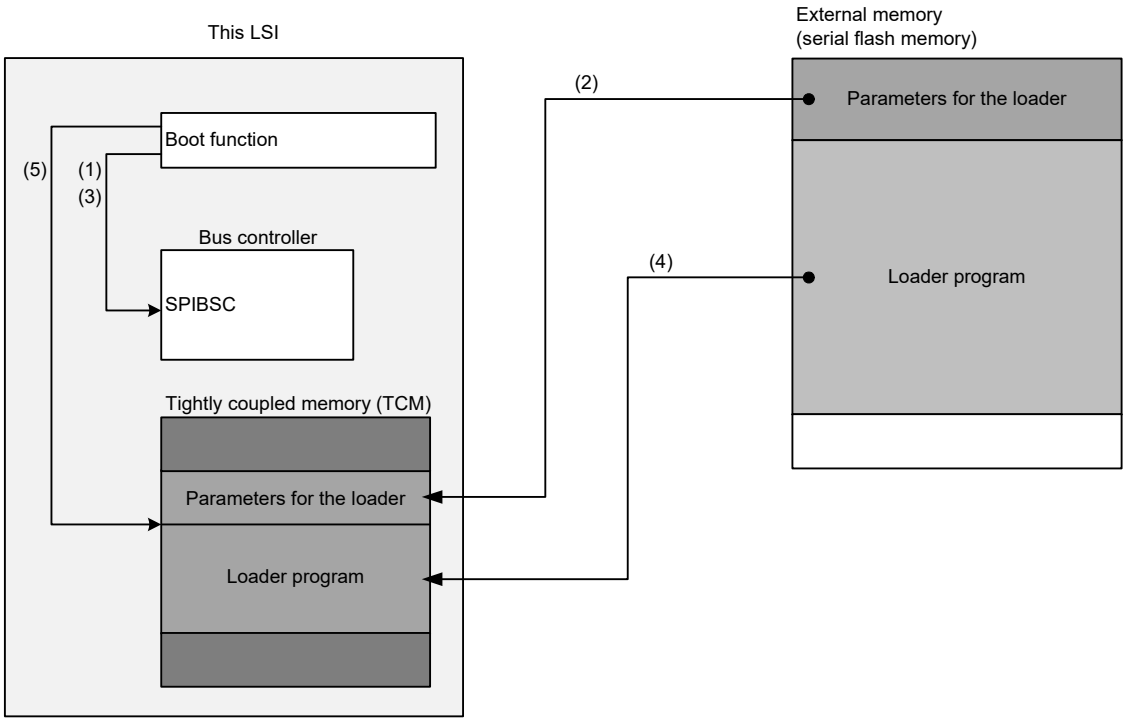
3.4.1 Boot Function

After reset is released on this LSI, the boot function executes the boot processing described below. The boot processing can extract a loader program that was stored in an external memory in advance by a user, to the internal tightly coupled memory (TCM) area, and hand over the processing to the loader program at the start address of that program.

(1) Setting the bus controller (SPIBSC)
(2) Loading parameters for the loader from an external memory, and executing checksum
(3) Setting for speeding up the bus controller (SPIBSC) by using parameters for the loader
(4) Loading the loader program from an external memory
(5) Branching off to the start address of the loader program extracted to the tightly coupled memory (TCM)

Parameters for the loader can have configuration information that suite for the user system, such as, loader program, cache setting for speeding up the boot processing, and bus controller (SPIBSC) settings. Parameters for the loader must be stored in an external memory in advance by a user.

Figure 3.1 shows the operating overview of boot processing.



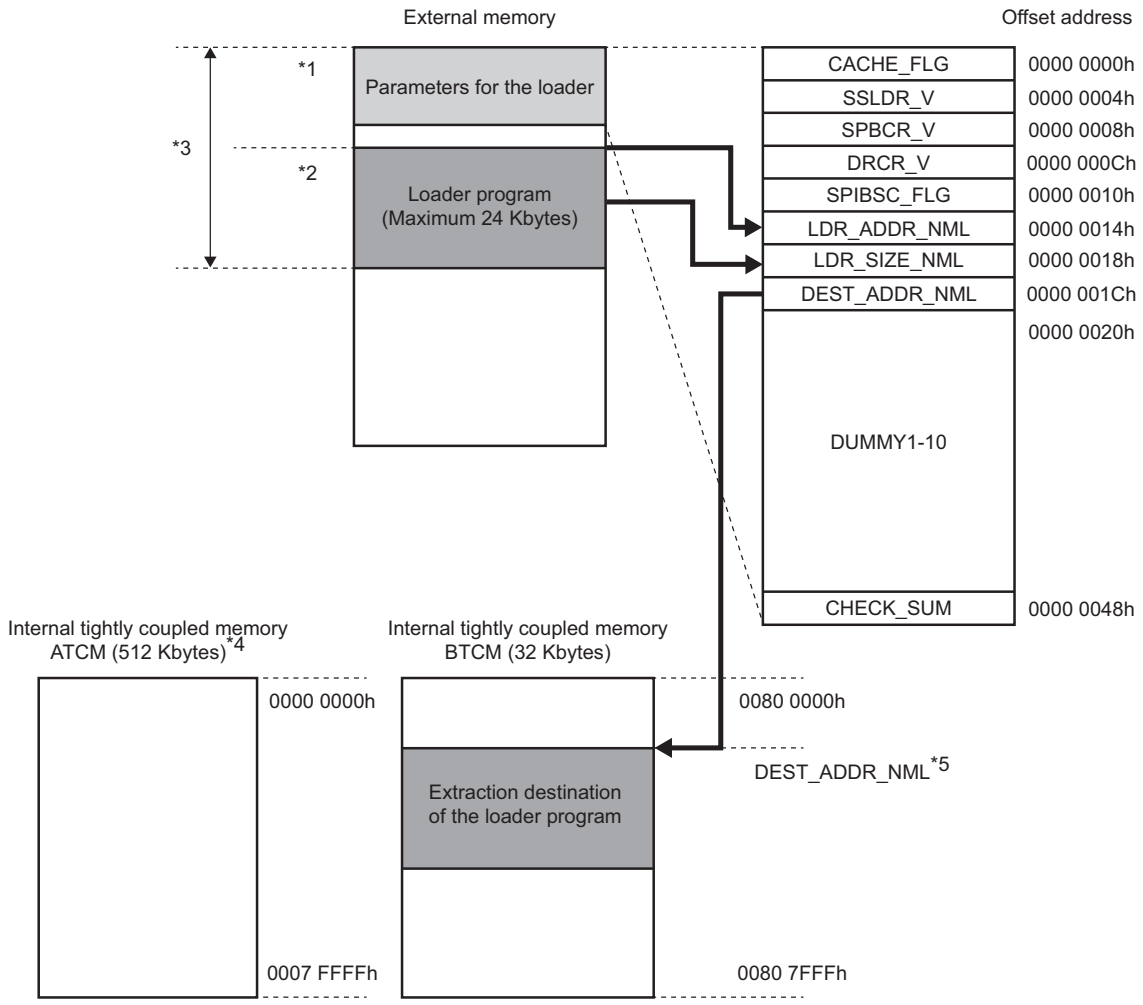
(1) Set the bus controller specified by the operating mode.
(2) Transfer the parameters for the loader from an external memory.
(3) Configure the bus controller based on the parameters for the loader.
(4) Transfer the loader program from an external memory.
(5) Hand over the processing to the loader program in the TCM area.

Figure 3.1 Operating Overview of Boot Processing

3.4.2 Parameters for the Loader

The parameters for the loader are setting parameters for boot processing, which are loaded from an external memory during boot processing and used by the boot function. The parameters for the loader specify information, such as the cache settings during boot processing in individual operating modes, setting of the bus controller (SPIBSC) used for communication with an external memory, and the size of the loader program.

Figure 3.2 shows memory assignment of the loader program and parameters for the loader.



Note 1. The parameters for the loader must be stored in the following address range:
SPI boot mode: SPI multi-I/O bus space (3000 0000h and higher)
Note 2. The loader program can be stored in any address (LDR_ADDR_NML). Note that, however, the program size (LDR_SIZE_NML) must be multiples of 512 bytes and no more than 24 Kbytes.
Note 3. In SPI boot mode, address output is 3 bytes, so only use addresses in the range from 3000 0000h to 30FF FFFFh.
Note 4. The ATCM area is assumed to be used for storing user application programs.
Note 5. Specify the extraction destination area of the loader program so that the area falls in the range from 0080 2000h to 0080 7FFFh.

Figure 3.2 Memory Assignment of the Loader Program and Parameters for the Loader

Table 3.3 describes parameter information for the loader in SPI boot mode.

Table 3.3 Parameter Information for the Loader in SPI Boot Mode

Offset Address	Parameter Name	Description
0000 0000h	CACHE_FLG	Selects whether to enable the I1 cache and D1 cache of Cortex-R4 at boot processing (for speeding up). 0000 0001h: Enables the I1 and D1 caches. Other setting values than above: Disables the I1 and D1 caches.
0000 0004h	SSLDR_V	Setting value of the SSL delay register (SSLDR) This parameter value is set to the SSLDR register during the setting for speeding up of SPIBSC in (3) of section 3.4.1, Boot Function.*1
0000 0008h	SPBCR_V	Setting value of the bit-rate configuration register (SPBCR) This parameter value is set to the SPBCR register during the setting for speeding up of SPIBSC in (3) of section 3.4.1, Boot Function.*1
0000 000Ch	DRCR_V	Setting value of the data read control register (DRCR) This parameter value is set to the DRCR register during the setting for speeding up of SPIBSC in (3) of section 3.4.1, Boot Function.*1
0000 0010h	SPIBSC_FLG	Selects whether to change the SPIBSC setting back to the initial value after the boot processing finishes. 2236 0679h: Changes the SPIBSC setting value back to the initial value after the boot processing finishes. Other setting values than above: Retains the SPIBCS setting value used during boot processing*2.
0000 0014h	LDR_ADDR_NML	Sets the start address of the loader program stored in the external memory.*3
0000 0018h	LDR_SIZE_NML	Specifies the size of the loader program. Note that the program size must be multiples of 512 bytes and no more than 24 Kbytes.*3
0000 000Ch	DEST_ADDR_NML	Specifies the start address of the tightly coupled memory (BTCM) that is used as the extraction destination of the loader program. Specify the extraction destination area of the loader program so that the area falls in the range from 0080 2000h to 0080 7FFFh.
0000 0020h	DUMMY1	Option (Not used in this mode.)
0000 0024h	DUMMY2	Option (Not used in this mode.)
0000 0028h	DUMMY3	Option (Not used in this mode.)
0000 002Ch	DUMMY4	Option (Not used in this mode.)
0000 0030h	DUMMY5	Option (Not used in this mode.)
0000 0034h	DUMMY6	Option (Not used in this mode.)
0000 0038h	DUMMY7	Option (Not used in this mode.)
0000 003Ch	DUMMY8	Option (Not used in this mode.)
0000 0040h	DUMMY9	Option (Not used in this mode.)
0000 0044h	DUMMY10	Option (Not used in this mode.)
0000 0048h	CHECK_SUM	Checksum value of the parameters for the loader This parameter specifies the sum of the higher-order 16 bits and the lower-order 16 bits of the parameters (in unsigned long (32-bit) format) in the range of the offset addresses 0000h to 0044h*4.

Note 1. For details about the SSLDR, SPBCR, and DRCR registers, see section 29, SPI Multi I/O Bus Controller (SPIBSC).
Note 2. For details about the setting status of the individual peripheral modules after the boot processing finishes, see section 3.4.4.1, Operation Settings in SPI Boot Mode.
Note 3. LDR_ADDR_NML must be in the range from 3000 004Ch and LDR_ADDR_NML + LDR_SIZE_NML ≤ 3100 0000h in the external address space (SPI).
Note 4. An example for calculating CHECK_SUM is given below.
If SSLDR_V = 0007 0707h,
SPBCR_V = 0000 0003h,
LDR_ADDR_NML = 3000 004Ch,
LDR_SIZE_NML = 0000 6000h,
DEST_ADDR_NML = 0080 2000h, and
others = 0000 0000h,
CHECK_SUM is calculated as below ((0000h) is omitted in the formula):
CHECK_SUM = (0007h)+(0707h)+(0003h)+(3000h)+(004Ch)+(6000h)+(0080h)+(2000h) = (0000 B7DDh)

3.4.3 Loader Program

The loader program is a user program that is transferred from an external memory to the internal tightly coupled memory (TCM) by the boot function, and starts its processing after the boot processing finishes. The loader program can execute such processing that suites the user system, for example, extracting a user application program from an external memory to the internal TCM area and executing it at high speed.
Set the loader program so that the following conditions are satisfied:

- Program size (LDR_SIZE_NML): Multiples of 512 bytes and no more than 24 Kbytes
- Storage address in the external memory (LDR_ADDR_NML) in SPI boot mode: Address range from 3000 004Ch and LDR_ADDR_NML + LDR_SIZE_NML ≤ 3100 0000h

These setting values must be stored in an external memory as parameters for the loader. For details, see section 3.4.2, Parameters for the Loader.

3.4.4 SPI Boot Mode (Serial Flash)

In SPI boot mode, this LSI boots a program from an external serial flash memory connected to the SPI multi-I/O bus space.
In this mode, the SPI multi-I/O bus controller is set to the mode of reading the external address space, and the SPBCLK, SPBSSL, SPBMO, and SPBMI pins are enabled.
After the reset is released, this LSI executes the boot processing. The loader program stored in a serial flash memory connected to the SPI multi-I/O bus space is extracted to the internal memory (TCM), and then the processing is executed.
Figure 3.3 shows the connection diagram of this LSI with a serial flash memory.

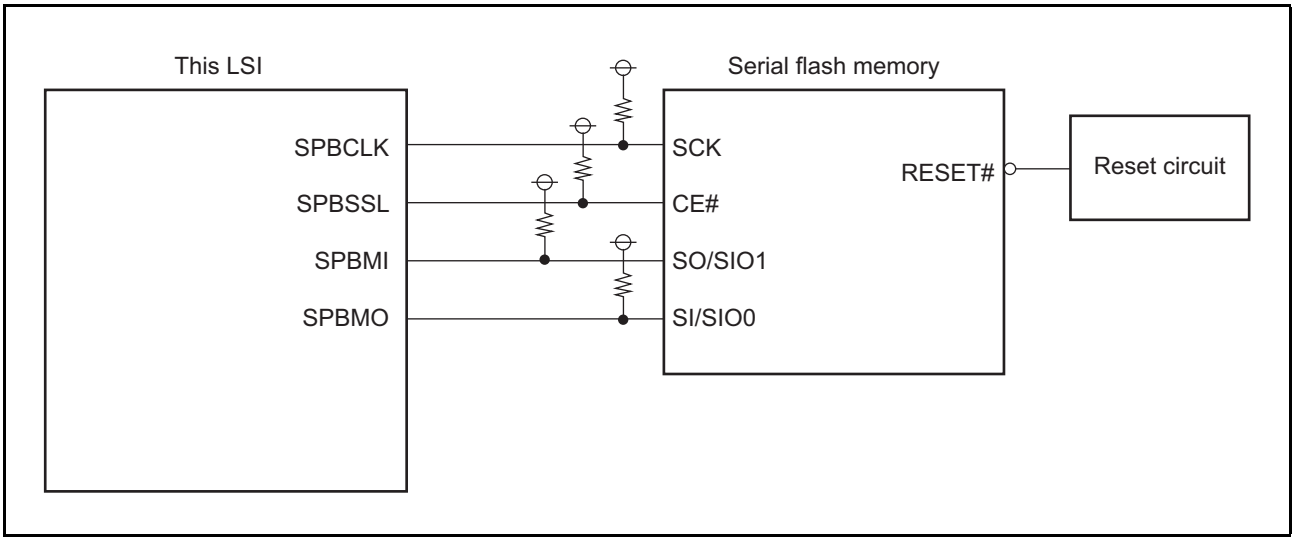


Figure 3.3 Connection Diagram of This LSI with a Serial Flash Memory

3.4.4.1 Operation Settings in SPI Boot Mode

Immediately after the boot processing starts in SPI boot mode after the reset is released, this LSI operates with the following initial setting values, and executes processing until transferring parameters for the loader.

- CPU clock (CPUCLK): 150 MHz
- SPIBSC bit rate (SPBCLK): 18.75 MHz
- Applicable command: Read (03h)
- Address output : 3 bytes
- Dummy cycles: None
- Data read width: 1 bit
- SPI mode: CPOL = 0 (positive pulse)
  CPHAR = 0 (data reception at odd edge)
  CPHAT = 0 (data transmission at even edge)

After the parameters for the loader are loaded, the settings for the I1 and D1 caches of Cortex-R4 and for the SSLDR, SPBCR, and DRCR registers are performed based on the values of parameters CACHE_FLG, SSLDR_V, SPBCR_V, and DRCR_V, so that the processing can be speed up.

Table 3.4 describes the setting values of the individual peripheral modules and registers at the time SPI boot mode finishes.

Also, Table 3.5 describes the setting values of Arm general-purpose registers at the time the boot processing finishes, and Table 3.6 describes the status of the Arm CP15 registers at the time the boot finishes.

Table 3.4 Setting Values of the Individual Peripheral Modules and Registers at the Time SPI Boot Mode Finishes

		Setting Value at the Time the Boot Processing Finishes	
Peripheral Module	**Register**	**When SPIBSC is initialized (SPIBSC_FLG = 2236 0679h)**	**When SPIBSC is not initialized (SPIBSC_FLG ≠ 2236 0679h)**
Low power consumption	MSTPCRC	0000 7DFEh (Initial value)	0000 7DFEh
SPIBSC	SSLDR	0007 0707h (Initial value)	Setting value of SSLDR_V
	SPBCR	0000 0003h (Initial value)	Setting value of SPBCR_V
	DRCR	0000 0000h (Initial value)	Setting value of DRCR_V
I/O ports	PORT6 .PMR	1Dh*1	1Dh*1
	MPC.PmnPFS	1Bh*1	1Bh*1

Note 1. Bits corresponding to the SPBCLK, SPBSSL, SPBMI, and SPBMO pins

Table 3.5 Setting Values of the Arm General-Purpose Registers at the Time the Boot Processing Finishes

No.	Register Name	Setting Values for Individual Processor Modes					
		User Mode/Current Mode	IRQ	FIQ	Undef	Abort	SVC
1	R0	Undefined	—	—	—	—	—
2	R1	Undefined	—	—	—	—	—
3	R2	Undefined	—	—	—	—	—
4	R3	Undefined	—	—	—	—	—
5	R4	Undefined	—	—	—	—	—
6	R5	Undefined	—	—	—	—	—
7	R6	Undefined	—	—	—	—	—
8	R7	Undefined	—	—	—	—	—
9	R8	Undefined	—	Undefined	—	—	—
10	R9	Undefined	—	Undefined	—	—	—
11	R10	Undefined	—	Undefined	—	—	—
12	R11	Undefined	—	Undefined	—	—	—
13	R12	Undefined	—	Undefined	—	—	—
14	R13(sp)	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
15	R14(lr)	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
16	R15(pc)	Arbitrary	—	—	—	—	—
17	cpsr	xxxx xx93h ([31:8] is undefined.) [7]I = 1 [6]F = 0*1 [5]T = 0 [4:0]MD = 10011b(SVC)	—	—	—	—	—
18	spsr	—	Undefined	Undefined	Undefined	Undefined	Undefined

—: Non-existent register

sp: Stack pointer

lr: Link register (which stores the return address when calling a subroutine call)

pc: Program counter

cpsr: Abbreviation of “current program status register”. It monitors or controls internal operations.

spsr: Abbreviation of “saved program status register”. It saves cpsr in the previous mode.

Note 1. In this product, the non-maskable interrupt has been assigned to FIQ. The boot processing changes the [6]F bit of the CPSR register from 1 to 0 to enable non-maskable interrupt after the boot processing finishes.

Note: After the boot processing sets the [6]F bit in the CPSR register to 0, if a non-maskable interrupt (FIQ exception) occurs before the processing reaches the branch to the loader program, the processing is jumped to the FIQ exception handler address and goes into an infinite loop. For details, see section 3.4.7, Note.

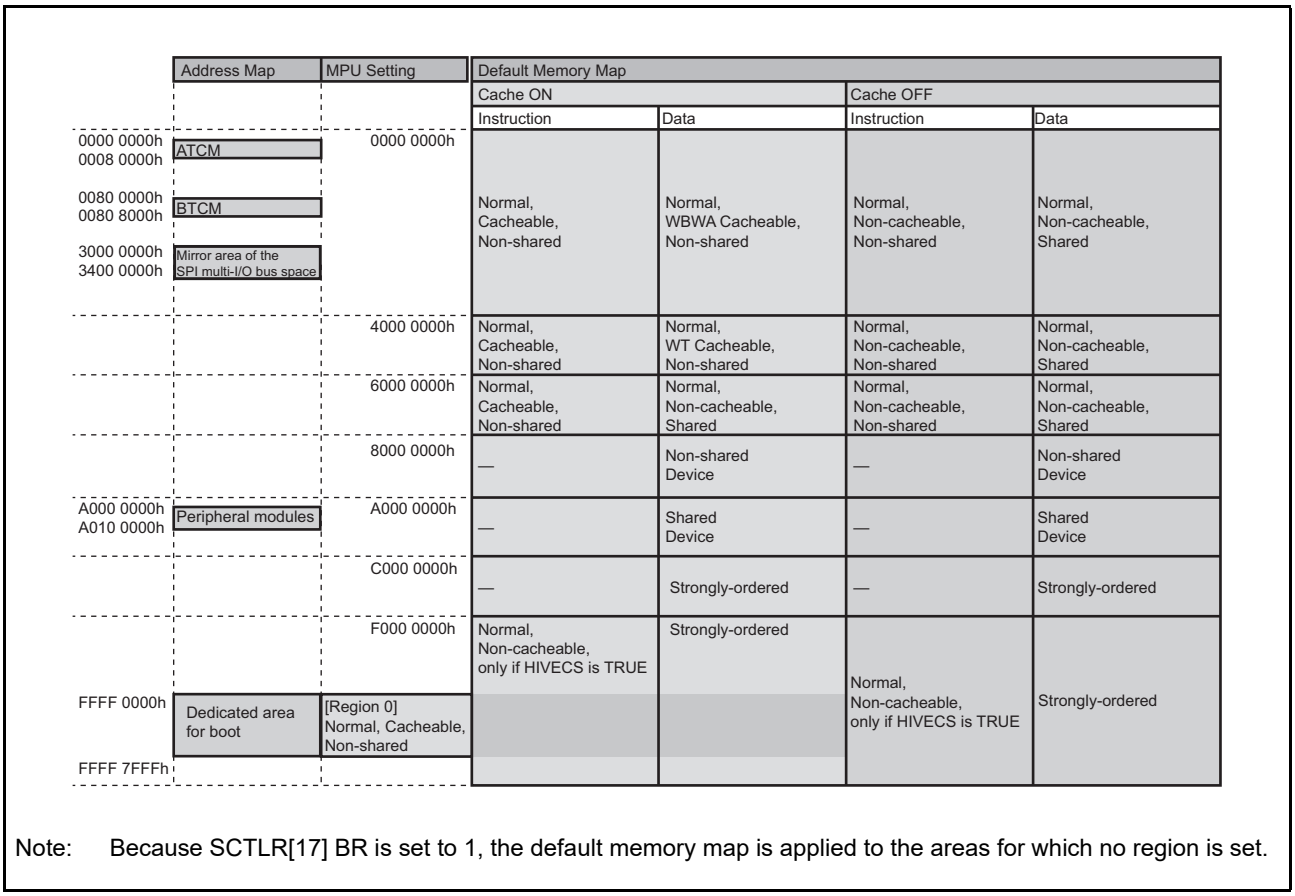
Table 3.6 Status of the Arm CP15 Registers at the Time the Boot Finishes

Register Name	Symbol	Setting Value at the Time the Boot Processing Finishes	Remarks
System control register	SCTLR	09E5 2878h*1	[24]VE = 1: Sets the IRQ exception vector address in VIC.
System control auxiliary register	ACTLR	0E00 0020h (When ATCM and BTCM are used)	ECC enable for TCM All areas of ATCM and BTCM are written and processed in 32-bit units during boot processing and initialized.
Invalidate all Instruction Caches Register	—	—	The I1 cache entry is not invalidated after the boot processing finishes.
Invalidate all Data Caches Register	—	—	The D1 cache entry is not invalidated after the boot processing finishes.
MPU Memory Region Number Register	RGNR	0000 0000h	All MPU settings are initialized even when the cache is enabled by the parameters for the loader.
Data Region Base Address Register	DRBAR	0000 0000h	
Data Region Size and Enable Register	DRSR	0000 0000h	
Data Region Access Control Register	DRACR	0000 0000h	

Note 1. When the boot processing finishes, the register is in high vector status, where V[13] = 1 (FFFF 0000h). Use the loader program to write appropriate processing in the low vectors (0000 0000h), and then to change the register to low vector status, where V[13] = 0 (0000 0000h).

3.4.5 MPU Setting

The boot function uses the primary instruction cache (I1) and primary data cache (D1) of Cortex-R4 when the parameter CACHE_FLG for the loader is set to 0000 0001h.
However, the dedicated area for boot processing (FFFF 0000h to FFFF 7FFFh), which is used by the boot function, is set as the non-cache area in the default map of Cortex-R4, so the MPU (memory protection unit) redefines the cache area during the boot processing.
The boot function defines the high-vector area (FFFF 0000h to FFFF 7FFFh) as the cache area of Region0, and uses other areas for the default memory map.
When the boot processing finishes, the I1 and D1 caches are invalidated, and all areas are initialized to the default memory map.
Figure 3.4 shows the relationship between the memory map definition during the boot processing and the default memory map of Cortex-R4.



Note: Because SCTLR[17] BR is set to 1, the default memory map is applied to the areas for which no region is set.

Figure 3.4 Relationship Between the Memory Map Definition During the Boot Processing and the Default Memory Map of Cortex-R4

3.4.6 Boot-Related Information and Error Processing

The boot function determines whether the boot processing finishes normally, and retains the result in a specific address. If the processing is determined to be an error, the boot processing is aborted, and an infinite loop is executed.
If the debugger is connected, reading the result of the boot processing that was stored in a specific address at the point of break can determine the error source.
Table 3.7 describes the error sources and the results of the boot processing.

Table 3.7 Error Sources and Results of Boot Processing

Storage Address*1	Stored Value*1	Error Sources and Results of Boot Processing
0080 09C4h	0	The boot processing finished normally.
	-2	Checksum error of the parameters for the loader When the checksum (CHECK_SUM) of the parameters for the loader does not match
	-3	Error in a parameter for the loader When one of the following is satisfied: - The size of the loader program is smaller than 512 bytes. - The size of the loader program is larger than 24 Kbytes. - The size of the loader program is not a multiple of 512 bytes. - The destination address of the loader program is outside of the TCM area.

Note 1. The access size is 32 bits.

3.4.7 Note

3.4.7.1 Exception Processing

Only the reset exception due to the RES#-pin reset can be accepted during the boot processing. When a reset exception occurs, this LSI is reset, and the boot processing restarts. If an exception processing other than reset exception occurs, the jump instruction to the relevant exception handler address repeats an infinite loop.

Table 3.8 Exception Processing During the Boot Processing

Exception	Handler Address	Operation During the Boot Processing
Reset exception	FFFF 0000h	Branched to the reset exception handler
Undefined instruction exception	FFFF 0004h	Branched to the undefined instruction exception handler (Infinite loop)
Software interrupt exception	FFFF 0008h	Branched to the software interrupt exception handler (Infinite loop)
Prefetch abort exception	FFFF 000Ch	Branched to the prefetch abort exception handler (Infinite loop)
Data abort exception	FFFF 0010h	Branched to the data abort exception handler (Infinite loop)
IRQ exception	FFFF 0018h	Branched to the IRQ exception handler (Infinite loop)
FIQ exception	FFFF 001Ch	Branched to the FIQ exception handler (Infinite loop)

Note: Before the boot processing finishes, the register is in high vector status, where SCTLR V[13] = 1 (FFFF 0000h). Use the loader program to write appropriate processing in the low vectors (0000 0000h), and then to change the register to low vector status, where V[13] = 0 (0000 0000h).

3.4.7.2 Serial Flash Memory in SPI Boot Mode

In SPI boot mode, after release from the reset state, boot processing starts by reading from the serial flash memory via the SPI multi-I/O bus controller (SPIBSC) with the initial settings given in section 3.4.4.1, Operation Settings in SPI Boot Mode.

The setting of serial flash memory can be changed via the SPIBSC after the processing to boot up is finished. Depending on the settings, however, reading from the serial flash memory may not be possible when boot processing needs to be started again following a reset. Therefore, caution is required on this point.

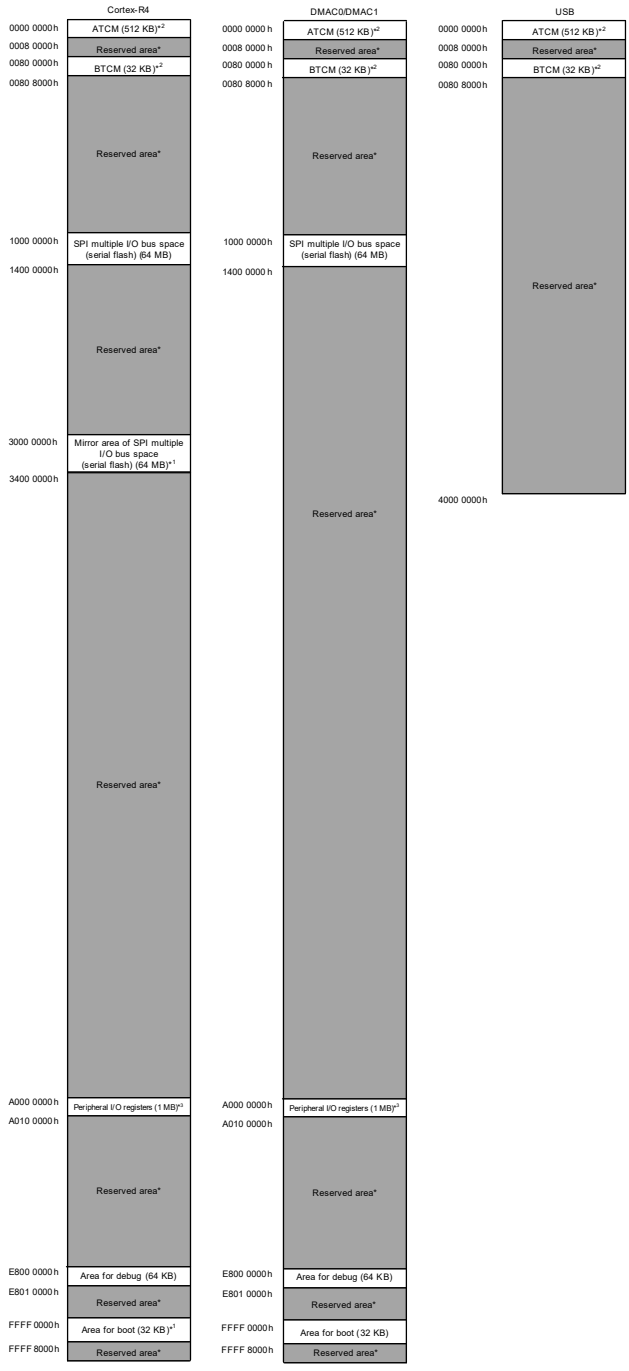
When the active level of the signal on the RES#-pin is applied to reset this LSI chip, the serial flash memory can be simultaneously initialized by input of the same reset signal to the reset pin of the serial flash memory. Therefore, we recommend using serial flash memory that includes a reset pin. As the reset signal of a serial flash memory in a small package may be multiplexed with another pin function, make sure that the reset function is selected.

In addition, when an internal reset such as a software reset or ECM reset is to be generated, initialize the serial flash memory by software in advance so that it can be connected in boot processing.

4. Address Space

4.1 Address Space

This LSI has a 4-Gbyte address space ranging from 0000 0000h to FFFF FFFFh. That is, up to total 4 Gbytes of program and data areas can be accessed linearly.
Figure 4.1 shows the memory maps for respective products.
Accessible areas will differ depending on the operating mode and the states of control bits.
In addition, since a non-cached access area from each bus master is assigned to the same area in this product, in access from the Cortex-R4, a mirror area is set as a cache-enabled area. The MPU should not enable caching of areas other than the mirror area.



Note 1. Can be cached.
(For other areas, caching should not be enabled by the MPU.)
Note 2. Cortex-R4 tightly coupled memory space
Note 3. Peripheral space

Note: Reserved areas should not be accessed.

Figure 4.1 Memory Map

5. I/O Registers

This section gives information on the on-chip I/O register addresses. The information is given as shown below.

(1) I/O register addresses (address order)

- Registers are listed from the lower allocation addresses.
- Registers are classified according to module symbols.
- Among the internal I/O register area, addresses not listed in the list of registers are reserved. Reserved addresses must not be accessed. Do not access these addresses; otherwise, the operation when accessing these bits and subsequent operations cannot be guaranteed.

5.1 I/O Register Addresses (Address Order)

Table 5.1 List of I/O Registers (Address Order) (1 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 0000h	PORT0	Port direction register	PDR	16	16
A000 0002h	PORT1	Port direction register	PDR	16	16
A000 0004h	PORT2	Port direction register	PDR	16	16
A000 0006h	PORT3	Port direction register	PDR	16	16
A000 0008h	PORT4	Port direction register	PDR	16	16
A000 000Ah	PORT5	Port direction register	PDR	16	16
A000 000Ch	PORT6	Port direction register	PDR	16	16
A000 000Eh	PORT7	Port direction register	PDR	16	16
A000 0010h	PORT8	Port direction register	PDR	16	16
A000 0012h	PORT9	Port direction register	PDR	16	16
A000 0014h	PORTA	Port direction register	PDR	16	16
A000 0016h	PORTB	Port direction register	PDR	16	16
A000 0018h	PORTC	Port direction register	PDR	16	16
A000 001Ah	PORTD	Port direction register	PDR	16	16
A000 001Ch	PORTE	Port direction register	PDR	16	16
A000 001Eh	PORTF	Port direction register	PDR	16	16
A000 0020h	PORTG	Port direction register	PDR	16	16
A000 0024h	PORTJ	Port direction register	PDR	16	16
A000 002Ah	PORTM	Port direction register	PDR	16	16
A000 0030h	PORTR	Port direction register	PDR	16	16
A000 0032h	PORTS	Port direction register	PDR	16	16
A000 0036h	PORTU	Port direction register	PDR	16	16
A000 0040h	PORT0	Port output data register	PODR	8	8
A000 0041h	PORT1	Port output data register	PODR	8	8
A000 0042h	PORT2	Port output data register	PODR	8	8
A000 0043h	PORT3	Port output data register	PODR	8	8
A000 0044h	PORT4	Port output data register	PODR	8	8
A000 0045h	PORT5	Port output data register	PODR	8	8
A000 0046h	PORT6	Port output data register	PODR	8	8
A000 0047h	PORT7	Port output data register	PODR	8	8
A000 0048h	PORT8	Port output data register	PODR	8	8
A000 0049h	PORT9	Port output data register	PODR	8	8
A000 004Ah	PORTA	Port output data register	PODR	8	8
A000 004Bh	PORTB	Port output data register	PODR	8	8
A000 004Ch	PORTC	Port output data register	PODR	8	8
A000 004Dh	PORTD	Port output data register	PODR	8	8
A000 004Eh	PORTE	Port output data register	PODR	8	8
A000 004Fh	PORTF	Port output data register	PODR	8	8
A000 0050h	PORTG	Port output data register	PODR	8	8
A000 0052h	PORTJ	Port output data register	PODR	8	8
A000 0055h	PORTM	Port output data register	PODR	8	8
A000 0058h	PORTR	Port output data register	PODR	8	8
A000 0059h	PORTS	Port output data register	PODR	8	8
A000 005Bh	PORTU	Port output data register	PODR	8	8
A000 0060h	PORT0	Port input data register	PIDR	8	8
A000 0061h	PORT1	Port input data register	PIDR	8	8

Table 5.1 List of I/O Registers (Address Order) (2 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 0062h	PORT2	Port input data register	PIDR	8	8
A000 0063h	PORT3	Port input data register	PIDR	8	8
A000 0064h	PORT4	Port input data register	PIDR	8	8
A000 0065h	PORT5	Port input data register	PIDR	8	8
A000 0066h	PORT6	Port input data register	PIDR	8	8
A000 0067h	PORT7	Port input data register	PIDR	8	8
A000 0068h	PORT8	Port input data register	PIDR	8	8
A000 0069h	PORT9	Port input data register	PIDR	8	8
A000 006Ah	PORTA	Port input data register	PIDR	8	8
A000 006Bh	PORTB	Port input data register	PIDR	8	8
A000 006Ch	PORTC	Port input data register	PIDR	8	8
A000 006Dh	PORTD	Port input data register	PIDR	8	8
A000 006Eh	PORTE	Port input data register	PIDR	8	8
A000 006Fh	PORTF	Port input data register	PIDR	8	8
A000 0070h	PORTG	Port input data register	PIDR	8	8
A000 0072h	PORTJ	Port input data register	PIDR	8	8
A000 0075h	PORTM	Port input data register	PIDR	8	8
A000 0078h	PORTR	Port input data register	PIDR	8	8
A000 0079h	PORTS	Port input data register	PIDR	8	8
A000 007Bh	PORTU	Port input data register	PIDR	8	8
A000 0080h	PORT0	Port mode register	PMR	8	8
A000 0081h	PORT1	Port mode register	PMR	8	8
A000 0082h	PORT2	Port mode register	PMR	8	8
A000 0083h	PORT3	Port mode register	PMR	8	8
A000 0084h	PORT4	Port mode register	PMR	8	8
A000 0085h	PORT5	Port mode register	PMR	8	8
A000 0086h	PORT6	Port mode register	PMR	8	8
A000 0087h	PORT7	Port mode register	PMR	8	8
A000 0088h	PORT8	Port mode register	PMR	8	8
A000 0089h	PORT9	Port mode register	PMR	8	8
A000 008Ah	PORTA	Port mode register	PMR	8	8
A000 008Bh	PORTB	Port mode register	PMR	8	8
A000 008Ch	PORTC	Port mode register	PMR	8	8
A000 008Dh	PORTD	Port mode register	PMR	8	8
A000 008Eh	PORTE	Port mode register	PMR	8	8
A000 008Fh	PORTF	Port mode register	PMR	8	8
A000 0090h	PORTG	Port mode register	PMR	8	8
A000 0092h	PORTJ	Port mode register	PMR	8	8
A000 0095h	PORTM	Port mode register	PMR	8	8
A000 0098h	PORTR	Port mode register	PMR	8	8
A000 0099h	PORTS	Port mode register	PMR	8	8
A000 009Bh	PORTU	Port mode register	PMR	8	8
A000 0100h	PORT0	Pull-up/pull-down control register	PCR	16	16
A000 0102h	PORT1	Pull-up/pull-down control register	PCR	16	16
A000 0104h	PORT2	Pull-up/pull-down control register	PCR	16	16
A000 0106h	PORT3	Pull-up/pull-down control register	PCR	16	16
A000 0108h	PORT4	Pull-up/pull-down control register	PCR	16	16
A000 010Ah	PORT5	Pull-up/pull-down control register	PCR	16	16

Table 5.1 List of I/O Registers (Address Order) (3 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 010Ch	PORT6	Pull-up/pull-down control register	PCR	16	16
A000 010Eh	PORT7	Pull-up/pull-down control register	PCR	16	16
A000 0110h	PORT8	Pull-up/pull-down control register	PCR	16	16
A000 0112h	PORT9	Pull-up/pull-down control register	PCR	16	16
A000 0114h	PORTA	Pull-up/pull-down control register	PCR	16	16
A000 0116h	PORTB	Pull-up/pull-down control register	PCR	16	16
A000 011Ah	PORTD	Pull-up/pull-down control register	PCR	16	16
A000 011Ch	PORTE	Pull-up/pull-down control register	PCR	16	16
A000 011Eh	PORTF	Pull-up/pull-down control register	PCR	16	16
A000 0120h	PORTG	Pull-up/pull-down control register	PCR	16	16
A000 0124h	PORTJ	Pull-up/pull-down control register	PCR	16	16
A000 012Ah	PORTM	Pull-up/pull-down control register	PCR	16	16
A000 0130h	PORTR	Pull-up/pull-down control register	PCR	16	16
A000 0132h	PORTS	Pull-up/pull-down control register	PCR	16	16
A000 0136h	PORTU	Pull-up/pull-down control register	PCR	16	16
A000 0200h	MPC	Port 00 pin function control register	P00PFS	8	8
A000 0208h	MPC	Port 10 pin function control register	P10PFS	8	8
A000 020Fh	MPC	Port 17 pin function control register	P17PFS	8	8
A000 0211h	MPC	Port 21 pin function control register	P21PFS	8	8
A000 0212h	MPC	Port 22 pin function control register	P22PFS	8	8
A000 0213h	MPC	Port 23 pin function control register	P23PFS	8	8
A000 0217h	MPC	Port 27 pin function control register	P27PFS	8	8
A000 021Bh	MPC	Port 33 pin function control register	P33PFS	8	8
A000 021Ch	MPC	Port 34 pin function control register	P34PFS	8	8
A000 021Dh	MPC	Port 35 pin function control register	P35PFS	8	8
A000 0220h	MPC	Port 40 pin function control register	P40PFS	8	8
A000 0222h	MPC	Port 42 pin function control register	P42PFS	8	8
A000 0224h	MPC	Port 44 pin function control register	P44PFS	8	8
A000 0228h	MPC	Port 50 pin function control register	P50PFS	8	8
A000 0229h	MPC	Port 51 pin function control register	P51PFS	8	8
A000 022Ah	MPC	Port 52 pin function control register	P52PFS	8	8
A000 022Bh	MPC	Port 53 pin function control register	P53PFS	8	8
A000 022Ch	MPC	Port 54 pin function control register	P54PFS	8	8
A000 0230h	MPC	Port 60 pin function control register	P60PFS	8	8
A000 0231h	MPC	Port 61 pin function control register	P61PFS	8	8
A000 0232h	MPC	Port 62 pin function control register	P62PFS	8	8
A000 0233h	MPC	Port 63 pin function control register	P63PFS	8	8
A000 0234h	MPC	Port 64 pin function control register	P64PFS	8	8
A000 0235h	MPC	Port 65 pin function control register	P65PFS	8	8
A000 0236h	MPC	Port 66 pin function control register	P66PFS	8	8
A000 0238h	MPC	Port 70 pin function control register	P70PFS	8	8
A000 0239h	MPC	Port 71 pin function control register	P71PFS	8	8
A000 023Ah	MPC	Port 72 pin function control register	P72PFS	8	8
A000 023Bh	MPC	Port 73 pin function control register	P73PFS	8	8
A000 023Ch	MPC	Port 74 pin function control register	P74PFS	8	8
A000 023Dh	MPC	Port 75 pin function control register	P75PFS	8	8
A000 023Eh	MPC	Port 76 pin function control register	P76PFS	8	8
A000 023Fh	MPC	Port 77 pin function control register	P77PFS	8	8

Table 5.1 List of I/O Registers (Address Order) (4 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 0240h	MPC	Port 80 pin function control register	P80PFS	8	8
A000 0241h	MPC	Port 81 pin function control register	P81PFS	8	8
A000 0242h	MPC	Port 82 pin function control register	P82PFS	8	8
A000 0243h	MPC	Port 83 pin function control register	P83PFS	8	8
A000 0244h	MPC	Port 84 pin function control register	P84PFS	8	8
A000 0245h	MPC	Port 85 pin function control register	P85PFS	8	8
A000 0246h	MPC	Port 86 pin function control register	P86PFS	8	8
A000 0247h	MPC	Port 87 pin function control register	P87PFS	8	8
A000 0248h	MPC	Port 90 pin function control register	P90PFS	8	8
A000 0249h	MPC	Port 91 pin function control register	P91PFS	8	8
A000 024Ah	MPC	Port 92 pin function control register	P92PFS	8	8
A000 024Bh	MPC	Port 93 pin function control register	P93PFS	8	8
A000 024Ch	MPC	Port 94 pin function control register	P94PFS	8	8
A000 024Dh	MPC	Port 95 pin function control register	P95PFS	8	8
A000 024Fh	MPC	Port 97 pin function control register	P97PFS	8	8
A000 0250h	MPC	Port A0 pin function control register	PA0PFS	8	8
A000 0251h	MPC	Port A1 pin function control register	PA1PFS	8	8
A000 0252h	MPC	Port A2 pin function control register	PA2PFS	8	8
A000 0253h	MPC	Port A3 pin function control register	PA3PFS	8	8
A000 0254h	MPC	Port A4 pin function control register	PA4PFS	8	8
A000 0255h	MPC	Port A5 pin function control register	PA5PFS	8	8
A000 0256h	MPC	Port A6 pin function control register	PA6PFS	8	8
A000 0257h	MPC	Port A7 pin function control register	PA7PFS	8	8
A000 0258h	MPC	Port B0 pin function control register	PB0PFS	8	8
A000 0259h	MPC	Port B1 pin function control register	PB1PFS	8	8
A000 025Ah	MPC	Port B2 pin function control register	PB2PFS	8	8
A000 025Bh	MPC	Port B3 pin function control register	PB3PFS	8	8
A000 025Ch	MPC	Port B4 pin function control register	PB4PFS	8	8
A000 025Dh	MPC	Port B5 pin function control register	PB5PFS	8	8
A000 025Eh	MPC	Port B6 pin function control register	PB6PFS	8	8
A000 025Fh	MPC	Port B7 pin function control register	PB7PFS	8	8
A000 0260h	MPC	Port C0 pin function control register	PC0PFS	8	8
A000 0261h	MPC	Port C1 pin function control register	PC1PFS	8	8
A000 0262h	MPC	Port C2 pin function control register	PC2PFS	8	8
A000 0263h	MPC	Port C3 pin function control register	PC3PFS	8	8
A000 0264h	MPC	Port C4 pin function control register	PC4PFS	8	8
A000 0265h	MPC	Port C5 pin function control register	PC5PFS	8	8
A000 0266h	MPC	Port C6 pin function control register	PC6PFS	8	8
A000 0267h	MPC	Port C7 pin function control register	PC7PFS	8	8
A000 026Dh	MPC	Port D5 pin function control register	PD5PFS	8	8
A000 026Eh	MPC	Port D6 pin function control register	PD6PFS	8	8
A000 026Fh	MPC	Port D7 pin function control register	PD7PFS	8	8
A000 0270h	MPC	Port E0 pin function control register	PE0PFS	8	8
A000 0271h	MPC	Port E1 pin function control register	PE1PFS	8	8
A000 0272h	MPC	Port E2 pin function control register	PE2PFS	8	8
A000 0273h	MPC	Port E3 pin function control register	PE3PFS	8	8
A000 0274h	MPC	Port E4 pin function control register	PE4PFS	8	8
A000 0275h	MPC	Port E5 pin function control register	PE5PFS	8	8

Table 5.1 List of I/O Registers (Address Order) (5 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 0276h	MPC	Port E6 pin function control register	PE6PFS	8	8
A000 0277h	MPC	Port E7 pin function control register	PE7PFS	8	8
A000 027Dh	MPC	Port F5 pin function control register	PF5PFS	8	8
A000 027Eh	MPC	Port F6 pin function control register	PF6PFS	8	8
A000 027Fh	MPC	Port F7 pin function control register	PF7PFS	8	8
A000 0282h	MPC	Port G2 pin function control register	PG2PFS	8	8
A000 0283h	MPC	Port G3 pin function control register	PG3PFS	8	8
A000 0284h	MPC	Port G4 pin function control register	PG4PFS	8	8
A000 0285h	MPC	Port G5 pin function control register	PG5PFS	8	8
A000 0286h	MPC	Port G6 pin function control register	PG6PFS	8	8
A000 0290h	MPC	Port J0 pin function control register	PJ0PFS	8	8
A000 0291h	MPC	Port J1 pin function control register	PJ1PFS	8	8
A000 0292h	MPC	Port J2 pin function control register	PJ2PFS	8	8
A000 0293h	MPC	Port J3 pin function control register	PJ3PFS	8	8
A000 0294h	MPC	Port J4 pin function control register	PJ4PFS	8	8
A000 0295h	MPC	Port J5 pin function control register	PJ5PFS	8	8
A000 0296h	MPC	Port J6 pin function control register	PJ6PFS	8	8
A000 0297h	MPC	Port J7 pin function control register	PJ7PFS	8	8
A000 02A9h	MPC	Port M1 pin function control register	PM1PFS	8	8
A000 02AAh	MPC	Port M2 pin function control register	PM2PFS	8	8
A000 02ABh	MPC	Port M3 pin function control register	PM3PFS	8	8
A000 02ACh	MPC	Port M4 pin function control register	PM4PFS	8	8
A000 02ADh	MPC	Port M5 pin function control register	PM5PFS	8	8
A000 02AEh	MPC	Port M6 pin function control register	PM6PFS	8	8
A000 02AFh	MPC	Port M7 pin function control register	PM7PFS	8	8
A000 02C1h	MPC	Port R1 pin function control register	PR1PFS	8	8
A000 02C9h	MPC	Port S1 pin function control register	PS1PFS	8	8
A000 02CEh	MPC	Port S6 pin function control register	PS6PFS	8	8
A000 02CFh	MPC	Port S7 pin function control register	PS7PFS	8	8
A000 02DFh	MPC	Port U7 pin function control register	PU7PFS	8	8
A000 02FFh	MPC	Write protection register	PWPR	8	8
A000 5000h	SPIBSC	Common control register	CMNCR	32	32
A000 5004h	SPIBSC	SSL delay register	SSLDR	32	32
A000 5008h	SPIBSC	Bit rate register	SPBCR	32	32
A000 500Ch	SPIBSC	Data read control register	DRCR	32	32
A000 5010h	SPIBSC	Data read command setting register	DRCMR	32	32
A000 5014h	SPIBSC	Data read extended address setting register	DREAR	32	32
A000 5018h	SPIBSC	Data read option setting register	DROPR	32	32
A000 501Ch	SPIBSC	Data read enable setting register	DRENR	32	32
A000 5020h	SPIBSC	SPI mode control register	SMCR	32	32
A000 5024h	SPIBSC	SPI mode command setting register	SMCMR	32	32
A000 5028h	SPIBSC	SPI mode address setting register	SMADR	32	32
A000 502Ch	SPIBSC	SPI mode option setting register	SMOPR	32	32
A000 5030h	SPIBSC	SPI mode enable setting register	SMENR	32	32
A000 5038h	SPIBSC	SPI mode read data register 0	SMRDR0	32	8, 16, 32
A000 5040h	SPIBSC	SPI mode write data register 0	SMWDR0	32	8, 16, 32
A000 5048h	SPIBSC	Common status register	CMNSR	32	32
A000 5058h	SPIBSC	Data read dummy cycle setting register	DRDMCR	32	32

Table 5.1 List of I/O Registers (Address Order) (6 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A000 5060h	SPIBSC	SPI mode dummy cycle setting register	SMDMCR	32	32
A001 0000h	VIC	IRQ status register 0	IRQS0	32	32
A001 0004h	VIC	IRQ status register 1	IRQS1	32	32
A001 0008h	VIC	IRQ status register 2	IRQS2	32	32
A001 000Ch	VIC	IRQ status register 3	IRQS3	32	32
A001 0010h	VIC	IRQ status register 4	IRQS4	32	32
A001 0014h	VIC	IRQ status register 5	IRQS5	32	32
A001 0018h	VIC	IRQ status register 6	IRQS6	32	32
A001 001Ch	VIC	IRQ status register 7	IRQS7	32	32
A001 0040h	VIC	Interrupt input status register 0	RAIS0	32	32
A001 0044h	VIC	Interrupt input status register 1	RAIS1	32	32
A001 0048h	VIC	Interrupt input status register 2	RAIS2	32	32
A001 004Ch	VIC	Interrupt input status register 3	RAIS3	32	32
A001 0050h	VIC	Interrupt input status register 4	RAIS4	32	32
A001 0054h	VIC	Interrupt input status register 5	RAIS5	32	32
A001 0058h	VIC	Interrupt input status register 6	RAIS6	32	32
A001 005Ch	VIC	Interrupt input status register 7	RAIS7	32	32
A001 0080h	VIC	Interrupt enable register 0	IEN0	32	32
A001 0084h	VIC	Interrupt enable register 1	IEN1	32	32
A001 0088h	VIC	Interrupt enable register 2	IEN2	32	32
A001 008Ch	VIC	Interrupt enable register 3	IEN3	32	32
A001 0090h	VIC	Interrupt enable register 4	IEN4	32	32
A001 0094h	VIC	Interrupt enable register 5	IEN5	32	32
A001 0098h	VIC	Interrupt enable register 6	IEN6	32	32
A001 009Ch	VIC	Interrupt enable register 7	IEN7	32	32
A001 00A0h	VIC	Interrupt enable clear register 0	IEC0	32	32
A001 00A4h	VIC	Interrupt enable clear register 1	IEC1	32	32
A001 00A8h	VIC	Interrupt enable clear register 2	IEC2	32	32
A001 00ACh	VIC	Interrupt enable clear register 3	IEC3	32	32
A001 00B0h	VIC	Interrupt enable clear register 4	IEC4	32	32
A001 00B4h	VIC	Interrupt enable clear register 5	IEC5	32	32
A001 00B8h	VIC	Interrupt enable clear register 6	IEC6	32	32
A001 00BCh	VIC	Interrupt enable clear register 7	IEC7	32	32
A001 0100h	VIC	Interrupt detection type select register 0	PLS0	32	32
A001 0104h	VIC	Interrupt detection type select register 1	PLS1	32	32
A001 0108h	VIC	Interrupt detection type select register 2	PLS2	32	32
A001 010Ch	VIC	Interrupt detection type select register 3	PLS3	32	32
A001 0110h	VIC	Interrupt detection type select register 4	PLS4	32	32
A001 0114h	VIC	Interrupt detection type select register 5	PLS5	32	32
A001 0118h	VIC	Interrupt detection type select register 6	PLS6	32	32
A001 011Ch	VIC	Interrupt detection type select register 7	PLS7	32	32
A001 0120h	VIC	Edge detection bit clear register 0	PIC0	32	32
A001 0124h	VIC	Edge detection bit clear register 1	PIC1	32	32
A001 0128h	VIC	Edge detection bit clear register 2	PIC2	32	32
A001 012Ch	VIC	Edge detection bit clear register 3	PIC3	32	32
A001 0130h	VIC	Edge detection bit clear register 4	PIC4	32	32
A001 0134h	VIC	Edge detection bit clear register 5	PIC5	32	32
A001 0138h	VIC	Edge detection bit clear register 6	PIC6	32	32

Table 5.1 List of I/O Registers (Address Order) (7 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 013Ch	VIC	Edge detection bit clear register 7	PIC7	32	32
A001 01C0h	VIC	Interrupt priority level mask register 0	PRLM0	32	32
A001 01C4h	VIC	Interrupt priority level mask clear register 0	PRLC0	32	32
A001 01C8h	VIC	User mode enable register 0	UEN0	32	32
A001 0200h	VIC	Interrupt address register 0	HVA0	32	32
A001 0210h	VIC	Interrupt service status register 0	ISS0	32	32
A001 0214h	VIC	Interrupt service status register 1	ISS1	32	32
A001 0218h	VIC	Interrupt service status register 2	ISS2	32	32
A001 021Ch	VIC	Interrupt service status register 3	ISS3	32	32
A001 0220h	VIC	Interrupt service status register 4	ISS4	32	32
A001 0224h	VIC	Interrupt service status register 5	ISS5	32	32
A001 0228h	VIC	Interrupt service status register 6	ISS6	32	32
A001 022Ch	VIC	Interrupt service status register 7	ISS7	32	32
A001 0230h	VIC	Interrupt service current register 0	ISC0	32	32
A001 0234h	VIC	Interrupt service current register 1	ISC1	32	32
A001 0238h	VIC	Interrupt service current register 2	ISC2	32	32
A001 023Ch	VIC	Interrupt service current register 3	ISC3	32	32
A001 0240h	VIC	Interrupt service current register 4	ISC4	32	32
A001 0244h	VIC	Interrupt service current register 5	ISC5	32	32
A001 0248h	VIC	Interrupt service current register 6	ISC6	32	32
A001 024Ch	VIC	Interrupt service current register 7	ISC7	32	32
A001 0404h	VIC	Interrupt address storage register 1	VAD1	32	32
A001 0408h	VIC	Interrupt address storage register 2	VAD2	32	32
A001 040Ch	VIC	Interrupt address storage register 3	VAD3	32	32
A001 0410h	VIC	Interrupt address storage register 4	VAD4	32	32
A001 0414h	VIC	Interrupt address storage register 5	VAD5	32	32
A001 0418h	VIC	Interrupt address storage register 6	VAD6	32	32
A001 041Ch	VIC	Interrupt address storage register 7	VAD7	32	32
A001 0420h	VIC	Interrupt address storage register 8	VAD8	32	32
A001 0424h	VIC	Interrupt address storage register 9	VAD9	32	32
A001 0428h	VIC	Interrupt address storage register 10	VAD10	32	32
A001 042Ch	VIC	Interrupt address storage register 11	VAD11	32	32
A001 0430h	VIC	Interrupt address storage register 12	VAD12	32	32
A001 0434h	VIC	Interrupt address storage register 13	VAD13	32	32
A001 0438h	VIC	Interrupt address storage register 14	VAD14	32	32
A001 043Ch	VIC	Interrupt address storage register 15	VAD15	32	32
A001 0440h	VIC	Interrupt address storage register 16	VAD16	32	32
A001 0444h	VIC	Interrupt address storage register 17	VAD17	32	32
A001 0448h	VIC	Interrupt address storage register 18	VAD18	32	32
A001 044Ch	VIC	Interrupt address storage register 19	VAD19	32	32
A001 0450h	VIC	Interrupt address storage register 20	VAD20	32	32
A001 0454h	VIC	Interrupt address storage register 21	VAD21	32	32
A001 0458h	VIC	Interrupt address storage register 22	VAD22	32	32
A001 045Ch	VIC	Interrupt address storage register 23	VAD23	32	32
A001 0460h	VIC	Interrupt address storage register 24	VAD24	32	32
A001 0464h	VIC	Interrupt address storage register 25	VAD25	32	32
A001 0468h	VIC	Interrupt address storage register 26	VAD26	32	32
A001 046Ch	VIC	Interrupt address storage register 27	VAD27	32	32

Table 5.1 List of I/O Registers (Address Order) (8 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0470h	VIC	Interrupt address storage register 28	VAD28	32	32
A001 0474h	VIC	Interrupt address storage register 29	VAD29	32	32
A001 0478h	VIC	Interrupt address storage register 30	VAD30	32	32
A001 047Ch	VIC	Interrupt address storage register 31	VAD31	32	32
A001 0480h	VIC	Interrupt address storage register 32	VAD32	32	32
A001 0484h	VIC	Interrupt address storage register 33	VAD33	32	32
A001 0488h	VIC	Interrupt address storage register 34	VAD34	32	32
A001 048Ch	VIC	Interrupt address storage register 35	VAD35	32	32
A001 0490h	VIC	Interrupt address storage register 36	VAD36	32	32
A001 0494h	VIC	Interrupt address storage register 37	VAD37	32	32
A001 0498h	VIC	Interrupt address storage register 38	VAD38	32	32
A001 049Ch	VIC	Interrupt address storage register 39	VAD39	32	32
A001 04A0h	VIC	Interrupt address storage register 40	VAD40	32	32
A001 04A4h	VIC	Interrupt address storage register 41	VAD41	32	32
A001 04A8h	VIC	Interrupt address storage register 42	VAD42	32	32
A001 04ACh	VIC	Interrupt address storage register 43	VAD43	32	32
A001 04B0h	VIC	Interrupt address storage register 44	VAD44	32	32
A001 04B4h	VIC	Interrupt address storage register 45	VAD45	32	32
A001 04B8h	VIC	Interrupt address storage register 46	VAD46	32	32
A001 04BCh	VIC	Interrupt address storage register 47	VAD47	32	32
A001 04C0h	VIC	Interrupt address storage register 48	VAD48	32	32
A001 04C4h	VIC	Interrupt address storage register 49	VAD49	32	32
A001 04C8h	VIC	Interrupt address storage register 50	VAD50	32	32
A001 04CCh	VIC	Interrupt address storage register 51	VAD51	32	32
A001 04D0h	VIC	Interrupt address storage register 52	VAD52	32	32
A001 04D4h	VIC	Interrupt address storage register 53	VAD53	32	32
A001 04D8h	VIC	Interrupt address storage register 54	VAD54	32	32
A001 04DCh	VIC	Interrupt address storage register 55	VAD55	32	32
A001 04E0h	VIC	Interrupt address storage register 56	VAD56	32	32
A001 04E4h	VIC	Interrupt address storage register 57	VAD57	32	32
A001 04E8h	VIC	Interrupt address storage register 58	VAD58	32	32
A001 04ECh	VIC	Interrupt address storage register 59	VAD59	32	32
A001 04F0h	VIC	Interrupt address storage register 60	VAD60	32	32
A001 04F4h	VIC	Interrupt address storage register 61	VAD61	32	32
A001 04F8h	VIC	Interrupt address storage register 62	VAD62	32	32
A001 04FCh	VIC	Interrupt address storage register 63	VAD63	32	32
A001 0500h	VIC	Interrupt address storage register 64	VAD64	32	32
A001 0504h	VIC	Interrupt address storage register 65	VAD65	32	32
A001 0508h	VIC	Interrupt address storage register 66	VAD66	32	32
A001 050Ch	VIC	Interrupt address storage register 67	VAD67	32	32
A001 0510h	VIC	Interrupt address storage register 68	VAD68	32	32
A001 0514h	VIC	Interrupt address storage register 69	VAD69	32	32
A001 0518h	VIC	Interrupt address storage register 70	VAD70	32	32
A001 051Ch	VIC	Interrupt address storage register 71	VAD71	32	32
A001 0520h	VIC	Interrupt address storage register 72	VAD72	32	32
A001 0524h	VIC	Interrupt address storage register 73	VAD73	32	32
A001 0528h	VIC	Interrupt address storage register 74	VAD74	32	32
A001 052Ch	VIC	Interrupt address storage register 75	VAD75	32	32

Table 5.1 List of I/O Registers (Address Order) (9 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0530h	VIC	Interrupt address storage register 76	VAD76	32	32
A001 0534h	VIC	Interrupt address storage register 77	VAD77	32	32
A001 0538h	VIC	Interrupt address storage register 78	VAD78	32	32
A001 053Ch	VIC	Interrupt address storage register 79	VAD79	32	32
A001 0540h	VIC	Interrupt address storage register 80	VAD80	32	32
A001 0544h	VIC	Interrupt address storage register 81	VAD81	32	32
A001 0548h	VIC	Interrupt address storage register 82	VAD82	32	32
A001 054Ch	VIC	Interrupt address storage register 83	VAD83	32	32
A001 0550h	VIC	Interrupt address storage register 84	VAD84	32	32
A001 0554h	VIC	Interrupt address storage register 85	VAD85	32	32
A001 0558h	VIC	Interrupt address storage register 86	VAD86	32	32
A001 055Ch	VIC	Interrupt address storage register 87	VAD87	32	32
A001 0560h	VIC	Interrupt address storage register 88	VAD88	32	32
A001 0564h	VIC	Interrupt address storage register 89	VAD89	32	32
A001 0568h	VIC	Interrupt address storage register 90	VAD90	32	32
A001 056Ch	VIC	Interrupt address storage register 91	VAD91	32	32
A001 0570h	VIC	Interrupt address storage register 92	VAD92	32	32
A001 0574h	VIC	Interrupt address storage register 93	VAD93	32	32
A001 0578h	VIC	Interrupt address storage register 94	VAD94	32	32
A001 057Ch	VIC	Interrupt address storage register 95	VAD95	32	32
A001 0580h	VIC	Interrupt address storage register 96	VAD96	32	32
A001 0584h	VIC	Interrupt address storage register 97	VAD97	32	32
A001 0588h	VIC	Interrupt address storage register 98	VAD98	32	32
A001 058Ch	VIC	Interrupt address storage register 99	VAD99	32	32
A001 0590h	VIC	Interrupt address storage register 100	VAD100	32	32
A001 0594h	VIC	Interrupt address storage register 101	VAD101	32	32
A001 0598h	VIC	Interrupt address storage register 102	VAD102	32	32
A001 059Ch	VIC	Interrupt address storage register 103	VAD103	32	32
A001 05A0h	VIC	Interrupt address storage register 104	VAD104	32	32
A001 05A4h	VIC	Interrupt address storage register 105	VAD105	32	32
A001 05A8h	VIC	Interrupt address storage register 106	VAD106	32	32
A001 05ACh	VIC	Interrupt address storage register 107	VAD107	32	32
A001 05B0h	VIC	Interrupt address storage register 108	VAD108	32	32
A001 05B4h	VIC	Interrupt address storage register 109	VAD109	32	32
A001 05B8h	VIC	Interrupt address storage register 110	VAD110	32	32
A001 05BCh	VIC	Interrupt address storage register 111	VAD111	32	32
A001 05C0h	VIC	Interrupt address storage register 112	VAD112	32	32
A001 05C4h	VIC	Interrupt address storage register 113	VAD113	32	32
A001 05C8h	VIC	Interrupt address storage register 114	VAD114	32	32
A001 05CCh	VIC	Interrupt address storage register 115	VAD115	32	32
A001 05D0h	VIC	Interrupt address storage register 116	VAD116	32	32
A001 05D4h	VIC	Interrupt address storage register 117	VAD117	32	32
A001 05D8h	VIC	Interrupt address storage register 118	VAD118	32	32
A001 05DCh	VIC	Interrupt address storage register 119	VAD119	32	32
A001 05E0h	VIC	Interrupt address storage register 120	VAD120	32	32
A001 05E4h	VIC	Interrupt address storage register 121	VAD121	32	32
A001 05E8h	VIC	Interrupt address storage register 122	VAD122	32	32
A001 05ECh	VIC	Interrupt address storage register 123	VAD123	32	32

Table 5.1 List of I/O Registers (Address Order) (10 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 05F0h	VIC	Interrupt address storage register 124	VAD124	32	32
A001 05F4h	VIC	Interrupt address storage register 125	VAD125	32	32
A001 05F8h	VIC	Interrupt address storage register 126	VAD126	32	32
A001 05FCh	VIC	Interrupt address storage register 127	VAD127	32	32
A001 0600h	VIC	Interrupt address storage register 128	VAD128	32	32
A001 0604h	VIC	Interrupt address storage register 129	VAD129	32	32
A001 0608h	VIC	Interrupt address storage register 130	VAD130	32	32
A001 060Ch	VIC	Interrupt address storage register 131	VAD131	32	32
A001 0610h	VIC	Interrupt address storage register 132	VAD132	32	32
A001 0614h	VIC	Interrupt address storage register 133	VAD133	32	32
A001 0618h	VIC	Interrupt address storage register 134	VAD134	32	32
A001 061Ch	VIC	Interrupt address storage register 135	VAD135	32	32
A001 0620h	VIC	Interrupt address storage register 136	VAD136	32	32
A001 0624h	VIC	Interrupt address storage register 137	VAD137	32	32
A001 0628h	VIC	Interrupt address storage register 138	VAD138	32	32
A001 062Ch	VIC	Interrupt address storage register 139	VAD139	32	32
A001 0630h	VIC	Interrupt address storage register 140	VAD140	32	32
A001 0634h	VIC	Interrupt address storage register 141	VAD141	32	32
A001 0638h	VIC	Interrupt address storage register 142	VAD142	32	32
A001 063Ch	VIC	Interrupt address storage register 143	VAD143	32	32
A001 0640h	VIC	Interrupt address storage register 144	VAD144	32	32
A001 0644h	VIC	Interrupt address storage register 145	VAD145	32	32
A001 0648h	VIC	Interrupt address storage register 146	VAD146	32	32
A001 064Ch	VIC	Interrupt address storage register 147	VAD147	32	32
A001 0650h	VIC	Interrupt address storage register 148	VAD148	32	32
A001 0654h	VIC	Interrupt address storage register 149	VAD149	32	32
A001 0658h	VIC	Interrupt address storage register 150	VAD150	32	32
A001 065Ch	VIC	Interrupt address storage register 151	VAD151	32	32
A001 0660h	VIC	Interrupt address storage register 152	VAD152	32	32
A001 0664h	VIC	Interrupt address storage register 153	VAD153	32	32
A001 0668h	VIC	Interrupt address storage register 154	VAD154	32	32
A001 066Ch	VIC	Interrupt address storage register 155	VAD155	32	32
A001 0670h	VIC	Interrupt address storage register 156	VAD156	32	32
A001 0674h	VIC	Interrupt address storage register 157	VAD157	32	32
A001 0678h	VIC	Interrupt address storage register 158	VAD158	32	32
A001 067Ch	VIC	Interrupt address storage register 159	VAD159	32	32
A001 0680h	VIC	Interrupt address storage register 160	VAD160	32	32
A001 0684h	VIC	Interrupt address storage register 161	VAD161	32	32
A001 0688h	VIC	Interrupt address storage register 162	VAD162	32	32
A001 068Ch	VIC	Interrupt address storage register 163	VAD163	32	32
A001 0690h	VIC	Interrupt address storage register 164	VAD164	32	32
A001 0694h	VIC	Interrupt address storage register 165	VAD165	32	32
A001 0698h	VIC	Interrupt address storage register 166	VAD166	32	32
A001 069Ch	VIC	Interrupt address storage register 167	VAD167	32	32
A001 06A0h	VIC	Interrupt address storage register 168	VAD168	32	32
A001 06A4h	VIC	Interrupt address storage register 169	VAD169	32	32
A001 06A8h	VIC	Interrupt address storage register 170	VAD170	32	32
A001 06ACh	VIC	Interrupt address storage register 171	VAD171	32	32

Table 5.1 List of I/O Registers (Address Order) (11 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 06B0h	VIC	Interrupt address storage register 172	VAD172	32	32
A001 06B4h	VIC	Interrupt address storage register 173	VAD173	32	32
A001 06B8h	VIC	Interrupt address storage register 174	VAD174	32	32
A001 06BCh	VIC	Interrupt address storage register 175	VAD175	32	32
A001 06C0h	VIC	Interrupt address storage register 176	VAD176	32	32
A001 06C4h	VIC	Interrupt address storage register 177	VAD177	32	32
A001 06C8h	VIC	Interrupt address storage register 178	VAD178	32	32
A001 06CCh	VIC	Interrupt address storage register 179	VAD179	32	32
A001 06D0h	VIC	Interrupt address storage register 180	VAD180	32	32
A001 06D4h	VIC	Interrupt address storage register 181	VAD181	32	32
A001 06D8h	VIC	Interrupt address storage register 182	VAD182	32	32
A001 06DCh	VIC	Interrupt address storage register 183	VAD183	32	32
A001 06E0h	VIC	Interrupt address storage register 184	VAD184	32	32
A001 06E4h	VIC	Interrupt address storage register 185	VAD185	32	32
A001 06E8h	VIC	Interrupt address storage register 186	VAD186	32	32
A001 06ECh	VIC	Interrupt address storage register 187	VAD187	32	32
A001 06F0h	VIC	Interrupt address storage register 188	VAD188	32	32
A001 06F4h	VIC	Interrupt address storage register 189	VAD189	32	32
A001 06F8h	VIC	Interrupt address storage register 190	VAD190	32	32
A001 06FCh	VIC	Interrupt address storage register 191	VAD191	32	32
A001 0700h	VIC	Interrupt address storage register 192	VAD192	32	32
A001 0704h	VIC	Interrupt address storage register 193	VAD193	32	32
A001 0708h	VIC	Interrupt address storage register 194	VAD194	32	32
A001 070Ch	VIC	Interrupt address storage register 195	VAD195	32	32
A001 0710h	VIC	Interrupt address storage register 196	VAD196	32	32
A001 0714h	VIC	Interrupt address storage register 197	VAD197	32	32
A001 0718h	VIC	Interrupt address storage register 198	VAD198	32	32
A001 071Ch	VIC	Interrupt address storage register 199	VAD199	32	32
A001 0720h	VIC	Interrupt address storage register 200	VAD200	32	32
A001 0724h	VIC	Interrupt address storage register 201	VAD201	32	32
A001 0728h	VIC	Interrupt address storage register 202	VAD202	32	32
A001 072Ch	VIC	Interrupt address storage register 203	VAD203	32	32
A001 0730h	VIC	Interrupt address storage register 204	VAD204	32	32
A001 0734h	VIC	Interrupt address storage register 205	VAD205	32	32
A001 0738h	VIC	Interrupt address storage register 206	VAD206	32	32
A001 073Ch	VIC	Interrupt address storage register 207	VAD207	32	32
A001 0740h	VIC	Interrupt address storage register 208	VAD208	32	32
A001 0744h	VIC	Interrupt address storage register 209	VAD209	32	32
A001 0748h	VIC	Interrupt address storage register 210	VAD210	32	32
A001 074Ch	VIC	Interrupt address storage register 211	VAD211	32	32
A001 0750h	VIC	Interrupt address storage register 212	VAD212	32	32
A001 0754h	VIC	Interrupt address storage register 213	VAD213	32	32
A001 0758h	VIC	Interrupt address storage register 214	VAD214	32	32
A001 075Ch	VIC	Interrupt address storage register 215	VAD215	32	32
A001 0760h	VIC	Interrupt address storage register 216	VAD216	32	32
A001 0764h	VIC	Interrupt address storage register 217	VAD217	32	32
A001 0768h	VIC	Interrupt address storage register 218	VAD218	32	32
A001 076Ch	VIC	Interrupt address storage register 219	VAD219	32	32

Table 5.1 List of I/O Registers (Address Order) (12 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0770h	VIC	Interrupt address storage register 220	VAD220	32	32
A001 0774h	VIC	Interrupt address storage register 221	VAD221	32	32
A001 0778h	VIC	Interrupt address storage register 222	VAD222	32	32
A001 077Ch	VIC	Interrupt address storage register 223	VAD223	32	32
A001 0780h	VIC	Interrupt address storage register 224	VAD224	32	32
A001 0784h	VIC	Interrupt address storage register 225	VAD225	32	32
A001 0788h	VIC	Interrupt address storage register 226	VAD226	32	32
A001 078Ch	VIC	Interrupt address storage register 227	VAD227	32	32
A001 0790h	VIC	Interrupt address storage register 228	VAD228	32	32
A001 0794h	VIC	Interrupt address storage register 229	VAD229	32	32
A001 0798h	VIC	Interrupt address storage register 230	VAD230	32	32
A001 079Ch	VIC	Interrupt address storage register 231	VAD231	32	32
A001 07A0h	VIC	Interrupt address storage register 232	VAD232	32	32
A001 07A4h	VIC	Interrupt address storage register 233	VAD233	32	32
A001 07A8h	VIC	Interrupt address storage register 234	VAD234	32	32
A001 07ACh	VIC	Interrupt address storage register 235	VAD235	32	32
A001 07B0h	VIC	Interrupt address storage register 236	VAD236	32	32
A001 07B4h	VIC	Interrupt address storage register 237	VAD237	32	32
A001 07B8h	VIC	Interrupt address storage register 238	VAD238	32	32
A001 07BCh	VIC	Interrupt address storage register 239	VAD239	32	32
A001 07C0h	VIC	Interrupt address storage register 240	VAD240	32	32
A001 07C4h	VIC	Interrupt address storage register 241	VAD241	32	32
A001 07C8h	VIC	Interrupt address storage register 242	VAD242	32	32
A001 07CCh	VIC	Interrupt address storage register 243	VAD243	32	32
A001 07D0h	VIC	Interrupt address storage register 244	VAD244	32	32
A001 07D4h	VIC	Interrupt address storage register 245	VAD245	32	32
A001 07D8h	VIC	Interrupt address storage register 246	VAD246	32	32
A001 07DCh	VIC	Interrupt address storage register 247	VAD247	32	32
A001 07E0h	VIC	Interrupt address storage register 248	VAD248	32	32
A001 07E4h	VIC	Interrupt address storage register 249	VAD249	32	32
A001 07E8h	VIC	Interrupt address storage register 250	VAD250	32	32
A001 07ECh	VIC	Interrupt address storage register 251	VAD251	32	32
A001 07F0h	VIC	Interrupt address storage register 252	VAD252	32	32
A001 07F4h	VIC	Interrupt address storage register 253	VAD253	32	32
A001 07F8h	VIC	Interrupt address storage register 254	VAD254	32	32
A001 07FCh	VIC	Interrupt address storage register 255	VAD255	32	32
A001 0804h	VIC	Interrupt priority level storage register 1	PRL1	32	32
A001 0808h	VIC	Interrupt priority level storage register 2	PRL2	32	32
A001 080Ch	VIC	Interrupt priority level storage register 3	PRL3	32	32
A001 0810h	VIC	Interrupt priority level storage register 4	PRL4	32	32
A001 0814h	VIC	Interrupt priority level storage register 5	PRL5	32	32
A001 0818h	VIC	Interrupt priority level storage register 6	PRL6	32	32
A001 081Ch	VIC	Interrupt priority level storage register 7	PRL7	32	32
A001 0820h	VIC	Interrupt priority level storage register 8	PRL8	32	32
A001 0824h	VIC	Interrupt priority level storage register 9	PRL9	32	32
A001 0828h	VIC	Interrupt priority level storage register 10	PRL10	32	32
A001 082Ch	VIC	Interrupt priority level storage register 11	PRL11	32	32
A001 0830h	VIC	Interrupt priority level storage register 12	PRL12	32	32

Table 5.1 List of I/O Registers (Address Order) (13 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0834h	VIC	Interrupt priority level storage register 13	PRL13	32	32
A001 0838h	VIC	Interrupt priority level storage register 14	PRL14	32	32
A001 083Ch	VIC	Interrupt priority level storage register 15	PRL15	32	32
A001 0840h	VIC	Interrupt priority level storage register 16	PRL16	32	32
A001 0844h	VIC	Interrupt priority level storage register 17	PRL17	32	32
A001 0848h	VIC	Interrupt priority level storage register 18	PRL18	32	32
A001 084Ch	VIC	Interrupt priority level storage register 19	PRL19	32	32
A001 0850h	VIC	Interrupt priority level storage register 20	PRL20	32	32
A001 0854h	VIC	Interrupt priority level storage register 21	PRL21	32	32
A001 0858h	VIC	Interrupt priority level storage register 22	PRL22	32	32
A001 085Ch	VIC	Interrupt priority level storage register 23	PRL23	32	32
A001 0860h	VIC	Interrupt priority level storage register 24	PRL24	32	32
A001 0864h	VIC	Interrupt priority level storage register 25	PRL25	32	32
A001 0868h	VIC	Interrupt priority level storage register 26	PRL26	32	32
A001 086Ch	VIC	Interrupt priority level storage register 27	PRL27	32	32
A001 0870h	VIC	Interrupt priority level storage register 28	PRL28	32	32
A001 0874h	VIC	Interrupt priority level storage register 29	PRL29	32	32
A001 0878h	VIC	Interrupt priority level storage register 30	PRL30	32	32
A001 087Ch	VIC	Interrupt priority level storage register 31	PRL31	32	32
A001 0880h	VIC	Interrupt priority level storage register 32	PRL32	32	32
A001 0884h	VIC	Interrupt priority level storage register 33	PRL33	32	32
A001 0888h	VIC	Interrupt priority level storage register 34	PRL34	32	32
A001 088Ch	VIC	Interrupt priority level storage register 35	PRL35	32	32
A001 0890h	VIC	Interrupt priority level storage register 36	PRL36	32	32
A001 0894h	VIC	Interrupt priority level storage register 37	PRL37	32	32
A001 0898h	VIC	Interrupt priority level storage register 38	PRL38	32	32
A001 089Ch	VIC	Interrupt priority level storage register 39	PRL39	32	32
A001 08A0h	VIC	Interrupt priority level storage register 40	PRL40	32	32
A001 08A4h	VIC	Interrupt priority level storage register 41	PRL41	32	32
A001 08A8h	VIC	Interrupt priority level storage register 42	PRL42	32	32
A001 08ACh	VIC	Interrupt priority level storage register 43	PRL43	32	32
A001 08B0h	VIC	Interrupt priority level storage register 44	PRL44	32	32
A001 08B4h	VIC	Interrupt priority level storage register 45	PRL45	32	32
A001 08B8h	VIC	Interrupt priority level storage register 46	PRL46	32	32
A001 08BCh	VIC	Interrupt priority level storage register 47	PRL47	32	32
A001 08C0h	VIC	Interrupt priority level storage register 48	PRL48	32	32
A001 08C4h	VIC	Interrupt priority level storage register 49	PRL49	32	32
A001 08C8h	VIC	Interrupt priority level storage register 50	PRL50	32	32
A001 08CCh	VIC	Interrupt priority level storage register 51	PRL51	32	32
A001 08D0h	VIC	Interrupt priority level storage register 52	PRL52	32	32
A001 08D4h	VIC	Interrupt priority level storage register 53	PRL53	32	32
A001 08D8h	VIC	Interrupt priority level storage register 54	PRL54	32	32
A001 08DCh	VIC	Interrupt priority level storage register 55	PRL55	32	32
A001 08E0h	VIC	Interrupt priority level storage register 56	PRL56	32	32
A001 08E4h	VIC	Interrupt priority level storage register 57	PRL57	32	32
A001 08E8h	VIC	Interrupt priority level storage register 58	PRL58	32	32
A001 08ECh	VIC	Interrupt priority level storage register 59	PRL59	32	32
A001 08F0h	VIC	Interrupt priority level storage register 60	PRL60	32	32

Table 5.1 List of I/O Registers (Address Order) (14 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 08F4h	VIC	Interrupt priority level storage register 61	PRL61	32	32
A001 08F8h	VIC	Interrupt priority level storage register 62	PRL62	32	32
A001 08FCh	VIC	Interrupt priority level storage register 63	PRL63	32	32
A001 0900h	VIC	Interrupt priority level storage register 64	PRL64	32	32
A001 0904h	VIC	Interrupt priority level storage register 65	PRL65	32	32
A001 0908h	VIC	Interrupt priority level storage register 66	PRL66	32	32
A001 090Ch	VIC	Interrupt priority level storage register 67	PRL67	32	32
A001 0910h	VIC	Interrupt priority level storage register 68	PRL68	32	32
A001 0914h	VIC	Interrupt priority level storage register 69	PRL69	32	32
A001 0918h	VIC	Interrupt priority level storage register 70	PRL70	32	32
A001 091Ch	VIC	Interrupt priority level storage register 71	PRL71	32	32
A001 0920h	VIC	Interrupt priority level storage register 72	PRL72	32	32
A001 0924h	VIC	Interrupt priority level storage register 73	PRL73	32	32
A001 0928h	VIC	Interrupt priority level storage register 74	PRL74	32	32
A001 092Ch	VIC	Interrupt priority level storage register 75	PRL75	32	32
A001 0930h	VIC	Interrupt priority level storage register 76	PRL76	32	32
A001 0934h	VIC	Interrupt priority level storage register 77	PRL77	32	32
A001 0938h	VIC	Interrupt priority level storage register 78	PRL78	32	32
A001 093Ch	VIC	Interrupt priority level storage register 79	PRL79	32	32
A001 0940h	VIC	Interrupt priority level storage register 80	PRL80	32	32
A001 0944h	VIC	Interrupt priority level storage register 81	PRL81	32	32
A001 0948h	VIC	Interrupt priority level storage register 82	PRL82	32	32
A001 094Ch	VIC	Interrupt priority level storage register 83	PRL83	32	32
A001 0950h	VIC	Interrupt priority level storage register 84	PRL84	32	32
A001 0954h	VIC	Interrupt priority level storage register 85	PRL85	32	32
A001 0958h	VIC	Interrupt priority level storage register 86	PRL86	32	32
A001 095Ch	VIC	Interrupt priority level storage register 87	PRL87	32	32
A001 0960h	VIC	Interrupt priority level storage register 88	PRL88	32	32
A001 0964h	VIC	Interrupt priority level storage register 89	PRL89	32	32
A001 0968h	VIC	Interrupt priority level storage register 90	PRL90	32	32
A001 096Ch	VIC	Interrupt priority level storage register 91	PRL91	32	32
A001 0970h	VIC	Interrupt priority level storage register 92	PRL92	32	32
A001 0974h	VIC	Interrupt priority level storage register 93	PRL93	32	32
A001 0978h	VIC	Interrupt priority level storage register 94	PRL94	32	32
A001 097Ch	VIC	Interrupt priority level storage register 95	PRL95	32	32
A001 0980h	VIC	Interrupt priority level storage register 96	PRL96	32	32
A001 0984h	VIC	Interrupt priority level storage register 97	PRL97	32	32
A001 0988h	VIC	Interrupt priority level storage register 98	PRL98	32	32
A001 098Ch	VIC	Interrupt priority level storage register 99	PRL99	32	32
A001 0990h	VIC	Interrupt priority level storage register 100	PRL100	32	32
A001 0994h	VIC	Interrupt priority level storage register 101	PRL101	32	32
A001 0998h	VIC	Interrupt priority level storage register 102	PRL102	32	32
A001 099Ch	VIC	Interrupt priority level storage register 103	PRL103	32	32
A001 09A0h	VIC	Interrupt priority level storage register 104	PRL104	32	32
A001 09A4h	VIC	Interrupt priority level storage register 105	PRL105	32	32
A001 09A8h	VIC	Interrupt priority level storage register 106	PRL106	32	32
A001 09ACh	VIC	Interrupt priority level storage register 107	PRL107	32	32
A001 09B0h	VIC	Interrupt priority level storage register 108	PRL108	32	32

Table 5.1 List of I/O Registers (Address Order) (15 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 09B4h	VIC	Interrupt priority level storage register 109	PRL109	32	32
A001 09B8h	VIC	Interrupt priority level storage register 110	PRL110	32	32
A001 09BCh	VIC	Interrupt priority level storage register 111	PRL111	32	32
A001 09C0h	VIC	Interrupt priority level storage register 112	PRL112	32	32
A001 09C4h	VIC	Interrupt priority level storage register 113	PRL113	32	32
A001 09C8h	VIC	Interrupt priority level storage register 114	PRL114	32	32
A001 09CCh	VIC	Interrupt priority level storage register 115	PRL115	32	32
A001 09D0h	VIC	Interrupt priority level storage register 116	PRL116	32	32
A001 09D4h	VIC	Interrupt priority level storage register 117	PRL117	32	32
A001 09D8h	VIC	Interrupt priority level storage register 118	PRL118	32	32
A001 09DCh	VIC	Interrupt priority level storage register 119	PRL119	32	32
A001 09E0h	VIC	Interrupt priority level storage register 120	PRL120	32	32
A001 09E4h	VIC	Interrupt priority level storage register 121	PRL121	32	32
A001 09E8h	VIC	Interrupt priority level storage register 122	PRL122	32	32
A001 09ECh	VIC	Interrupt priority level storage register 123	PRL123	32	32
A001 09F0h	VIC	Interrupt priority level storage register 124	PRL124	32	32
A001 09F4h	VIC	Interrupt priority level storage register 125	PRL125	32	32
A001 09F8h	VIC	Interrupt priority level storage register 126	PRL126	32	32
A001 09FCh	VIC	Interrupt priority level storage register 127	PRL127	32	32
A001 0A00h	VIC	Interrupt priority level storage register 128	PRL128	32	32
A001 0A04h	VIC	Interrupt priority level storage register 129	PRL129	32	32
A001 0A08h	VIC	Interrupt priority level storage register 130	PRL130	32	32
A001 0A0Ch	VIC	Interrupt priority level storage register 131	PRL131	32	32
A001 0A10h	VIC	Interrupt priority level storage register 132	PRL132	32	32
A001 0A14h	VIC	Interrupt priority level storage register 133	PRL133	32	32
A001 0A18h	VIC	Interrupt priority level storage register 134	PRL134	32	32
A001 0A1Ch	VIC	Interrupt priority level storage register 135	PRL135	32	32
A001 0A20h	VIC	Interrupt priority level storage register 136	PRL136	32	32
A001 0A24h	VIC	Interrupt priority level storage register 137	PRL137	32	32
A001 0A28h	VIC	Interrupt priority level storage register 138	PRL138	32	32
A001 0A2Ch	VIC	Interrupt priority level storage register 139	PRL139	32	32
A001 0A30h	VIC	Interrupt priority level storage register 140	PRL140	32	32
A001 0A34h	VIC	Interrupt priority level storage register 141	PRL141	32	32
A001 0A38h	VIC	Interrupt priority level storage register 142	PRL142	32	32
A001 0A3Ch	VIC	Interrupt priority level storage register 143	PRL143	32	32
A001 0A40h	VIC	Interrupt priority level storage register 144	PRL144	32	32
A001 0A44h	VIC	Interrupt priority level storage register 145	PRL145	32	32
A001 0A48h	VIC	Interrupt priority level storage register 146	PRL146	32	32
A001 0A4Ch	VIC	Interrupt priority level storage register 147	PRL147	32	32
A001 0A50h	VIC	Interrupt priority level storage register 148	PRL148	32	32
A001 0A54h	VIC	Interrupt priority level storage register 149	PRL149	32	32
A001 0A58h	VIC	Interrupt priority level storage register 150	PRL150	32	32
A001 0A5Ch	VIC	Interrupt priority level storage register 151	PRL151	32	32
A001 0A60h	VIC	Interrupt priority level storage register 152	PRL152	32	32
A001 0A64h	VIC	Interrupt priority level storage register 153	PRL153	32	32
A001 0A68h	VIC	Interrupt priority level storage register 154	PRL154	32	32
A001 0A6Ch	VIC	Interrupt priority level storage register 155	PRL155	32	32
A001 0A70h	VIC	Interrupt priority level storage register 156	PRL156	32	32

Table 5.1 List of I/O Registers (Address Order) (16 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0A74h	VIC	Interrupt priority level storage register 157	PRL157	32	32
A001 0A78h	VIC	Interrupt priority level storage register 158	PRL158	32	32
A001 0A7Ch	VIC	Interrupt priority level storage register 159	PRL159	32	32
A001 0A80h	VIC	Interrupt priority level storage register 160	PRL160	32	32
A001 0A84h	VIC	Interrupt priority level storage register 161	PRL161	32	32
A001 0A88h	VIC	Interrupt priority level storage register 162	PRL162	32	32
A001 0A8Ch	VIC	Interrupt priority level storage register 163	PRL163	32	32
A001 0A90h	VIC	Interrupt priority level storage register 164	PRL164	32	32
A001 0A94h	VIC	Interrupt priority level storage register 165	PRL165	32	32
A001 0A98h	VIC	Interrupt priority level storage register 166	PRL166	32	32
A001 0A9Ch	VIC	Interrupt priority level storage register 167	PRL167	32	32
A001 0AA0h	VIC	Interrupt priority level storage register 168	PRL168	32	32
A001 0AA4h	VIC	Interrupt priority level storage register 169	PRL169	32	32
A001 0AA8h	VIC	Interrupt priority level storage register 170	PRL170	32	32
A001 0AACh	VIC	Interrupt priority level storage register 171	PRL171	32	32
A001 0AB0h	VIC	Interrupt priority level storage register 172	PRL172	32	32
A001 0AB4h	VIC	Interrupt priority level storage register 173	PRL173	32	32
A001 0AB8h	VIC	Interrupt priority level storage register 174	PRL174	32	32
A001 0ABCh	VIC	Interrupt priority level storage register 175	PRL175	32	32
A001 0AC0h	VIC	Interrupt priority level storage register 176	PRL176	32	32
A001 0AC4h	VIC	Interrupt priority level storage register 177	PRL177	32	32
A001 0AC8h	VIC	Interrupt priority level storage register 178	PRL178	32	32
A001 0ACCh	VIC	Interrupt priority level storage register 179	PRL179	32	32
A001 0AD0h	VIC	Interrupt priority level storage register 180	PRL180	32	32
A001 0AD4h	VIC	Interrupt priority level storage register 181	PRL181	32	32
A001 0AD8h	VIC	Interrupt priority level storage register 182	PRL182	32	32
A001 0ADCh	VIC	Interrupt priority level storage register 183	PRL183	32	32
A001 0AE0h	VIC	Interrupt priority level storage register 184	PRL184	32	32
A001 0AE4h	VIC	Interrupt priority level storage register 185	PRL185	32	32
A001 0AE8h	VIC	Interrupt priority level storage register 186	PRL186	32	32
A001 0AECh	VIC	Interrupt priority level storage register 187	PRL187	32	32
A001 0AF0h	VIC	Interrupt priority level storage register 188	PRL188	32	32
A001 0AF4h	VIC	Interrupt priority level storage register 189	PRL189	32	32
A001 0AF8h	VIC	Interrupt priority level storage register 190	PRL190	32	32
A001 0AFCh	VIC	Interrupt priority level storage register 191	PRL191	32	32
A001 0B00h	VIC	Interrupt priority level storage register 192	PRL192	32	32
A001 0B04h	VIC	Interrupt priority level storage register 193	PRL193	32	32
A001 0B08h	VIC	Interrupt priority level storage register 194	PRL194	32	32
A001 0B0Ch	VIC	Interrupt priority level storage register 195	PRL195	32	32
A001 0B10h	VIC	Interrupt priority level storage register 196	PRL196	32	32
A001 0B14h	VIC	Interrupt priority level storage register 197	PRL197	32	32
A001 0B18h	VIC	Interrupt priority level storage register 198	PRL198	32	32
A001 0B1Ch	VIC	Interrupt priority level storage register 199	PRL199	32	32
A001 0B20h	VIC	Interrupt priority level storage register 200	PRL200	32	32
A001 0B24h	VIC	Interrupt priority level storage register 201	PRL201	32	32
A001 0B28h	VIC	Interrupt priority level storage register 202	PRL202	32	32
A001 0B2Ch	VIC	Interrupt priority level storage register 203	PRL203	32	32
A001 0B30h	VIC	Interrupt priority level storage register 204	PRL204	32	32

Table 5.1 List of I/O Registers (Address Order) (17 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0B34h	VIC	Interrupt priority level storage register 205	PRL205	32	32
A001 0B38h	VIC	Interrupt priority level storage register 206	PRL206	32	32
A001 0B3Ch	VIC	Interrupt priority level storage register 207	PRL207	32	32
A001 0B40h	VIC	Interrupt priority level storage register 208	PRL208	32	32
A001 0B44h	VIC	Interrupt priority level storage register 209	PRL209	32	32
A001 0B48h	VIC	Interrupt priority level storage register 210	PRL210	32	32
A001 0B4Ch	VIC	Interrupt priority level storage register 211	PRL211	32	32
A001 0B50h	VIC	Interrupt priority level storage register 212	PRL212	32	32
A001 0B54h	VIC	Interrupt priority level storage register 213	PRL213	32	32
A001 0B58h	VIC	Interrupt priority level storage register 214	PRL214	32	32
A001 0B5Ch	VIC	Interrupt priority level storage register 215	PRL215	32	32
A001 0B60h	VIC	Interrupt priority level storage register 216	PRL216	32	32
A001 0B64h	VIC	Interrupt priority level storage register 217	PRL217	32	32
A001 0B68h	VIC	Interrupt priority level storage register 218	PRL218	32	32
A001 0B6Ch	VIC	Interrupt priority level storage register 219	PRL219	32	32
A001 0B70h	VIC	Interrupt priority level storage register 220	PRL220	32	32
A001 0B74h	VIC	Interrupt priority level storage register 221	PRL221	32	32
A001 0B78h	VIC	Interrupt priority level storage register 222	PRL222	32	32
A001 0B7Ch	VIC	Interrupt priority level storage register 223	PRL223	32	32
A001 0B80h	VIC	Interrupt priority level storage register 224	PRL224	32	32
A001 0B84h	VIC	Interrupt priority level storage register 225	PRL225	32	32
A001 0B88h	VIC	Interrupt priority level storage register 226	PRL226	32	32
A001 0B8Ch	VIC	Interrupt priority level storage register 227	PRL227	32	32
A001 0B90h	VIC	Interrupt priority level storage register 228	PRL228	32	32
A001 0B94h	VIC	Interrupt priority level storage register 229	PRL229	32	32
A001 0B98h	VIC	Interrupt priority level storage register 230	PRL230	32	32
A001 0B9Ch	VIC	Interrupt priority level storage register 231	PRL231	32	32
A001 0BA0h	VIC	Interrupt priority level storage register 232	PRL232	32	32
A001 0BA4h	VIC	Interrupt priority level storage register 233	PRL233	32	32
A001 0BA8h	VIC	Interrupt priority level storage register 234	PRL234	32	32
A001 0BACh	VIC	Interrupt priority level storage register 235	PRL235	32	32
A001 0BB0h	VIC	Interrupt priority level storage register 236	PRL236	32	32
A001 0BB4h	VIC	Interrupt priority level storage register 237	PRL237	32	32
A001 0BB8h	VIC	Interrupt priority level storage register 238	PRL238	32	32
A001 0BBCh	VIC	Interrupt priority level storage register 239	PRL239	32	32
A001 0BC0h	VIC	Interrupt priority level storage register 240	PRL240	32	32
A001 0BC4h	VIC	Interrupt priority level storage register 241	PRL241	32	32
A001 0BC8h	VIC	Interrupt priority level storage register 242	PRL242	32	32
A001 0BCCh	VIC	Interrupt priority level storage register 243	PRL243	32	32
A001 0BD0h	VIC	Interrupt priority level storage register 244	PRL244	32	32
A001 0BD4h	VIC	Interrupt priority level storage register 245	PRL245	32	32
A001 0BD8h	VIC	Interrupt priority level storage register 246	PRL246	32	32
A001 0BDCh	VIC	Interrupt priority level storage register 247	PRL247	32	32
A001 0BE0h	VIC	Interrupt priority level storage register 248	PRL248	32	32
A001 0BE4h	VIC	Interrupt priority level storage register 249	PRL249	32	32
A001 0BE8h	VIC	Interrupt priority level storage register 250	PRL250	32	32
A001 0BECh	VIC	Interrupt priority level storage register 251	PRL251	32	32
A001 0BF0h	VIC	Interrupt priority level storage register 252	PRL252	32	32

Table 5.1 List of I/O Registers (Address Order) (18 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 0BF4h	VIC	Interrupt priority level storage register 253	PRL253	32	32
A001 0BF8h	VIC	Interrupt priority level storage register 254	PRL254	32	32
A001 0BFCh	VIC	Interrupt priority level storage register 255	PRL255	32	32
A001 1000h	VIC	IRQ status register 8	IRQS8	32	32
A001 1004h	VIC	IRQ status register 9	IRQS9	32	32
A001 1040h	VIC	Interrupt input status register 8	RAIS8	32	32
A001 1044h	VIC	Interrupt input status register 9	RAIS9	32	32
A001 1080h	VIC	Interrupt enable register 8	IEN8	32	32
A001 1084h	VIC	Interrupt enable register 9	IEN9	32	32
A001 10A0h	VIC	Interrupt enable clear register 8	IEC8	32	32
A001 10A4h	VIC	Interrupt enable clear register 9	IEC9	32	32
A001 1100h	VIC	Interrupt detection type select register 8	PLS8	32	32
A001 1104h	VIC	Interrupt detection type select register 9	PLS9	32	32
A001 1120h	VIC	Edge detection bit clear register 8	PIC8	32	32
A001 1124h	VIC	Edge detection bit clear register 9	PIC9	32	32
A001 11C0h	VIC	Interrupt priority level mask register 1	PRLM1	32	32
A001 11C4h	VIC	Interrupt priority level mask clear register 1	PRLC1	32	32
A001 11C8h	VIC	User mode enable register 1	UEN1	32	32
A001 1210h	VIC	Interrupt service status register 8	ISS8	32	32
A001 1214h	VIC	Interrupt service status register 9	ISS9	32	32
A001 1230h	VIC	Interrupt service current register 8	ISC8	32	32
A001 1234h	VIC	Interrupt service current register 9	ISC9	32	32
A001 1400h	VIC	Interrupt address storage register 256	VAD256	32	32
A001 1404h	VIC	Interrupt address storage register 257	VAD257	32	32
A001 1408h	VIC	Interrupt address storage register 258	VAD258	32	32
A001 140Ch	VIC	Interrupt address storage register 259	VAD259	32	32
A001 1410h	VIC	Interrupt address storage register 260	VAD260	32	32
A001 1414h	VIC	Interrupt address storage register 261	VAD261	32	32
A001 1418h	VIC	Interrupt address storage register 262	VAD262	32	32
A001 141Ch	VIC	Interrupt address storage register 263	VAD263	32	32
A001 1420h	VIC	Interrupt address storage register 264	VAD264	32	32
A001 1424h	VIC	Interrupt address storage register 265	VAD265	32	32
A001 1428h	VIC	Interrupt address storage register 266	VAD266	32	32
A001 142Ch	VIC	Interrupt address storage register 267	VAD267	32	32
A001 1430h	VIC	Interrupt address storage register 268	VAD268	32	32
A001 1434h	VIC	Interrupt address storage register 269	VAD269	32	32
A001 1438h	VIC	Interrupt address storage register 270	VAD270	32	32
A001 143Ch	VIC	Interrupt address storage register 271	VAD271	32	32
A001 1440h	VIC	Interrupt address storage register 272	VAD272	32	32
A001 1444h	VIC	Interrupt address storage register 273	VAD273	32	32
A001 1448h	VIC	Interrupt address storage register 274	VAD274	32	32
A001 144Ch	VIC	Interrupt address storage register 275	VAD275	32	32
A001 1450h	VIC	Interrupt address storage register 276	VAD276	32	32
A001 1454h	VIC	Interrupt address storage register 277	VAD277	32	32
A001 1458h	VIC	Interrupt address storage register 278	VAD278	32	32
A001 145Ch	VIC	Interrupt address storage register 279	VAD279	32	32
A001 1460h	VIC	Interrupt address storage register 280	VAD280	32	32
A001 1464h	VIC	Interrupt address storage register 281	VAD281	32	32

Table 5.1 List of I/O Registers (Address Order) (19 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 1468h	VIC	Interrupt address storage register 282	VAD282	32	32
A001 146Ch	VIC	Interrupt address storage register 283	VAD283	32	32
A001 1470h	VIC	Interrupt address storage register 284	VAD284	32	32
A001 1474h	VIC	Interrupt address storage register 285	VAD285	32	32
A001 1478h	VIC	Interrupt address storage register 286	VAD286	32	32
A001 147Ch	VIC	Interrupt address storage register 287	VAD287	32	32
A001 1480h	VIC	Interrupt address storage register 288	VAD288	32	32
A001 1484h	VIC	Interrupt address storage register 289	VAD289	32	32
A001 1488h	VIC	Interrupt address storage register 290	VAD290	32	32
A001 148Ch	VIC	Interrupt address storage register 291	VAD291	32	32
A001 1490h	VIC	Interrupt address storage register 292	VAD292	32	32
A001 1494h	VIC	Interrupt address storage register 293	VAD293	32	32
A001 1498h	VIC	Interrupt address storage register 294	VAD294	32	32
A001 149Ch	VIC	Interrupt address storage register 295	VAD295	32	32
A001 14A0h	VIC	Interrupt address storage register 296	VAD296	32	32
A001 14A4h	VIC	Interrupt address storage register 297	VAD297	32	32
A001 14A8h	VIC	Interrupt address storage register 298	VAD298	32	32
A001 14ACh	VIC	Interrupt address storage register 299	VAD299	32	32
A001 14B0h	VIC	Interrupt address storage register 300	VAD300	32	32
A001 1800h	VIC	Interrupt priority level storage register 256	PRL256	32	32
A001 1804h	VIC	Interrupt priority level storage register 257	PRL257	32	32
A001 1808h	VIC	Interrupt priority level storage register 258	PRL258	32	32
A001 180Ch	VIC	Interrupt priority level storage register 259	PRL259	32	32
A001 1810h	VIC	Interrupt priority level storage register 260	PRL260	32	32
A001 1814h	VIC	Interrupt priority level storage register 261	PRL261	32	32
A001 1818h	VIC	Interrupt priority level storage register 262	PRL262	32	32
A001 181Ch	VIC	Interrupt priority level storage register 263	PRL263	32	32
A001 1820h	VIC	Interrupt priority level storage register 264	PRL264	32	32
A001 1824h	VIC	Interrupt priority level storage register 265	PRL265	32	32
A001 1828h	VIC	Interrupt priority level storage register 266	PRL266	32	32
A001 182Ch	VIC	Interrupt priority level storage register 267	PRL267	32	32
A001 1830h	VIC	Interrupt priority level storage register 268	PRL268	32	32
A001 1834h	VIC	Interrupt priority level storage register 269	PRL269	32	32
A001 1838h	VIC	Interrupt priority level storage register 270	PRL270	32	32
A001 183Ch	VIC	Interrupt priority level storage register 271	PRL271	32	32
A001 1840h	VIC	Interrupt priority level storage register 272	PRL272	32	32
A001 1844h	VIC	Interrupt priority level storage register 273	PRL273	32	32
A001 1848h	VIC	Interrupt priority level storage register 274	PRL274	32	32
A001 184Ch	VIC	Interrupt priority level storage register 275	PRL275	32	32
A001 1850h	VIC	Interrupt priority level storage register 276	PRL276	32	32
A001 1854h	VIC	Interrupt priority level storage register 277	PRL277	32	32
A001 1858h	VIC	Interrupt priority level storage register 278	PRL278	32	32
A001 185Ch	VIC	Interrupt priority level storage register 279	PRL279	32	32
A001 1860h	VIC	Interrupt priority level storage register 280	PRL280	32	32
A001 1864h	VIC	Interrupt priority level storage register 281	PRL281	32	32
A001 1868h	VIC	Interrupt priority level storage register 282	PRL282	32	32
A001 186Ch	VIC	Interrupt priority level storage register 283	PRL283	32	32
A001 1870h	VIC	Interrupt priority level storage register 284	PRL284	32	32

Table 5.1 List of I/O Registers (Address Order) (20 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A001 1874h	VIC	Interrupt priority level storage register 285	PRL285	32	32
A001 1878h	VIC	Interrupt priority level storage register 286	PRL286	32	32
A001 187Ch	VIC	Interrupt priority level storage register 287	PRL287	32	32
A001 1880h	VIC	Interrupt priority level storage register 288	PRL288	32	32
A001 1884h	VIC	Interrupt priority level storage register 289	PRL289	32	32
A001 1888h	VIC	Interrupt priority level storage register 290	PRL290	32	32
A001 188Ch	VIC	Interrupt priority level storage register 291	PRL291	32	32
A001 1890h	VIC	Interrupt priority level storage register 292	PRL292	32	32
A001 1894h	VIC	Interrupt priority level storage register 293	PRL293	32	32
A001 1898h	VIC	Interrupt priority level storage register 294	PRL294	32	32
A001 189Ch	VIC	Interrupt priority level storage register 295	PRL295	32	32
A001 18A0h	VIC	Interrupt priority level storage register 296	PRL296	32	32
A001 18A4h	VIC	Interrupt priority level storage register 297	PRL297	32	32
A001 18A8h	VIC	Interrupt priority level storage register 298	PRL298	32	32
A001 18ACh	VIC	Interrupt priority level storage register 299	PRL299	32	32
A001 18B0h	VIC	Interrupt priority level storage register 300	PRL300	32	32
A004 0000h	USBh	HcRevision register	HcRevision	32	32
A004 0004h	USBh	HcControl register	HcControl	32	32
A004 0008h	USBh	HcCommandStatus register	HcCommandStatus	32	32
A004 000Ch	USBh	HcInterruptStatus register	HcIntStatus	32	32
A004 0010h	USBh	HcInterruptEnable register	HcIntEnable	32	32
A004 0014h	USBh	HcInterruptDisable register	HcIntDisable	32	32
A004 0018h	USBh	HcHCCA register	HcHCCA	32	32
A004 001Ch	USBh	HcPeriodicCurrentED register	HcPeriodCurED	32	32
A004 0020h	USBh	HcControlHeadED register	HcContHeadED	32	32
A004 0024h	USBh	HcControlCurrentED register	HcContCurrentED	32	32
A004 0028h	USBh	HcBulkHeadED register	HcBulkHeadED	32	32
A004 002Ch	USBh	HcBulkCurrentED register	HcBulkCurrentED	32	32
A004 0030h	USBh	HcDoneHead register	HcDoneHead	32	32
A004 0034h	USBh	HcFmInterval register	HcFmInterval	32	32
A004 0038h	USBh	HcFmRemaining register	HcFmRemaining	32	32
A004 003Ch	USBh	HcFmNumber register	HcFmNumber	32	32
A004 0040h	USBh	HcPeriodicStart register	HcPeriodicStart	32	32
A004 0048h	USBh	HcRhDescriptorA register	HcRhDescriptorA	32	32
A004 004Ch	USBh	HcRhDescriptorB register	HcRhDescriptorB	32	32
A004 0050h	USBh	HcRhStatus register	HcRhStatus_A	32	32
A004 0050h	USBh	HcRhStatus register	HcRhStatus_B	32	32
A004 0054h	USBh	HcRhPortStatus1 register	HcRhPortStatus1_A	32	32
A004 0054h	USBh	HcRhPortStatus1 register	HcRhPortStatus1_B	32	32
A004 1000h	USBh	HCIVERSION / CAPLENGTH register	CAPL_VERSION	32	32
A004 1004h	USBh	HCSPARAMS register	HCSPARAMS	32	32
A004 1008h	USBh	HCCPARAMS register	HCCPARAMS	32	32
A004 100Ch	USBh	HCSP_PORTROUTE register	HCSP_PORTROUTE	32	32
A004 1020h	USBh	USBCMD register	USBCMD	32	32
A004 1024h	USBh	USBSTS register	USBSTS	32	32
A004 1028h	USBh	USBINTR register	USBINTR	32	32
A004 102Ch	USBh	FRINDEX register	FRINDEX	32	32
A004 1030h	USBh	CTRLDSSEGMENT register	CTRLDSSEGMENT	32	32

Table 5.1 List of I/O Registers (Address Order) (21 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A004 1034h	USBh	PERIODICLISTBASE register	PERIODICLIST	32	32
A004 1038h	USBh	ASYNCLISTADDR register	ASYNCLISTADDR	32	32
A004 1060h	USBh	CONFIGFLAG register	CONFIGFLAG	32	32
A004 1064h	USBh	PORTSC1 register	PORTSC1	32	32
A005 0000h	USBh	PCI configuration register for OHCI	VID_DID_O	32	32
A005 0000h	USBh	PCI configuration register for AHB-PCI bridge	VID_DID_A	32	32
A005 0004h	USBh	PCI configuration register for OHCI	CMND_STS_O	32	32
A005 0004h	USBh	PCI configuration register for AHB-PCI bridge	CMND_STS_A	32	32
A005 0008h	USBh	PCI configuration register for OHCI	REVID_CC_O	32	32
A005 0008h	USBh	PCI configuration register for AHB-PCI bridge	REVID_CC_A	32	32
A005 000Ch	USBh	PCI configuration register for OHCI	CLS_LT_HT_BIST_O	32	32
A005 000Ch	USBh	PCI configuration register for AHB-PCI bridge	CLS_LT_HT_BIST_A	32	32
A005 0010h	USBh	PCI configuration register for OHCI	BASEAD_O	32	32
A005 0010h	USBh	PCI configuration register for AHB-PCI bridge	BASEAD_A	32	32
A005 0014h	USBh	PCI configuration register for AHB-PCI bridge	WIN1_BASEAD	32	32
A005 002Ch	USBh	PCI configuration register for OHCI	SSVID_SSID_O	32	32
A005 002Ch	USBh	PCI configuration register for AHB-PCI bridge	SSVID_SSID_A	32	32
A005 0030h	USBh	PCI configuration register for OHCI	EROM_BASEAD	32	32
A005 0034h	USBh	PCI configuration register for OHCI	CAPPTR	32	32
A005 003Ch	USBh	PCI configuration register for OHCI	INTR_LINE_PIN_O	32	32
A005 003Ch	USBh	PCI configuration register for AHB-PCI bridge	INTR_LINE_PIN_A	32	32
A005 0040h	USBh	PCI configuration register for OHCI	CAPID_NIP_PMCAP	32	32
A005 0044h	USBh	PCI configuration register for OHCI	PMC_STS_PMCSR	32	32
A005 00E0h	USBh	PCI configuration register for OHCI	EXT1	32	32
A005 00E4h	USBh	PCI configuration register for OHCI	EXT2	32	32
A005 0100h	USBh	PCI configuration register for EHCI	VID_DID_E	32	32
A005 0104h	USBh	PCI configuration register for EHCI	CMND_STS_E	32	32
A005 0108h	USBh	PCI configuration register for EHCI	REVID_CC_E	32	32
A005 010Ch	USBh	PCI configuration register for EHCI	CLS_LT_HT_BIST_E	32	32
A005 0110h	USBh	PCI configuration register for EHCI	BASEAD_E	32	32
A005 012Ch	USBh	PCI configuration register for EHCI	SSVID_SSID_E	32	32
A005 0130h	USBh	PCI configuration register for EHCI	EROM_BASEAD_E	32	32
A005 0134h	USBh	PCI configuration register for EHCI	CAPPTR_E	32	32
A005 013Ch	USBh	PCI configuration register for EHCI	INTR_LINE_PIN_E	32	32
A005 0140h	USBh	PCI configuration register for EHCI	CAPID_NIP_PMCAP_E	32	32
A005 0144h	USBh	PCI configuration register for EHCI	PMC_STS_PMCSR_E	32	32
A005 0160h	USBh	PCI configuration register for EHCI	SBRN_FLADJ_PW	32	32
A005 01E0h	USBh	PCI configuration register for EHCI	EXT1_E	32	32
A005 01E4h	USBh	PCI configuration register for EHCI	EXT2_E	32	32
A005 0800h	USBh	PCIAHB_WIN1_CTR register	PCIAHB_WIN1_CTR	32	32
A005 0810h	USBh	AHBPCI_WIN1_CTR register	AHBPCI_WIN1_CTR	32	32
A005 0814h	USBh	AHBPCI_WIN2_CTR register	AHBPCI_WIN2_CTR	32	32
A005 0820h	USBh	PCI_INT_ENABLE register	PCI_INT_ENABLE	32	32
A005 0824h	USBh	PCI_INT_STATUS register	PCI_INT_STATUS	32	32
A005 0830h	USBh	AHB_BUS_CTR register	AHB_BUS_CTR	32	32
A005 0834h	USBh	USBCTR register	USBCTR	32	32
A005 0840h	USBh	PCI_ARBITER_CTR register	PCI_ARBITER_CTR	32	32
A005 0848h	USBh	PCI_UNIT_REV register	PCI_UNIT_REV	32	32

Table 5.1 List of I/O Registers (Address Order) (22 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 0000h	USBf	System configuration control register 0	SYSCFG0	16	16
A006 0002h	USBf	System configuration control register 1	SYSCFG1	16	16
A006 0004h	USBf	System configuration status register	SYSSTS0	16	16
A006 0008h	USBf	Device state control register 0	DVSTCTR0	16	16
A006 000Ch	USBf	USB test mode register	TESTMODE	16	16
A006 0010h	USBf	DMA0-FIFO bus configuration register	D0FBCFG	16	16
A006 0012h	USBf	DMA1-FIFO bus configuration register	D1FBCFG	16	16
A006 0014h	USBf	CFIFO port register	CFIFO	32	8, 16, 32
A006 0018h	USBf	D0FIFO port register	D0FIFO	32	8, 16, 32
A006 001Ch	USBf	D1FIFO port register	D1FIFO	32	8, 16, 32
A006 0020h	USBf	CFIFO port select register	CFIFOSEL	16	16
A006 0022h	USBf	CFIFO port control register	CFIFOCTR	16	16
A006 0028h	USBf	D0FIFO port select register	D0FIFOSEL	16	16
A006 002Ah	USBf	D0FIFO port control register	D0FIFOCTR	16	16
A006 002Ch	USBf	D1FIFO port select register	D1FIFOSEL	16	16
A006 002Eh	USBf	D1FIFO port control register	D1FIFOCTR	16	16
A006 0030h	USBf	Interrupt enable register 0	INTENB0	16	16
A006 0036h	USBf	BRDY interrupt enable register	BRDYENB	16	16
A006 0038h	USBf	NRDY interrupt enable register	NRDYENB	16	16
A006 003Ah	USBf	BEMP interrupt enable register	BEMPENB	16	16
A006 003Ch	USBf	SOF pin configuration register	SOFCFG	16	16
A006 0040h	USBf	Interrupt status register 0	INTSTS0	16	16
A006 0046h	USBf	BRDY interrupt status register	BRDYSTS	16	16
A006 0048h	USBf	NRDY interrupt status register	NRDYSTS	16	16
A006 004Ah	USBf	BEMP interrupt status register	BEMPSTS	16	16
A006 004Ch	USBf	Frame number register	FRMNUM	16	16
A006 004Eh	USBf	μ frame number register	UFRMNUM	16	16
A006 0050h	USBf	USB address register	USBADDR	16	16
A006 0054h	USBf	USB request type register	USBREQ	16	16
A006 0056h	USBf	USB request value register	USBVAL	16	16
A006 0058h	USBf	USB request index register	USBINDX	16	16
A006 005Ah	USBf	USB request length register	USBLENG	16	16
A006 005Eh	USBf	DCP max packet size register	DCPMAXP	16	16
A006 0060h	USBf	DCP control register	DCPCTR	16	16
A006 0064h	USBf	Pipe window select register	PIPESEL	16	16
A006 0068h	USBf	Pipe configuration register	PIPECFG	16	16
A006 006Ah	USBf	Pipe buffer specification register	PIPEBUF	16	16
A006 006Ch	USBf	Pipe max packet size register	PIPEMAXP	16	16
A006 006Eh	USBf	Pipe cycle control register	PIPEPERI	16	16
A006 0070h	USBf	PIPE1 control register	PIPE1CTR	16	16
A006 0072h	USBf	PIPE2 control register	PIPE2CTR	16	16
A006 0074h	USBf	PIPE3 control register	PIPE3CTR	16	16
A006 0076h	USBf	PIPE4 control register	PIPE4CTR	16	16
A006 0078h	USBf	PIPE5 control register	PIPE5CTR	16	16
A006 007Ah	USBf	PIPE6 control register	PIPE6CTR	16	16
A006 007Ch	USBf	PIPE7 control register	PIPE7CTR	16	16
A006 007Eh	USBf	PIPE8 control register	PIPE8CTR	16	16
A006 0080h	USBf	PIPE9 control register	PIPE9CTR	16	16

Table 5.1 List of I/O Registers (Address Order) (23 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 0090h	USBf	PIPE1 transaction counter enable register	PIPE1TRE	16	16
A006 0092h	USBf	PIPE1 transaction counter register	PIPE1TRN	16	16
A006 0094h	USBf	PIPE2 transaction counter enable register	PIPE2TRE	16	16
A006 0096h	USBf	PIPE2 transaction counter register	PIPE2TRN	16	16
A006 0098h	USBf	PIPE3 transaction counter enable register	PIPE3TRE	16	16
A006 009Ah	USBf	PIPE3 transaction counter register	PIPE3TRN	16	16
A006 009Ch	USBf	PIPE4 transaction counter enable register	PIPE4TRE	16	16
A006 009Eh	USBf	PIPE4 transaction counter register	PIPE4TRN	16	16
A006 00A0h	USBf	PIPE5 transaction counter enable register	PIPE5TRE	16	16
A006 00A2h	USBf	PIPE5 transaction counter register	PIPE5TRN	16	16
A006 0102h	USBf	Low-power status register	LPSTS	16	16
A006 0160h	USBf	D0FIFO continuous transfer port register 0	D0FIFOB0	32	32
A006 0164h	USBf	D0FIFO continuous transfer port register 1	D0FIFOB1	32	32
A006 0168h	USBf	D0FIFO continuous transfer port register 2	D0FIFOB2	32	32
A006 016Ch	USBf	D0FIFO continuous transfer port register 3	D0FIFOB3	32	32
A006 0170h	USBf	D0FIFO continuous transfer port register 4	D0FIFOB4	32	32
A006 0174h	USBf	D0FIFO continuous transfer port register 5	D0FIFOB5	32	32
A006 0178h	USBf	D0FIFO continuous transfer port register 6	D0FIFOB6	32	32
A006 017Ch	USBf	D0FIFO continuous transfer port register 7	D0FIFOB7	32	32
A006 0180h	USBf	D1FIFO continuous transfer port register 0	D1FIFOB0	32	32
A006 0184h	USBf	D1FIFO continuous transfer port register 1	D1FIFOB1	32	32
A006 0188h	USBf	D1FIFO continuous transfer port register 2	D1FIFOB2	32	32
A006 018Ch	USBf	D1FIFO continuous transfer port register 3	D1FIFOB3	32	32
A006 0190h	USBf	D1FIFO continuous transfer port register 4	D1FIFOB4	32	32
A006 0194h	USBf	D1FIFO continuous transfer port register 5	D1FIFOB5	32	32
A006 0198h	USBf	D1FIFO continuous transfer port register 6	D1FIFOB6	32	32
A006 019Ch	USBf	D1FIFO continuous transfer port register 7	D1FIFOB7	32	32
A006 01A0h	USBf	PHY configuration register 1	PHYSET1	16	16
A006 2000h	DMA0	Next 0 source address register 0	DMAC0_N0SA_0_N	32	32
A006 2000h	DMA0	Next 0 source address register 0	DMAC0_N0SA_0_W	32	32
A006 2004h	DMA0	Next 0 destination address register 0	DMAC0_N0DA_0	32	32
A006 2008h	DMA0	Next 0 transaction byte register 0	DMAC0_N0TB_0	32	32
A006 200Ch	DMA0	Next 1 source address register 0	DMAC0_N1SA_0_N	32	32
A006 200Ch	DMA0	Next 1 source address register 0	DMAC0_N1SA_0_W	32	32
A006 2010h	DMA0	Next 1 destination address register 0	DMAC0_N1DA_0	32	32
A006 2014h	DMA0	Next 1 transaction byte register 0	DMAC0_N1TB_0	32	32
A006 2018h	DMA0	Current source address register 0	DMAC0_CRSA_0	32	32
A006 201Ch	DMA0	Current destination address register 0	DMAC0_CRDA_0	32	32
A006 2020h	DMA0	Current transaction byte register 0	DMAC0_CRTB_0	32	32
A006 2024h	DMA0	Channel status register 0	DMAC0_CHSTAT_0	32	32
A006 2028h	DMA0	Channel control register 0	DMAC0_CHCTRL_0	32	32
A006 202Ch	DMA0	Channel configuration register 0	DMAC0_CHCFG_0	32	32
A006 2030h	DMA0	Channel interval register 0	DMAC0_CHITVL_0	32	32
A006 2038h	DMA0	Next link address register 0	DMAC0_NXLA_0	32	32
A006 203Ch	DMA0	Current link address register 0	DMAC0_CRLA_0	32	32
A006 2040h	DMA0	Next 0 source address register 1	DMAC0_N0SA_1_N	32	32
A006 2040h	DMA0	Next 0 source address register 1	DMAC0_N0SA_1_W	32	32
A006 2044h	DMA0	Next 0 destination address register 1	DMAC0_N0DA_1	32	32

Table 5.1 List of I/O Registers (Address Order) (24 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 2048h	DMA0	Next 0 transaction byte register 1	DMAC0_N0TB_1	32	32
A006 204Ch	DMA0	Next 1 source address register 1	DMAC0_N1SA_1_N	32	32
A006 204Ch	DMA0	Next 1 source address register 1	DMAC0_N1SA_1_W	32	32
A006 2050h	DMA0	Next 1 destination address register 1	DMAC0_N1DA_1	32	32
A006 2054h	DMA0	Next 1 transaction byte register 1	DMAC0_N1TB_1	32	32
A006 2058h	DMA0	Current source address register 1	DMAC0_CRSA_1	32	32
A006 205Ch	DMA0	Current destination address register 1	DMAC0_CRDA_1	32	32
A006 2060h	DMA0	Current transaction byte register 1	DMAC0_CRTB_1	32	32
A006 2064h	DMA0	Channel status register 1	DMAC0_CHSTAT_1	32	32
A006 2068h	DMA0	Channel control register 1	DMAC0_CHCTRL_1	32	32
A006 206Ch	DMA0	Channel configuration register 1	DMAC0_CHCFG_1	32	32
A006 2070h	DMA0	Channel interval register 1	DMAC0_CHITVL_1	32	32
A006 2078h	DMA0	Current link address register 1	DMAC0_NXLA_1	32	32
A006 207Ch	DMA0	Next link address register 1	DMAC0_CRLA_1	32	32
A006 2080h	DMA0	Next 0 source address register 2	DMAC0_N0SA_2_N	32	32
A006 2080h	DMA0	Next 0 source address register 2	DMAC0_N0SA_2_W	32	32
A006 2084h	DMA0	Next 0 destination address register 2	DMAC0_N0DA_2	32	32
A006 2088h	DMA0	Next 0 transaction byte register 2	DMAC0_N0TB_2	32	32
A006 208Ch	DMA0	Next 1 source address register 2	DMAC0_N1SA_2_N	32	32
A006 208Ch	DMA0	Next 1 source address register 2	DMAC0_N1SA_2_W	32	32
A006 2090h	DMA0	Next 1 destination address register 2	DMAC0_N1DA_2	32	32
A006 2094h	DMA0	Next 1 transaction byte register 2	DMAC0_N1TB_2	32	32
A006 2098h	DMA0	Current source address register 2	DMAC0_CRSA_2	32	32
A006 209Ch	DMA0	Current destination address register 2	DMAC0_CRDA_2	32	32
A006 20A0h	DMA0	Current transaction byte register 2	DMAC0_CRTB_2	32	32
A006 20A4h	DMA0	Channel status register 2	DMAC0_CHSTAT_2	32	32
A006 20A8h	DMA0	Channel control register 2	DMAC0_CHCTRL_2	32	32
A006 20ACh	DMA0	Channel configuration register 2	DMAC0_CHCFG_2	32	32
A006 20B0h	DMA0	Channel interval register 2	DMAC0_CHITVL_2	32	32
A006 20B8h	DMA0	Next link address register 2	DMAC0_NXLA_2	32	32
A006 20BCh	DMA0	Current link address register 2	DMAC0_CRLA_2	32	32
A006 20C0h	DMA0	Next 0 source address register 3	DMAC0_N0SA_3_N	32	32
A006 20C0h	DMA0	Next 0 source address register 3	DMAC0_N0SA_3_W	32	32
A006 20C4h	DMA0	Next 0 destination address register 3	DMAC0_N0DA_3	32	32
A006 20C8h	DMA0	Next 0 transaction byte register 3	DMAC0_N0TB_3	32	32
A006 20CCh	DMA0	Next 1 source address register 3	DMAC0_N1SA_3_N	32	32
A006 20CCh	DMA0	Next 1 source address register 3	DMAC0_N1SA_3_W	32	32
A006 20D0h	DMA0	Next 1 destination address register 3	DMAC0_N1DA_3	32	32
A006 20D4h	DMA0	Next 1 transaction byte register 3	DMAC0_N1TB_3	32	32
A006 20D8h	DMA0	Current source address register 3	DMAC0_CRSA_3	32	32
A006 20DCh	DMA0	Current destination address register 3	DMAC0_CRDA_3	32	32
A006 20E0h	DMA0	Current transaction byte register 3	DMAC0_CRTB_3	32	32
A006 20E4h	DMA0	Channel status register 3	DMAC0_CHSTAT_3	32	32
A006 20E8h	DMA0	Channel control register 3	DMAC0_CHCTRL_3	32	32
A006 20ECh	DMA0	Channel configuration register 3	DMAC0_CHCFG_3	32	32
A006 20F0h	DMA0	Channel interval register 3	DMAC0_CHITVL_3	32	32
A006 20F8h	DMA0	Next link address register 3	DMAC0_NXLA_3	32	32
A006 20FCh	DMA0	Current link address register 3	DMAC0_CRLA_3	32	32

Table 5.1 List of I/O Registers (Address Order) (25 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 2100h	DMA0	Next 0 source address register 4	DMAC0_N0SA_4_N	32	32
A006 2100h	DMA0	Next 0 source address register 4	DMAC0_N0SA_4_W	32	32
A006 2104h	DMA0	Next 0 destination address register 4	DMAC0_N0DA_4	32	32
A006 2108h	DMA0	Next 0 transaction byte register 4	DMAC0_N0TB_4	32	32
A006 210Ch	DMA0	Next 1 source address register 4	DMAC0_N1SA_4_N	32	32
A006 210Ch	DMA0	Next 1 source address register 4	DMAC0_N1SA_4_W	32	32
A006 2110h	DMA0	Next 1 destination address register 4	DMAC0_N1DA_4	32	32
A006 2114h	DMA0	Next 1 transaction byte register 4	DMAC0_N1TB_4	32	32
A006 2118h	DMA0	Current source address register 4	DMAC0_CRSA_4	32	32
A006 211Ch	DMA0	Current destination address register 4	DMAC0_CRDA_4	32	32
A006 2120h	DMA0	Current transaction byte register 4	DMAC0_CRTB_4	32	32
A006 2124h	DMA0	Channel status register 4	DMAC0_CHSTAT_4	32	32
A006 2128h	DMA0	Channel control register 4	DMAC0_CHCTRL_4	32	32
A006 212Ch	DMA0	Channel configuration register 4	DMAC0_CHCFG_4	32	32
A006 2130h	DMA0	Channel interval register 4	DMAC0_CHITVL_4	32	32
A006 2138h	DMA0	Next link address register 4	DMAC0_NXLA_4	32	32
A006 213Ch	DMA0	Current link address register 4	DMAC0_CRLA_4	32	32
A006 2140h	DMA0	Next 0 source address register 5	DMAC0_N0SA_5_N	32	32
A006 2140h	DMA0	Next 0 source address register 5	DMAC0_N0SA_5_W	32	32
A006 2144h	DMA0	Next 0 destination address register 5	DMAC0_N0DA_5	32	32
A006 2148h	DMA0	Next 0 transaction byte register 5	DMAC0_N0TB_5	32	32
A006 214Ch	DMA0	Next 1 source address register 5	DMAC0_N1SA_5_N	32	32
A006 214Ch	DMA0	Next 1 source address register 5	DMAC0_N1SA_5_W	32	32
A006 2150h	DMA0	Next 1 destination address register 5	DMAC0_N1DA_5	32	32
A006 2154h	DMA0	Next 1 transaction byte register 5	DMAC0_N1TB_5	32	32
A006 2158h	DMA0	Current source address register 5	DMAC0_CRSA_5	32	32
A006 215Ch	DMA0	Current destination address register 5	DMAC0_CRDA_5	32	32
A006 2160h	DMA0	Current transaction byte register 5	DMAC0_CRTB_5	32	32
A006 2164h	DMA0	Channel status register 5	DMAC0_CHSTAT_5	32	32
A006 2168h	DMA0	Channel control register 5	DMAC0_CHCTRL_5	32	32
A006 216Ch	DMA0	Channel configuration register 5	DMAC0_CHCFG_5	32	32
A006 2170h	DMA0	Channel interval register 5	DMAC0_CHITVL_5	32	32
A006 2178h	DMA0	Next link address register 5	DMAC0_NXLA_5	32	32
A006 217Ch	DMA0	Current link address register 5	DMAC0_CRLA_5	32	32
A006 2180h	DMA0	Next 0 source address register 6	DMAC0_N0SA_6_N	32	32
A006 2180h	DMA0	Next 0 source address register 6	DMAC0_N0SA_6_W	32	32
A006 2184h	DMA0	Next 0 destination address register 6	DMAC0_N0DA_6	32	32
A006 2188h	DMA0	Next 0 transaction byte register 6	DMAC0_N0TB_6	32	32
A006 218Ch	DMA0	Next 1 source address register 6	DMAC0_N1SA_6_N	32	32
A006 218Ch	DMA0	Next 1 source address register 6	DMAC0_N1SA_6_W	32	32
A006 2190h	DMA0	Next 1 destination address register 6	DMAC0_N1DA_6	32	32
A006 2194h	DMA0	Next 1 transaction byte register 6	DMAC0_N1TB_6	32	32
A006 2198h	DMA0	Current source address register 6	DMAC0_CRSA_6	32	32
A006 219Ch	DMA0	Current destination address register 6	DMAC0_CRDA_6	32	32
A006 21A0h	DMA0	Current transaction byte register 6	DMAC0_CRTB_6	32	32
A006 21A4h	DMA0	Channel status register 6	DMAC0_CHSTAT_6	32	32
A006 21A8h	DMA0	Channel control register 6	DMAC0_CHCTRL_6	32	32
A006 21ACh	DMA0	Channel configuration register 6	DMAC0_CHCFG_6	32	32

Table 5.1 List of I/O Registers (Address Order) (26 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 21B0h	DMA0	Channel interval register 6	DMAC0_CHITVL_6	32	32
A006 21B8h	DMA0	Next link address register 6	DMAC0_NXLA_6	32	32
A006 21BCh	DMA0	Current link address register 6	DMAC0_CRLA_6	32	32
A006 21C0h	DMA0	Next 0 source address register 7	DMAC0_N0SA_7_N	32	32
A006 21C0h	DMA0	Next 0 source address register 7	DMAC0_N0SA_7_W	32	32
A006 21C4h	DMA0	Next 0 destination address register 7	DMAC0_N0DA_7	32	32
A006 21C8h	DMA0	Next 0 transaction byte register 7	DMAC0_N0TB_7	32	32
A006 21CCh	DMA0	Next 1 source address register 7	DMAC0_N1SA_7_N	32	32
A006 21CCh	DMA0	Next 1 source address register 7	DMAC0_N1SA_7_W	32	32
A006 21D0h	DMA0	Next 1 destination address register 7	DMAC0_N1DA_7	32	32
A006 21D4h	DMA0	Next 1 transaction byte register 7	DMAC0_N1TB_7	32	32
A006 21D8h	DMA0	Current source address register 7	DMAC0_CRSA_7	32	32
A006 21DCh	DMA0	Current destination address register 7	DMAC0_CRDA_7	32	32
A006 21E0h	DMA0	Current transaction byte register 7	DMAC0_CRTB_7	32	32
A006 21E4h	DMA0	Channel status register 7	DMAC0_CHSTAT_7	32	32
A006 21E8h	DMA0	Channel control register 7	DMAC0_CHCTRL_7	32	32
A006 21ECh	DMA0	Channel configuration register 7	DMAC0_CHCFG_7	32	32
A006 21F0h	DMA0	Channel interval register 7	DMAC0_CHITVL_7	32	32
A006 21F8h	DMA0	Next link address register 7	DMAC0_NXLA_7	32	32
A006 21FCh	DMA0	Current link address register 7	DMAC0_CRLA_7	32	32
A006 2200h	DMA0	Source continuous register 0	DMAC0_SCNT_0	32	32
A006 2204h	DMA0	Source skip register 0	DMAC0_SSKP_0	32	32
A006 2208h	DMA0	Destination continuous register 0	DMAC0_DCNT_0	32	32
A006 220Ch	DMA0	Destination skip register 0	DMAC0_DSKP_0	32	32
A006 2220h	DMA0	Source continuous register 1	DMAC0_SCNT_1	32	32
A006 2224h	DMA0	Source skip register 1	DMAC0_SSKP_1	32	32
A006 2228h	DMA0	Destination continuous register 1	DMAC0_DCNT_1	32	32
A006 222Ch	DMA0	Destination skip register 1	DMAC0_DSKP_1	32	32
A006 2240h	DMA0	Source continuous register 2	DMAC0_SCNT_2	32	32
A006 2244h	DMA0	Source skip register 2	DMAC0_SSKP_2	32	32
A006 2248h	DMA0	Destination continuous register 2	DMAC0_DCNT_2	32	32
A006 224Ch	DMA0	Destination skip register 2	DMAC0_DSKP_2	32	32
A006 2260h	DMA0	Source continuous register 3	DMAC0_SCNT_3	32	32
A006 2264h	DMA0	Source skip register 3	DMAC0_SSKP_3	32	32
A006 2268h	DMA0	Destination continuous register 3	DMAC0_DCNT_3	32	32
A006 226Ch	DMA0	Destination skip register 3	DMAC0_DSKP_3	32	32
A006 2280h	DMA0	Source continuous register 4	DMAC0_SCNT_4	32	32
A006 2284h	DMA0	Source skip register 4	DMAC0_SSKP_4	32	32
A006 2288h	DMA0	Destination continuous register 4	DMAC0_DCNT_4	32	32
A006 228Ch	DMA0	Destination skip register 4	DMAC0_DSKP_4	32	32
A006 22A0h	DMA0	Source continuous register 5	DMAC0_SCNT_5	32	32
A006 22A4h	DMA0	Source skip register 5	DMAC0_SSKP_5	32	32
A006 22A8h	DMA0	Destination continuous register 5	DMAC0_DCNT_5	32	32
A006 22ACh	DMA0	Destination skip register 5	DMAC0_DSKP_5	32	32
A006 22C0h	DMA0	Source continuous register 6	DMAC0_SCNT_6	32	32
A006 22C4h	DMA0	Source skip register 6	DMAC0_SSKP_6	32	32
A006 22C8h	DMA0	Destination continuous register 6	DMAC0_DCNT_6	32	32
A006 22CCh	DMA0	Destination skip register 6	DMAC0_DSKP_6	32	32

Table 5.1 List of I/O Registers (Address Order) (27 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 22E0h	DMA0	Source continuous register 7	DMAC0_SCNT_7	32	32
A006 22E4h	DMA0	Source skip register 7	DMAC0_SSKP_7	32	32
A006 22E8h	DMA0	Destination continuous register 7	DMAC0_DCNT_7	32	32
A006 22ECh	DMA0	Destination skip register 7	DMAC0_DSKP_7	32	32
A006 2300h	DMA0	DMA control register A	DMAC0_DCTRL_A	32	32
A006 2304h	DMA0	Descriptor interval register A	DMAC0_DSCITVL_A	32	32
A006 2310h	DMA0	DMA status EN register A	DMAC0_DST_EN_A	32	32
A006 2314h	DMA0	DMA status ER register A	DMAC0_DST_ER_A	32	32
A006 2318h	DMA0	DMA status END register A	DMAC0_DST_END_A	32	32
A006 2320h	DMA0	DMA status SUS register A	DMAC0_DST_SUS_A	32	32
A006 2400h	DMA0	Next 0 source address register 8	DMAC0_N0SA_8_N	32	32
A006 2400h	DMA0	Next 0 source address register 8	DMAC0_N0SA_8_W	32	32
A006 2404h	DMA0	Next 0 destination address register 8	DMAC0_N0DA_8	32	32
A006 2408h	DMA0	Next 0 transaction byte register 8	DMAC0_N0TB_8	32	32
A006 240Ch	DMA0	Next 1 source address register 8	DMAC0_N1SA_8_N	32	32
A006 240Ch	DMA0	Next 1 source address register 8	DMAC0_N1SA_8_W	32	32
A006 2410h	DMA0	Next 1 destination address register 8	DMAC0_N1DA_8	32	32
A006 2414h	DMA0	Next 1 transaction byte register 8	DMAC0_N1TB_8	32	32
A006 2418h	DMA0	Current source address register 8	DMAC0_CRSA_8	32	32
A006 241Ch	DMA0	Current destination address register 8	DMAC0_CRDA_8	32	32
A006 2420h	DMA0	Current transaction byte register 8	DMAC0_CRTB_8	32	32
A006 2424h	DMA0	Channel status register 8	DMAC0_CHSTAT_8	32	32
A006 2428h	DMA0	Channel control register 8	DMAC0_CHCTRL_8	32	32
A006 242Ch	DMA0	Channel configuration register 8	DMAC0_CHCFG_8	32	32
A006 2430h	DMA0	Channel interval register 8	DMAC0_CHITVL_8	32	32
A006 2438h	DMA0	Next link address register 8	DMAC0_NXLA_8	32	32
A006 243Ch	DMA0	Current link address register 8	DMAC0_CRLA_8	32	32
A006 2440h	DMA0	Next 0 source address register 9	DMAC0_N0SA_9_N	32	32
A006 2440h	DMA0	Next 0 source address register 9	DMAC0_N0SA_9_W	32	32
A006 2444h	DMA0	Next 0 destination address register 9	DMAC0_N0DA_9	32	32
A006 2448h	DMA0	Next 0 transaction byte register 9	DMAC0_N0TB_9	32	32
A006 244Ch	DMA0	Next 1 source address register 9	DMAC0_N1SA_9_N	32	32
A006 244Ch	DMA0	Next 1 source address register 9	DMAC0_N1SA_9_W	32	32
A006 2450h	DMA0	Next 1 destination address register 9	DMAC0_N1DA_9	32	32
A006 2454h	DMA0	Next 1 transaction byte register 9	DMAC0_N1TB_9	32	32
A006 2458h	DMA0	Current source address register 9	DMAC0_CRSA_9	32	32
A006 245Ch	DMA0	Current destination address register 9	DMAC0_CRDA_9	32	32
A006 2460h	DMA0	Current transaction byte register 9	DMAC0_CRTB_9	32	32
A006 2464h	DMA0	Channel status register 9	DMAC0_CHSTAT_9	32	32
A006 2468h	DMA0	Channel control register 9	DMAC0_CHCTRL_9	32	32
A006 246Ch	DMA0	Channel configuration register 9	DMAC0_CHCFG_9	32	32
A006 2470h	DMA0	Channel interval register 9	DMAC0_CHITVL_9	32	32
A006 2478h	DMA0	Next link address register 9	DMAC0_NXLA_9	32	32
A006 247Ch	DMA0	Current link address register 9	DMAC0_CRLA_9	32	32
A006 2480h	DMA0	Next 0 source address register 10	DMAC0_N0SA_10_N	32	32
A006 2480h	DMA0	Next 0 source address register 10	DMAC0_N0SA_10_W	32	32
A006 2484h	DMA0	Next 0 destination address register 10	DMAC0_N0DA_10	32	32
A006 2488h	DMA0	Next 0 transaction byte register 10	DMAC0_N0TB_10	32	32

Table 5.1 List of I/O Registers (Address Order) (28 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 248Ch	DMA0	Next 1 source address register 10	DMAC0_N1SA_10_N	32	32
A006 248Ch	DMA0	Next 1 source address register 10	DMAC0_N1SA_10_W	32	32
A006 2490h	DMA0	Next 1 destination address register 10	DMAC0_N1DA_10	32	32
A006 2494h	DMA0	Next 1 transaction byte register 10	DMAC0_N1TB_10	32	32
A006 2498h	DMA0	Current source address register 10	DMAC0_CRSA_10	32	32
A006 249Ch	DMA0	Current destination address register 10	DMAC0_CRDA_10	32	32
A006 24A0h	DMA0	Current transaction byte register 10	DMAC0_CRTB_10	32	32
A006 24A4h	DMA0	Channel status register 10	DMAC0_CHSTAT_10	32	32
A006 24A8h	DMA0	Channel control register 10	DMAC0_CHCTRL_10	32	32
A006 24ACh	DMA0	Channel configuration register 10	DMAC0_CHCFG_10	32	32
A006 24B0h	DMA0	Channel interval register 10	DMAC0_CHITVL_10	32	32
A006 24B8h	DMA0	Next link address register 10	DMAC0_NXLA_10	32	32
A006 24BCh	DMA0	Current link address register 10	DMAC0_CRLA_10	32	32
A006 24C0h	DMA0	Next 0 source address register 11	DMAC0_N0SA_11_N	32	32
A006 24C0h	DMA0	Next 0 source address register 11	DMAC0_N0SA_11_W	32	32
A006 24C4h	DMA0	Next 0 destination address register 11	DMAC0_N0DA_11	32	32
A006 24C8h	DMA0	Next 0 transaction byte register 11	DMAC0_N0TB_11	32	32
A006 24CCh	DMA0	Next 1 source address register 11	DMAC0_N1SA_11_N	32	32
A006 24CCh	DMA0	Next 1 source address register 11	DMAC0_N1SA_11_W	32	32
A006 24D0h	DMA0	Next 1 destination address register 11	DMAC0_N1DA_11	32	32
A006 24D4h	DMA0	Next 1 transaction byte register 11	DMAC0_N1TB_11	32	32
A006 24D8h	DMA0	Current source address register 11	DMAC0_CRSA_11	32	32
A006 24DCh	DMA0	Current destination address register 11	DMAC0_CRDA_11	32	32
A006 24E0h	DMA0	Current transaction byte register 11	DMAC0_CRTB_11	32	32
A006 24E4h	DMA0	Channel status register 11	DMAC0_CHSTAT_11	32	32
A006 24E8h	DMA0	Channel control register 11	DMAC0_CHCTRL_11	32	32
A006 24ECh	DMA0	Channel configuration register 11	DMAC0_CHCFG_11	32	32
A006 24F0h	DMA0	Channel interval register 11	DMAC0_CHITVL_11	32	32
A006 24F8h	DMA0	Next link address register 11	DMAC0_NXLA_11	32	32
A006 24FCh	DMA0	Current link address register 11	DMAC0_CRLA_11	32	32
A006 2500h	DMA0	Next 0 source address register 12	DMAC0_N0SA_12_N	32	32
A006 2500h	DMA0	Next 0 source address register 12	DMAC0_N0SA_12_W	32	32
A006 2504h	DMA0	Next 0 destination address register 12	DMAC0_N0DA_12	32	32
A006 2508h	DMA0	Next 0 transaction byte register 12	DMAC0_N0TB_12	32	32
A006 250Ch	DMA0	Next 1 source address register 12	DMAC0_N1SA_12_N	32	32
A006 250Ch	DMA0	Next 1 source address register 12	DMAC0_N1SA_12_W	32	32
A006 2510h	DMA0	Next 1 destination address register 12	DMAC0_N1DA_12	32	32
A006 2514h	DMA0	Next 1 transaction byte register 12	DMAC0_N1TB_12	32	32
A006 2518h	DMA0	Current source address register 12	DMAC0_CRSA_12	32	32
A006 251Ch	DMA0	Current destination address register 12	DMAC0_CRDA_12	32	32
A006 2520h	DMA0	Current transaction byte register 12	DMAC0_CRTB_12	32	32
A006 2524h	DMA0	Channel status register 12	DMAC0_CHSTAT_12	32	32
A006 2528h	DMA0	Channel control register 12	DMAC0_CHCTRL_12	32	32
A006 252Ch	DMA0	Channel configuration register 12	DMAC0_CHCFG_12	32	32
A006 2530h	DMA0	Channel interval register 12	DMAC0_CHITVL_12	32	32
A006 2538h	DMA0	Next link address register 12	DMAC0_NXLA_12	32	32
A006 253Ch	DMA0	Current link address register 12	DMAC0_CRLA_12	32	32
A006 2540h	DMA0	Next 0 source address register 13	DMAC0_N0SA_13_N	32	32

Table 5.1 List of I/O Registers (Address Order) (29 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 2540h	DMA0	Next 0 source address register 13	DMAC0_N0SA_13_W	32	32
A006 2544h	DMA0	Next 0 destination address register 13	DMAC0_N0DA_13	32	32
A006 2548h	DMA0	Next 0 transaction byte register 13	DMAC0_N0TB_13	32	32
A006 254Ch	DMA0	Next 1 source address register 13	DMAC0_N1SA_13_N	32	32
A006 254Ch	DMA0	Next 1 source address register 13	DMAC0_N1SA_13_W	32	32
A006 2550h	DMA0	Next 1 destination address register 13	DMAC0_N1DA_13	32	32
A006 2554h	DMA0	Next 1 transaction byte register 13	DMAC0_N1TB_13	32	32
A006 2558h	DMA0	Current source address register 13	DMAC0_CRSA_13	32	32
A006 255Ch	DMA0	Current destination address register 13	DMAC0_CRDA_13	32	32
A006 2560h	DMA0	Current transaction byte register 13	DMAC0_CRTB_13	32	32
A006 2564h	DMA0	Channel status register 13	DMAC0_CHSTAT_13	32	32
A006 2568h	DMA0	Channel control register 13	DMAC0_CHCTRL_13	32	32
A006 256Ch	DMA0	Channel configuration register 13	DMAC0_CHCFG_13	32	32
A006 2570h	DMA0	Channel interval register 13	DMAC0_CHITVL_13	32	32
A006 2578h	DMA0	Next link address register 13	DMAC0_NXLA_13	32	32
A006 257Ch	DMA0	Current link address register 13	DMAC0_CRLA_13	32	32
A006 2580h	DMA0	Next 0 source address register 14	DMAC0_N0SA_14_N	32	32
A006 2580h	DMA0	Next 0 source address register 14	DMAC0_N0SA_14_W	32	32
A006 2584h	DMA0	Next 0 destination address register 14	DMAC0_N0DA_14	32	32
A006 2588h	DMA0	Next 0 transaction byte register 14	DMAC0_N0TB_14	32	32
A006 258Ch	DMA0	Next 1 source address register 14	DMAC0_N1SA_14_N	32	32
A006 258Ch	DMA0	Next 1 source address register 14	DMAC0_N1SA_14_W	32	32
A006 2590h	DMA0	Next 1 destination address register 14	DMAC0_N1DA_14	32	32
A006 2594h	DMA0	Next 1 transaction byte register 14	DMAC0_N1TB_14	32	32
A006 2598h	DMA0	Current source address register 14	DMAC0_CRSA_14	32	32
A006 259Ch	DMA0	Current destination address register 14	DMAC0_CRDA_14	32	32
A006 25A0h	DMA0	Current transaction byte register 14	DMAC0_CRTB_14	32	32
A006 25A4h	DMA0	Channel status register 14	DMAC0_CHSTAT_14	32	32
A006 25A8h	DMA0	Channel control register 14	DMAC0_CHCTRL_14	32	32
A006 25ACh	DMA0	Channel configuration register 14	DMAC0_CHCFG_14	32	32
A006 25B0h	DMA0	Channel interval register 14	DMAC0_CHITVL_14	32	32
A006 25B8h	DMA0	Next link address register 14	DMAC0_NXLA_14	32	32
A006 25BCh	DMA0	Current link address register 14	DMAC0_CRLA_14	32	32
A006 25C0h	DMA0	Next 0 source address register 15	DMAC0_N0SA_15_N	32	32
A006 25C0h	DMA0	Next 0 source address register 15	DMAC0_N0SA_15_W	32	32
A006 25C4h	DMA0	Next 0 destination address register 15	DMAC0_N0DA_15	32	32
A006 25C8h	DMA0	Next 0 transaction byte register 15	DMAC0_N0TB_15	32	32
A006 25CCh	DMA0	Next 1 source address register 15	DMAC0_N1SA_15_N	32	32
A006 25CCh	DMA0	Next 1 source address register 15	DMAC0_N1SA_15_W	32	32
A006 25D0h	DMA0	Next 1 destination address register 15	DMAC0_N1DA_15	32	32
A006 25D4h	DMA0	Next 1 transaction byte register 15	DMAC0_N1TB_15	32	32
A006 25D8h	DMA0	Current source address register 15	DMAC0_CRSA_15	32	32
A006 25DCh	DMA0	Current destination address register 15	DMAC0_CRDA_15	32	32
A006 25E0h	DMA0	Current transaction byte register 15	DMAC0_CRTB_15	32	32
A006 25E4h	DMA0	Channel status register 15	DMAC0_CHSTAT_15	32	32
A006 25E8h	DMA0	Channel control register 15	DMAC0_CHCTRL_15	32	32
A006 25ECh	DMA0	Channel configuration register 15	DMAC0_CHCFG_15	32	32
A006 25F0h	DMA0	Channel interval register 15	DMAC0_CHITVL_15	32	32

Table 5.1 List of I/O Registers (Address Order) (30 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 25F8h	DMA0	Next link address register 15	DMAC0_NXLA_15	32	32
A006 25FCh	DMA0	Current link address register 15	DMAC0_CRLA_15	32	32
A006 2600h	DMA0	Source continuous register 8	DMAC0_SCNT_8	32	32
A006 2604h	DMA0	Source skip register 8	DMAC0_SSKP_8	32	32
A006 2608h	DMA0	Destination continuous register 8	DMAC0_DCNT_8	32	32
A006 260Ch	DMA0	Destination skip register 8	DMAC0_DSKP_8	32	32
A006 2620h	DMA0	Source continuous register 9	DMAC0_SCNT_9	32	32
A006 2624h	DMA0	Source skip register 9	DMAC0_SSKP_9	32	32
A006 2628h	DMA0	Destination continuous register 9	DMAC0_DCNT_9	32	32
A006 262Ch	DMA0	Destination skip register 9	DMAC0_DSKP_9	32	32
A006 2640h	DMA0	Source continuous register 10	DMAC0_SCNT_10	32	32
A006 2644h	DMA0	Source skip register 10	DMAC0_SSKP_10	32	32
A006 2648h	DMA0	Destination continuous register 10	DMAC0_DCNT_10	32	32
A006 264Ch	DMA0	Destination skip register 10	DMAC0_DSKP_10	32	32
A006 2660h	DMA0	Source continuous register 11	DMAC0_SCNT_11	32	32
A006 2664h	DMA0	Source skip register 11	DMAC0_SSKP_11	32	32
A006 2668h	DMA0	Destination continuous register 11	DMAC0_DCNT_11	32	32
A006 266Ch	DMA0	Destination skip register 11	DMAC0_DSKP_11	32	32
A006 2680h	DMA0	Source continuous register 12	DMAC0_SCNT_12	32	32
A006 2684h	DMA0	Source skip register 12	DMAC0_SSKP_12	32	32
A006 2688h	DMA0	Destination continuous register 12	DMAC0_DCNT_12	32	32
A006 268Ch	DMA0	Destination skip register 12	DMAC0_DSKP_12	32	32
A006 26A0h	DMA0	Source continuous register 13	DMAC0_SCNT_13	32	32
A006 26A4h	DMA0	Source skip register 13	DMAC0_SSKP_13	32	32
A006 26A8h	DMA0	Destination continuous register 13	DMAC0_DCNT_13	32	32
A006 26ACh	DMA0	Destination skip register 13	DMAC0_DSKP_13	32	32
A006 26C0h	DMA0	Source continuous register 14	DMAC0_SCNT_14	32	32
A006 26C4h	DMA0	Source skip register 14	DMAC0_SSKP_14	32	32
A006 26C8h	DMA0	Destination continuous register 14	DMAC0_DCNT_14	32	32
A006 26CCh	DMA0	Destination skip register 14	DMAC0_DSKP_14	32	32
A006 26E0h	DMA0	Source continuous register 15	DMAC0_SCNT_15	32	32
A006 26E4h	DMA0	Source skip register 15	DMAC0_SSKP_15	32	32
A006 26E8h	DMA0	Destination continuous register 15	DMAC0_DCNT_15	32	32
A006 26ECh	DMA0	Destination skip register 15	DMAC0_DSKP_15	32	32
A006 2700h	DMA0	DMA control register B	DMAC0_DCTRL_B	32	32
A006 2704h	DMA0	Descriptor interval register B	DMAC0_DSCITVL_B	32	32
A006 2710h	DMA0	DMA status EN register B	DMAC0_DST_EN_B	32	32
A006 2714h	DMA0	DMA status ER register B	DMAC0_DST_ER_B	32	32
A006 2718h	DMA0	DMA status END register B	DMAC0_DST_END_B	32	32
A006 2720h	DMA0	DMA status SUS register B	DMAC0_DST_SUS_B	32	32
A006 3000h	DMA1	Next 0 source address register 0	DMAC1_N0SA_0_N	32	32
A006 3000h	DMA1	Next 0 source address register 0	DMAC1_N0SA_0_W	32	32
A006 3004h	DMA1	Next 0 destination address register 0	DMAC1_N0DA_0	32	32
A006 3008h	DMA1	Next 0 transaction byte register 0	DMAC1_N0TB_0	32	32
A006 300Ch	DMA1	Next 1 source address register 0	DMAC1_N1SA_0_N	32	32
A006 300Ch	DMA1	Next 1 source address register 0	DMAC1_N1SA_0_W	32	32
A006 3010h	DMA1	Next 1 destination address register 0	DMAC1_N1DA_0	32	32
A006 3014h	DMA1	Next 1 transaction byte register 0	DMAC1_N1TB_0	32	32

Table 5.1 List of I/O Registers (Address Order) (31 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3018h	DMA1	Current source address register 0	DMAC1_CRSA_0	32	32
A006 301Ch	DMA1	Current destination address register 0	DMAC1_CRDA_0	32	32
A006 3020h	DMA1	Current transaction byte register 0	DMAC1_CRTB_0	32	32
A006 3024h	DMA1	Channel status register 0	DMAC1_CHSTAT_0	32	32
A006 3028h	DMA1	Channel control register 0	DMAC1_CHCTRL_0	32	32
A006 302Ch	DMA1	Channel configuration register 0	DMAC1_CHCFG_0	32	32
A006 3030h	DMA1	Channel interval register 0	DMAC1_CHITVL_0	32	32
A006 3038h	DMA1	Next link address register 0	DMAC1_NXLA_0	32	32
A006 303Ch	DMA1	Current link address register 0	DMAC1_CRLA_0	32	32
A006 3040h	DMA1	Next 0 source address register 1	DMAC1_N0SA_1_N	32	32
A006 3040h	DMA1	Next 0 source address register 1	DMAC1_N0SA_1_W	32	32
A006 3044h	DMA1	Next 0 destination address register 1	DMAC1_N0DA_1	32	32
A006 3048h	DMA1	Next 0 transaction byte register 1	DMAC1_N0TB_1	32	32
A006 304Ch	DMA1	Next 1 source address register 1	DMAC1_N1SA_1_N	32	32
A006 304Ch	DMA1	Next 1 source address register 1	DMAC1_N1SA_1_W	32	32
A006 3050h	DMA1	Next 1 destination address register 1	DMAC1_N1DA_1	32	32
A006 3054h	DMA1	Next 1 transaction byte register 1	DMAC1_N1TB_1	32	32
A006 3058h	DMA1	Current source address register 1	DMAC1_CRSA_1	32	32
A006 305Ch	DMA1	Current destination address register 1	DMAC1_CRDA_1	32	32
A006 3060h	DMA1	Current transaction byte register 1	DMAC1_CRTB_1	32	32
A006 3064h	DMA1	Channel status register 1	DMAC1_CHSTAT_1	32	32
A006 3068h	DMA1	Channel control register 1	DMAC1_CHCTRL_1	32	32
A006 306Ch	DMA1	Channel configuration register 1	DMAC1_CHCFG_1	32	32
A006 3070h	DMA1	Channel interval register 1	DMAC1_CHITVL_1	32	32
A006 3078h	DMA1	Next link address register 1	DMAC1_NXLA_1	32	32
A006 307Ch	DMA1	Current link address register 1	DMAC1_CRLA_1	32	32
A006 3080h	DMA1	Next 0 source address register 2	DMAC1_N0SA_2_N	32	32
A006 3080h	DMA1	Next 0 source address register 2	DMAC1_N0SA_2_W	32	32
A006 3084h	DMA1	Next 0 destination address register 2	DMAC1_N0DA_2	32	32
A006 3088h	DMA1	Next 0 transaction byte register 2	DMAC1_N0TB_2	32	32
A006 308Ch	DMA1	Next 1 source address register 2	DMAC1_N1SA_2_N	32	32
A006 308Ch	DMA1	Next 1 source address register 2	DMAC1_N1SA_2_W	32	32
A006 3090h	DMA1	Next 1 destination address register 2	DMAC1_N1DA_2	32	32
A006 3094h	DMA1	Next 1 transaction byte register 2	DMAC1_N1TB_2	32	32
A006 3098h	DMA1	Current source address register 2	DMAC1_CRSA_2	32	32
A006 309Ch	DMA1	Current destination address register 2	DMAC1_CRDA_2	32	32
A006 30A0h	DMA1	Current transaction byte register 2	DMAC1_CRTB_2	32	32
A006 30A4h	DMA1	Channel status register 2	DMAC1_CHSTAT_2	32	32
A006 30A8h	DMA1	Channel control register 2	DMAC1_CHCTRL_2	32	32
A006 30ACh	DMA1	Channel configuration register 2	DMAC1_CHCFG_2	32	32
A006 30B0h	DMA1	Channel interval register 2	DMAC1_CHITVL_2	32	32
A006 30B8h	DMA1	Next link address register 2	DMAC1_NXLA_2	32	32
A006 30BCh	DMA1	Current link address register 2	DMAC1_CRLA_2	32	32
A006 30C0h	DMA1	Next 0 source address register 3	DMAC1_N0SA_3_N	32	32
A006 30C0h	DMA1	Next 0 source address register 3	DMAC1_N0SA_3_W	32	32
A006 30C4h	DMA1	Next 0 destination address register 3	DMAC1_N0DA_3	32	32
A006 30C8h	DMA1	Next 0 transaction byte register 3	DMAC1_N0TB_3	32	32
A006 30CCh	DMA1	Next 1 source address register 3	DMAC1_N1SA_3_N	32	32

Table 5.1 List of I/O Registers (Address Order) (32 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 30CCh	DMA1	Next 1 source address register 3	DMAC1_N1SA_3_W	32	32
A006 30D0h	DMA1	Next 1 destination address register 3	DMAC1_N1DA_3	32	32
A006 30D4h	DMA1	Next 1 transaction byte register 3	DMAC1_N1TB_3	32	32
A006 30D8h	DMA1	Current source address register 3	DMAC1_CRSA_3	32	32
A006 30DCh	DMA1	Current destination address register 3	DMAC1_CRDA_3	32	32
A006 30E0h	DMA1	Current transaction byte register 3	DMAC1_CRTB_3	32	32
A006 30E4h	DMA1	Channel status register 3	DMAC1_CHSTAT_3	32	32
A006 30E8h	DMA1	Channel control register 3	DMAC1_CHCTRL_3	32	32
A006 30ECh	DMA1	Channel configuration register 3	DMAC1_CHCFG_3	32	32
A006 30F0h	DMA1	Channel interval register 3	DMAC1_CHITVL_3	32	32
A006 30F8h	DMA1	Next link address register 3	DMAC1_NXLA_3	32	32
A006 30FCh	DMA1	Current link address register 3	DMAC1_CRLA_3	32	32
A006 3100h	DMA1	Next 0 source address register 4	DMAC1_N0SA_4_N	32	32
A006 3100h	DMA1	Next 0 source address register 4	DMAC1_N0SA_4_W	32	32
A006 3104h	DMA1	Next 0 destination address register 4	DMAC1_N0DA_4	32	32
A006 3108h	DMA1	Next 0 transaction byte register 4	DMAC1_N0TB_4	32	32
A006 310Ch	DMA1	Next 1 source address register 4	DMAC1_N1SA_4_N	32	32
A006 310Ch	DMA1	Next 1 source address register 4	DMAC1_N1SA_4_W	32	32
A006 3110h	DMA1	Next 1 destination address register 4	DMAC1_N1DA_4	32	32
A006 3114h	DMA1	Next 1 transaction byte register 4	DMAC1_N1TB_4	32	32
A006 3118h	DMA1	Current source address register 4	DMAC1_CRSA_4	32	32
A006 311Ch	DMA1	Current destination address register 4	DMAC1_CRDA_4	32	32
A006 3120h	DMA1	Current transaction byte register 4	DMAC1_CRTB_4	32	32
A006 3124h	DMA1	Channel status register 4	DMAC1_CHSTAT_4	32	32
A006 3128h	DMA1	Channel control register 4	DMAC1_CHCTRL_4	32	32
A006 312Ch	DMA1	Channel configuration register 4	DMAC1_CHCFG_4	32	32
A006 3130h	DMA1	Channel interval register 4	DMAC1_CHITVL_4	32	32
A006 3138h	DMA1	Next link address register 4	DMAC1_NXLA_4	32	32
A006 313Ch	DMA1	Current link address register 4	DMAC1_CRLA_4	32	32
A006 3140h	DMA1	Next 0 source address register 5	DMAC1_N0SA_5_N	32	32
A006 3140h	DMA1	Next 0 source address register 5	DMAC1_N0SA_5_W	32	32
A006 3144h	DMA1	Next 0 destination address register 5	DMAC1_N0DA_5	32	32
A006 3148h	DMA1	Next 0 transaction byte register 5	DMAC1_N0TB_5	32	32
A006 314Ch	DMA1	Next 1 source address register 5	DMAC1_N1SA_5_N	32	32
A006 314Ch	DMA1	Next 1 source address register 5	DMAC1_N1SA_5_W	32	32
A006 3150h	DMA1	Next 1 destination address register 5	DMAC1_N1DA_5	32	32
A006 3154h	DMA1	Next 1 transaction byte register 5	DMAC1_N1TB_5	32	32
A006 3158h	DMA1	Current source address register 5	DMAC1_CRSA_5	32	32
A006 315Ch	DMA1	Current destination address register 5	DMAC1_CRDA_5	32	32
A006 3160h	DMA1	Current transaction byte register 5	DMAC1_CRTB_5	32	32
A006 3164h	DMA1	Channel status register 5	DMAC1_CHSTAT_5	32	32
A006 3168h	DMA1	Channel control register 5	DMAC1_CHCTRL_5	32	32
A006 316Ch	DMA1	Channel configuration register 5	DMAC1_CHCFG_5	32	32
A006 3170h	DMA1	Channel interval register 5	DMAC1_CHITVL_5	32	32
A006 3178h	DMA1	Next link address register 5	DMAC1_NXLA_5	32	32
A006 317Ch	DMA1	Current link address register 5	DMAC1_CRLA_5	32	32
A006 3180h	DMA1	Next 0 source address register 6	DMAC1_N0SA_6_N	32	32
A006 3180h	DMA1	Next 0 source address register 6	DMAC1_N0SA_6_W	32	32

Table 5.1 List of I/O Registers (Address Order) (33 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3184h	DMA1	Next 0 destination address register 6	DMAC1_N0DA_6	32	32
A006 3188h	DMA1	Next 0 transaction byte register 6	DMAC1_N0TB_6	32	32
A006 318Ch	DMA1	Next 1 source address register 6	DMAC1_N1SA_6_N	32	32
A006 318Ch	DMA1	Next 1 source address register 6	DMAC1_N1SA_6_W	32	32
A006 3190h	DMA1	Next 1 destination address register 6	DMAC1_N1DA_6	32	32
A006 3194h	DMA1	Next 1 transaction byte register 6	DMAC1_N1TB_6	32	32
A006 3198h	DMA1	Current source address register 6	DMAC1_CRSA_6	32	32
A006 319Ch	DMA1	Current destination address register 6	DMAC1_CRDA_6	32	32
A006 31A0h	DMA1	Current transaction byte register 6	DMAC1_CRTB_6	32	32
A006 31A4h	DMA1	Channel status register 6	DMAC1_CHSTAT_6	32	32
A006 31A8h	DMA1	Channel control register 6	DMAC1_CHCTRL_6	32	32
A006 31ACh	DMA1	Channel configuration register 6	DMAC1_CHCFG_6	32	32
A006 31B0h	DMA1	Channel interval register 6	DMAC1_CHITVL_6	32	32
A006 31B8h	DMA1	Next link address register 6	DMAC1_NXLA_6	32	32
A006 31BCh	DMA1	Current link address register 6	DMAC1_CRLA_6	32	32
A006 31C0h	DMA1	Next 0 source address register 7	DMAC1_N0SA_7_N	32	32
A006 31C0h	DMA1	Next 0 source address register 7	DMAC1_N0SA_7_W	32	32
A006 31C4h	DMA1	Next 0 destination address register 7	DMAC1_N0DA_7	32	32
A006 31C8h	DMA1	Next 0 transaction byte register 7	DMAC1_N0TB_7	32	32
A006 31CCh	DMA1	Next 1 source address register 7	DMAC1_N1SA_7_N	32	32
A006 31CCh	DMA1	Next 1 source address register 7	DMAC1_N1SA_7_W	32	32
A006 31D0h	DMA1	Next 1 destination address register 7	DMAC1_N1DA_7	32	32
A006 31D4h	DMA1	Next 1 transaction byte register 7	DMAC1_N1TB_7	32	32
A006 31D8h	DMA1	Current source address register 7	DMAC1_CRSA_7	32	32
A006 31DCh	DMA1	Current destination address register 7	DMAC1_CRDA_7	32	32
A006 31E0h	DMA1	Current transaction byte register 7	DMAC1_CRTB_7	32	32
A006 31E4h	DMA1	Channel status register 7	DMAC1_CHSTAT_7	32	32
A006 31E8h	DMA1	Channel control register 7	DMAC1_CHCTRL_7	32	32
A006 31ECh	DMA1	Channel configuration register 7	DMAC1_CHCFG_7	32	32
A006 31F0h	DMA1	Channel interval register 7	DMAC1_CHITVL_7	32	32
A006 31F8h	DMA1	Next link address register 7	DMAC1_NXLA_7	32	32
A006 31FCh	DMA1	Current link address register 7	DMAC1_CRLA_7	32	32
A006 3200h	DMA1	Source continuous register 0	DMAC1_SCNT_0	32	32
A006 3204h	DMA1	Source skip register 0	DMAC1_SSKP_0	32	32
A006 3208h	DMA1	Destination continuous register 0	DMAC1_DCNT_0	32	32
A006 320Ch	DMA1	Destination skip register 0	DMAC1_DSKP_0	32	32
A006 3220h	DMA1	Source continuous register 1	DMAC1_SCNT_1	32	32
A006 3224h	DMA1	Source skip register 1	DMAC1_SSKP_1	32	32
A006 3228h	DMA1	Destination continuous register 1	DMAC1_DCNT_1	32	32
A006 322Ch	DMA1	Destination skip register 1	DMAC1_DSKP_1	32	32
A006 3240h	DMA1	Source continuous register 2	DMAC1_SCNT_2	32	32
A006 3244h	DMA1	Source skip register 2	DMAC1_SSKP_2	32	32
A006 3248h	DMA1	Destination continuous register 2	DMAC1_DCNT_2	32	32
A006 324Ch	DMA1	Destination skip register 2	DMAC1_DSKP_2	32	32
A006 3260h	DMA1	Source continuous register 3	DMAC1_SCNT_3	32	32
A006 3264h	DMA1	Source skip register 3	DMAC1_SSKP_3	32	32
A006 3268h	DMA1	Destination continuous register 3	DMAC1_DCNT_3	32	32
A006 326Ch	DMA1	Destination skip register 3	DMAC1_DSKP_3	32	32

Table 5.1 List of I/O Registers (Address Order) (34 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3280h	DMA1	Source continuous register 4	DMAC1_SCNT_4	32	32
A006 3284h	DMA1	Source skip register 4	DMAC1_SSKP_4	32	32
A006 3288h	DMA1	Destination continuous register 4	DMAC1_DCNT_4	32	32
A006 328Ch	DMA1	Destination skip register 4	DMAC1_DSKP_4	32	32
A006 32A0h	DMA1	Source continuous register 5	DMAC1_SCNT_5	32	32
A006 32A4h	DMA1	Source skip register 5	DMAC1_SSKP_5	32	32
A006 32A8h	DMA1	Destination continuous register 5	DMAC1_DCNT_5	32	32
A006 32ACh	DMA1	Destination skip register 5	DMAC1_DSKP_5	32	32
A006 32C0h	DMA1	Source continuous register 6	DMAC1_SCNT_6	32	32
A006 32C4h	DMA1	Source skip register 6	DMAC1_SSKP_6	32	32
A006 32C8h	DMA1	Destination continuous register 6	DMAC1_DCNT_6	32	32
A006 32CCh	DMA1	Destination skip register 6	DMAC1_DSKP_6	32	32
A006 32E0h	DMA1	Source continuous register 7	DMAC1_SCNT_7	32	32
A006 32E4h	DMA1	Source skip register 7	DMAC1_SSKP_7	32	32
A006 32E8h	DMA1	Destination continuous register 7	DMAC1_DCNT_7	32	32
A006 32ECh	DMA1	Destination skip register 7	DMAC1_DSKP_7	32	32
A006 3300h	DMA1	DMA control register A	DMAC1_DCTRL_A	32	32
A006 3304h	DMA1	Descriptor interval register A	DMAC1_DSCITVL_A	32	32
A006 3310h	DMA1	DMA status EN register A	DMAC1_DST_EN_A	32	32
A006 3314h	DMA1	DMA status ER register A	DMAC1_DST_ER_A	32	32
A006 3318h	DMA1	DMA status END register A	DMAC1_DST_END_A	32	32
A006 3320h	DMA1	DMA status SUS register A	DMAC1_DST_SUS_A	32	32
A006 3400h	DMA1	Next 0 source address register 8	DMAC1_N0SA_8_N	32	32
A006 3400h	DMA1	Next 0 source address register 8	DMAC1_N0SA_8_W	32	32
A006 3404h	DMA1	Next 0 destination address register 8	DMAC1_N0DA_8	32	32
A006 3408h	DMA1	Next 0 transaction byte register 8	DMAC1_N0TB_8	32	32
A006 340Ch	DMA1	Next 1 source address register 8	DMAC1_N1SA_8_N	32	32
A006 340Ch	DMA1	Next 1 source address register 8	DMAC1_N1SA_8_W	32	32
A006 3410h	DMA1	Next 1 destination address register 8	DMAC1_N1DA_8	32	32
A006 3414h	DMA1	Next 1 transaction byte register 8	DMAC1_N1TB_8	32	32
A006 3418h	DMA1	Current source address register 8	DMAC1_CRSA_8	32	32
A006 341Ch	DMA1	Current destination address register 8	DMAC1_CRDA_8	32	32
A006 3420h	DMA1	Current transaction byte register 8	DMAC1_CRTB_8	32	32
A006 3424h	DMA1	Channel status register 8	DMAC1_CHSTAT_8	32	32
A006 3428h	DMA1	Channel control register 8	DMAC1_CHCTRL_8	32	32
A006 342Ch	DMA1	Channel configuration register 8	DMAC1_CHCFG_8	32	32
A006 3430h	DMA1	Channel interval register 8	DMAC1_CHITVL_8	32	32
A006 3438h	DMA1	Next link address register 8	DMAC1_NXLA_8	32	32
A006 343Ch	DMA1	Current link address register 8	DMAC1_CRLA_8	32	32
A006 3440h	DMA1	Next 0 source address register 9	DMAC1_N0SA_9_N	32	32
A006 3440h	DMA1	Next 0 source address register 9	DMAC1_N0SA_9_W	32	32
A006 3444h	DMA1	Next 0 destination address register 9	DMAC1_N0DA_9	32	32
A006 3448h	DMA1	Next 0 transaction byte register 9	DMAC1_N0TB_9	32	32
A006 344Ch	DMA1	Next 1 source address register 9	DMAC1_N1SA_9_N	32	32
A006 344Ch	DMA1	Next 1 source address register 9	DMAC1_N1SA_9_W	32	32
A006 3450h	DMA1	Next 1 destination address register 9	DMAC1_N1DA_9	32	32
A006 3454h	DMA1	Next 1 transaction byte register 9	DMAC1_N1TB_9	32	32
A006 3458h	DMA1	Current source address register 9	DMAC1_CRSA_9	32	32

Table 5.1 List of I/O Registers (Address Order) (35 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 345Ch	DMA1	Current destination address register 9	DMAC1_CRDA_9	32	32
A006 3460h	DMA1	Current transaction byte register 9	DMAC1_CRTB_9	32	32
A006 3464h	DMA1	Channel status register 9	DMAC1_CHSTAT_9	32	32
A006 3468h	DMA1	Channel control register 9	DMAC1_CHCTRL_9	32	32
A006 346Ch	DMA1	Channel configuration register 9	DMAC1_CHCFG_9	32	32
A006 3470h	DMA1	Channel interval register 9	DMAC1_CHITVL_9	32	32
A006 3478h	DMA1	Next link address register 9	DMAC1_NXLA_9	32	32
A006 347Ch	DMA1	Current link address register 9	DMAC1_CRLA_9	32	32
A006 3480h	DMA1	Next 0 source address register 10	DMAC1_N0SA_10_N	32	32
A006 3480h	DMA1	Next 0 source address register 10	DMAC1_N0SA_10_W	32	32
A006 3484h	DMA1	Next 0 destination address register 10	DMAC1_N0DA_10	32	32
A006 3488h	DMA1	Next 0 transaction byte register 10	DMAC1_N0TB_10	32	32
A006 348Ch	DMA1	Next 1 source address register 10	DMAC1_N1SA_10_N	32	32
A006 348Ch	DMA1	Next 1 source address register 10	DMAC1_N1SA_10_W	32	32
A006 3490h	DMA1	Next 1 destination address register 10	DMAC1_N1DA_10	32	32
A006 3494h	DMA1	Next 1 transaction byte register 10	DMAC1_N1TB_10	32	32
A006 3498h	DMA1	Current source address register 10	DMAC1_CRSA_10	32	32
A006 349Ch	DMA1	Current destination address register 10	DMAC1_CRDA_10	32	32
A006 34A0h	DMA1	Current transaction byte register 10	DMAC1_CRTB_10	32	32
A006 34A4h	DMA1	Channel status register 10	DMAC1_CHSTAT_10	32	32
A006 34A8h	DMA1	Channel control register 10	DMAC1_CHCTRL_10	32	32
A006 34ACh	DMA1	Channel configuration register 10	DMAC1_CHCFG_10	32	32
A006 34B0h	DMA1	Channel interval register 10	DMAC1_CHITVL_10	32	32
A006 34B8h	DMA1	Next link address register 10	DMAC1_NXLA_10	32	32
A006 34BCh	DMA1	Current link address register 10	DMAC1_CRLA_10	32	32
A006 34C0h	DMA1	Next 0 source address register 11	DMAC1_N0SA_11_N	32	32
A006 34C0h	DMA1	Next 0 source address register 11	DMAC1_N0SA_11_W	32	32
A006 34C4h	DMA1	Next 0 destination address register 11	DMAC1_N0DA_11	32	32
A006 34C8h	DMA1	Next 0 transaction byte register 11	DMAC1_N0TB_11	32	32
A006 34CCh	DMA1	Next 1 source address register 11	DMAC1_N1SA_11_N	32	32
A006 34CCh	DMA1	Next 1 source address register 11	DMAC1_N1SA_11_W	32	32
A006 34D0h	DMA1	Next 1 destination address register 11	DMAC1_N1DA_11	32	32
A006 34D4h	DMA1	Next 1 transaction byte register 11	DMAC1_N1TB_11	32	32
A006 34D8h	DMA1	Current source address register 11	DMAC1_CRSA_11	32	32
A006 34DCh	DMA1	Current destination address register 11	DMAC1_CRDA_11	32	32
A006 34E0h	DMA1	Current transaction byte register 11	DMAC1_CRTB_11	32	32
A006 34E4h	DMA1	Channel status register 11	DMAC1_CHSTAT_11	32	32
A006 34E8h	DMA1	Channel control register 11	DMAC1_CHCTRL_11	32	32
A006 34ECh	DMA1	Channel configuration register 11	DMAC1_CHCFG_11	32	32
A006 34F0h	DMA1	Channel interval register 11	DMAC1_CHITVL_11	32	32
A006 34F8h	DMA1	Next link address register 11	DMAC1_NXLA_11	32	32
A006 34FCh	DMA1	Current link address register 11	DMAC1_CRLA_11	32	32
A006 3500h	DMA1	Next 0 source address register 12	DMAC1_N0SA_12_N	32	32
A006 3500h	DMA1	Next 0 source address register 12	DMAC1_N0SA_12_W	32	32
A006 3504h	DMA1	Next 0 destination address register 12	DMAC1_N0DA_12	32	32
A006 3508h	DMA1	Next 0 transaction byte register 12	DMAC1_N0TB_12	32	32
A006 350Ch	DMA1	Next 1 source address register 12	DMAC1_N1SA_12_N	32	32
A006 350Ch	DMA1	Next 1 source address register 12	DMAC1_N1SA_12_W	32	32

Table 5.1 List of I/O Registers (Address Order) (36 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3510h	DMA1	Next 1 destination address register 12	DMAC1_N1DA_12	32	32
A006 3514h	DMA1	Next 1 transaction byte register 12	DMAC1_N1TB_12	32	32
A006 3518h	DMA1	Current source address register 12	DMAC1_CRSA_12	32	32
A006 351Ch	DMA1	Current destination address register 12	DMAC1_CRDA_12	32	32
A006 3520h	DMA1	Current transaction byte register 12	DMAC1_CRTB_12	32	32
A006 3524h	DMA1	Channel status register 12	DMAC1_CHSTAT_12	32	32
A006 3528h	DMA1	Channel control register 12	DMAC1_CHCTRL_12	32	32
A006 352Ch	DMA1	Channel configuration register 12	DMAC1_CHCFG_12	32	32
A006 3530h	DMA1	Channel interval register 12	DMAC1_CHITVL_12	32	32
A006 3538h	DMA1	Next link address register 12	DMAC1_NXLA_12	32	32
A006 353Ch	DMA1	Current link address register 12	DMAC1_CRLA_12	32	32
A006 3540h	DMA1	Next 0 source address register 13	DMAC1_N0SA_13_N	32	32
A006 3540h	DMA1	Next 0 source address register 13	DMAC1_N0SA_13_W	32	32
A006 3544h	DMA1	Next 0 destination address register 13	DMAC1_N0DA_13	32	32
A006 3548h	DMA1	Next 0 transaction byte register 13	DMAC1_N0TB_13	32	32
A006 354Ch	DMA1	Next 1 source address register 13	DMAC1_N1SA_13_N	32	32
A006 354Ch	DMA1	Next 1 source address register 13	DMAC1_N1SA_13_W	32	32
A006 3550h	DMA1	Next 1 destination address register 13	DMAC1_N1DA_13	32	32
A006 3554h	DMA1	Next 1 transaction byte register 13	DMAC1_N1TB_13	32	32
A006 3558h	DMA1	Current source address register 13	DMAC1_CRSA_13	32	32
A006 355Ch	DMA1	Current destination address register 13	DMAC1_CRDA_13	32	32
A006 3560h	DMA1	Current transaction byte register 13	DMAC1_CRTB_13	32	32
A006 3564h	DMA1	Channel status register 13	DMAC1_CHSTAT_13	32	32
A006 3568h	DMA1	Channel control register 13	DMAC1_CHCTRL_13	32	32
A006 356Ch	DMA1	Channel configuration register 13	DMAC1_CHCFG_13	32	32
A006 3570h	DMA1	Channel interval register 13	DMAC1_CHITVL_13	32	32
A006 3578h	DMA1	Next link address register 13	DMAC1_NXLA_13	32	32
A006 357Ch	DMA1	Current link address register 13	DMAC1_CRLA_13	32	32
A006 3580h	DMA1	Next 0 source address register 14	DMAC1_N0SA_14_N	32	32
A006 3580h	DMA1	Next 0 source address register 14	DMAC1_N0SA_14_W	32	32
A006 3584h	DMA1	Next 0 destination address register 14	DMAC1_N0DA_14	32	32
A006 3588h	DMA1	Next 0 transaction byte register 14	DMAC1_N0TB_14	32	32
A006 358Ch	DMA1	Next 1 source address register 14	DMAC1_N1SA_14_N	32	32
A006 358Ch	DMA1	Next 1 source address register 14	DMAC1_N1SA_14_W	32	32
A006 3590h	DMA1	Next 1 destination address register 14	DMAC1_N1DA_14	32	32
A006 3594h	DMA1	Next 1 transaction byte register 14	DMAC1_N1TB_14	32	32
A006 3598h	DMA1	Current source address register 14	DMAC1_CRSA_14	32	32
A006 359Ch	DMA1	Current destination address register 14	DMAC1_CRDA_14	32	32
A006 35A0h	DMA1	Current transaction byte register 14	DMAC1_CRTB_14	32	32
A006 35A4h	DMA1	Channel status register 14	DMAC1_CHSTAT_14	32	32
A006 35A8h	DMA1	Channel control register 14	DMAC1_CHCTRL_14	32	32
A006 35ACh	DMA1	Channel configuration register 14	DMAC1_CHCFG_14	32	32
A006 35B0h	DMA1	Channel interval register 14	DMAC1_CHITVL_14	32	32
A006 35B8h	DMA1	Next link address register 14	DMAC1_NXLA_14	32	32
A006 35BCh	DMA1	Current link address register 14	DMAC1_CRLA_14	32	32
A006 35C0h	DMA1	Next 0 source address register 15	DMAC1_N0SA_15_N	32	32
A006 35C0h	DMA1	Next 0 source address register 15	DMAC1_N0SA_15_W	32	32
A006 35C4h	DMA1	Next 0 destination address register 15	DMAC1_N0DA_15	32	32

Table 5.1 List of I/O Registers (Address Order) (37 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 35C8h	DMA1	Next 0 transaction byte register 15	DMAC1_N0TB_15	32	32
A006 35CCh	DMA1	Next 1 source address register 15	DMAC1_N1SA_15_N	32	32
A006 35CCh	DMA1	Next 1 source address register 15	DMAC1_N1SA_15_W	32	32
A006 35D0h	DMA1	Next 1 destination address register 15	DMAC1_N1DA_15	32	32
A006 35D4h	DMA1	Next 1 transaction byte register 15	DMAC1_N1TB_15	32	32
A006 35D8h	DMA1	Current source address register 15	DMAC1_CRSA_15	32	32
A006 35DCh	DMA1	Current destination address register 15	DMAC1_CRDA_15	32	32
A006 35E0h	DMA1	Current transaction byte register 15	DMAC1_CRTB_15	32	32
A006 35E4h	DMA1	Channel status register 15	DMAC1_CHSTAT_15	32	32
A006 35E8h	DMA1	Channel control register 15	DMAC1_CHCTRL_15	32	32
A006 35ECh	DMA1	Channel configuration register 15	DMAC1_CHCFG_15	32	32
A006 35F0h	DMA1	Channel interval register 15	DMAC1_CHITVL_15	32	32
A006 35F8h	DMA1	Next link address register 15	DMAC1_NXLA_15	32	32
A006 35FCh	DMA1	Current link address register 15	DMAC1_CRLA_15	32	32
A006 3600h	DMA1	Source continuous register 8	DMAC1_SCNT_8	32	32
A006 3604h	DMA1	Source skip register 8	DMAC1_SSKP_8	32	32
A006 3608h	DMA1	Destination continuous register 8	DMAC1_DCNT_8	32	32
A006 360Ch	DMA1	Destination skip register 8	DMAC1_DSKP_8	32	32
A006 3620h	DMA1	Source continuous register 9	DMAC1_SCNT_9	32	32
A006 3624h	DMA1	Source skip register 9	DMAC1_SSKP_9	32	32
A006 3628h	DMA1	Destination continuous register 9	DMAC1_DCNT_9	32	32
A006 362Ch	DMA1	Destination skip register 9	DMAC1_DSKP_9	32	32
A006 3640h	DMA1	Source continuous register 10	DMAC1_SCNT_10	32	32
A006 3644h	DMA1	Source skip register 10	DMAC1_SSKP_10	32	32
A006 3648h	DMA1	Destination continuous register 10	DMAC1_DCNT_10	32	32
A006 364Ch	DMA1	Destination skip register 10	DMAC1_DSKP_10	32	32
A006 3660h	DMA1	Source continuous register 11	DMAC1_SCNT_11	32	32
A006 3664h	DMA1	Source skip register 11	DMAC1_SSKP_11	32	32
A006 3668h	DMA1	Destination continuous register 11	DMAC1_DCNT_11	32	32
A006 366Ch	DMA1	Destination skip register 11	DMAC1_DSKP_11	32	32
A006 3680h	DMA1	Source continuous register 12	DMAC1_SCNT_12	32	32
A006 3684h	DMA1	Source skip register 12	DMAC1_SSKP_12	32	32
A006 3688h	DMA1	Destination continuous register 12	DMAC1_DCNT_12	32	32
A006 368Ch	DMA1	Destination skip register 12	DMAC1_DSKP_12	32	32
A006 36A0h	DMA1	Source continuous register 13	DMAC1_SCNT_13	32	32
A006 36A4h	DMA1	Source skip register 13	DMAC1_SSKP_13	32	32
A006 36A8h	DMA1	Destination continuous register 13	DMAC1_DCNT_13	32	32
A006 36ACh	DMA1	Destination skip register 13	DMAC1_DSKP_13	32	32
A006 36C0h	DMA1	Source continuous register 14	DMAC1_SCNT_14	32	32
A006 36C4h	DMA1	Source skip register 14	DMAC1_SSKP_14	32	32
A006 36C8h	DMA1	Destination continuous register 14	DMAC1_DCNT_14	32	32
A006 36CCh	DMA1	Destination skip register 14	DMAC1_DSKP_14	32	32
A006 36E0h	DMA1	Source continuous register 15	DMAC1_SCNT_15	32	32
A006 36E4h	DMA1	Source skip register 15	DMAC1_SSKP_15	32	32
A006 36E8h	DMA1	Destination continuous register 15	DMAC1_DCNT_15	32	32
A006 36ECh	DMA1	Destination skip register 15	DMAC1_DSKP_15	32	32
A006 3700h	DMA1	DMA control register B	DMAC1_DCTRL_B	32	32
A006 3704h	DMA1	Descriptor interval register B	DMAC1_DSCITVL_B	32	32

Table 5.1 List of I/O Registers (Address Order) (38 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 3710h	DMA1	DMA status EN register B	DMAC1_DST_EN_B	32	32
A006 3714h	DMA1	DMA status ER register B	DMAC1_DST_ER_B	32	32
A006 3718h	DMA1	DMA status END register B	DMAC1_DST_END_B	32	32
A006 3720h	DMA1	DMA status SUS register B	DMAC1_DST_SUS_B	32	32
A006 5000h	SCIFA0	Serial mode register	SMR	16	16
A006 5002h	SCIFA0	Bit rate register	BRR	8	8
A006 5002h	SCIFA0	Modulation duty register	MDDR	8	8
A006 5004h	SCIFA0	Serial control register	SCR	16	16
A006 5006h	SCIFA0	Transmit FIFO data register	FTDR	8	8
A006 5008h	SCIFA0	Serial status register	FSR	16	16
A006 500Ah	SCIFA0	Receive FIFO data register	FRDR	8	8
A006 500Ch	SCIFA0	FIFO control register	FCR	16	16
A006 500Eh	SCIFA0	FIFO data count register	FDR	16	16
A006 5010h	SCIFA0	Serial port register	SPTR	16	16
A006 5012h	SCIFA0	Line status register	LSR	16	16
A006 5014h	SCIFA0	Serial extended mode register	SEMR	8	8
A006 5016h	SCIFA0	FIFO trigger control register	FTCR	16	16
A006 5400h	SCIFA1	Serial mode register	SMR	16	16
A006 5402h	SCIFA1	Bit rate register	BRR	8	8
A006 5402h	SCIFA1	Modulation duty register	MDDR	8	8
A006 5404h	SCIFA1	Serial control register	SCR	16	16
A006 5406h	SCIFA1	Transmit FIFO data register	FTDR	8	8
A006 5408h	SCIFA1	Serial status register	FSR	16	16
A006 540Ah	SCIFA1	Receive FIFO data register	FRDR	8	8
A006 540Ch	SCIFA1	FIFO control register	FCR	16	16
A006 540Eh	SCIFA1	FIFO data count register	FDR	16	16
A006 5410h	SCIFA1	Serial port register	SPTR	16	16
A006 5412h	SCIFA1	Line status register	LSR	16	16
A006 5414h	SCIFA1	Serial extended mode register	SEMR	8	8
A006 5416h	SCIFA1	FIFO trigger control register	FTCR	16	16
A006 5800h	SCIFA2	Serial mode register	SMR	16	16
A006 5802h	SCIFA2	Bit rate register	BRR	8	8
A006 5802h	SCIFA2	Modulation duty register	MDDR	8	8
A006 5804h	SCIFA2	Serial control register	SCR	16	16
A006 5806h	SCIFA2	Transmit FIFO data register	FTDR	8	8
A006 5808h	SCIFA2	Serial status register	FSR	16	16
A006 580Ah	SCIFA2	Receive FIFO data register	FRDR	8	8
A006 580Ch	SCIFA2	FIFO control register	FCR	16	16
A006 580Eh	SCIFA2	FIFO data count register	FDR	16	16
A006 5810h	SCIFA2	Serial port register	SPTR	16	16
A006 5812h	SCIFA2	Line status register	LSR	16	16
A006 5814h	SCIFA2	Serial extended mode register	SEMR	8	8
A006 5816h	SCIFA2	FIFO trigger control register	FTCR	16	16
A006 5C00h	SCIFA3	Serial mode register	SMR	16	16
A006 5C02h	SCIFA3	Bit rate register	BRR	8	8
A006 5C02h	SCIFA3	Modulation duty register	MDDR	8	8
A006 5C04h	SCIFA3	Serial control register	SCR	16	16
A006 5C06h	SCIFA3	Transmit FIFO data register	FTDR	8	8

Table 5.1 List of I/O Registers (Address Order) (39 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 5C08h	SCIFA3	Serial status register	FSR	16	16
A006 5C0Ah	SCIFA3	Receive FIFO data register	FRDR	8	8
A006 5C0Ch	SCIFA3	FIFO control register	FCR	16	16
A006 5C0Eh	SCIFA3	FIFO data count register	FDR	16	16
A006 5C10h	SCIFA3	Serial port register	SPTR	16	16
A006 5C12h	SCIFA3	Line status register	LSR	16	16
A006 5C14h	SCIFA3	Serial extended mode register	SEMR	8	8
A006 5C16h	SCIFA3	FIFO trigger control register	FTCR	16	16
A006 6000h	SCIFA4	Serial mode register	SMR	16	16
A006 6002h	SCIFA4	Bit rate register	BRR	8	8
A006 6002h	SCIFA4	Modulation duty register	MDDR	8	8
A006 6004h	SCIFA4	Serial control register	SCR	16	16
A006 6006h	SCIFA4	Transmit FIFO data register	FTDR	8	8
A006 6008h	SCIFA4	Serial status register	FSR	16	16
A006 600Ah	SCIFA4	Receive FIFO data register	FRDR	8	8
A006 600Ch	SCIFA4	FIFO control register	FCR	16	16
A006 600Eh	SCIFA4	FIFO data count register	FDR	16	16
A006 6010h	SCIFA4	Serial port register	SPTR	16	16
A006 6012h	SCIFA4	Line status register	LSR	16	16
A006 6014h	SCIFA4	Serial extended mode register	SEMR	8	8
A006 6016h	SCIFA4	FIFO trigger control register	FTCR	16	16
A006 8000h	RSPI0	RSPI control register	SPCR	8	8
A006 8001h	RSPI0	RSPI slave select polarity register	SSLP	8	8
A006 8002h	RSPI0	RSPI pin control register	SPPCR	8	8
A006 8003h	RSPI0	RSPI status register	SPSR	8	8
A006 8004h	RSPI0	RSPI data register	SPDR	32	16, 32
A006 8008h	RSPI0	RSPI sequence control register	SPSCR	8	8
A006 8009h	RSPI0	RSPI sequence status register	SPSSR	8	8
A006 800Ah	RSPI0	RSPI bit rate register	SPBR	8	8
A006 800Bh	RSPI0	RSPI data control register	SPDCR	8	8
A006 800Ch	RSPI0	RSPI clock delay register	SPCKD	8	8
A006 800Dh	RSPI0	RSPI slave select negation delay register	SSLND	8	8
A006 800Eh	RSPI0	RSPI next-access delay register	SPND	8	8
A006 800Fh	RSPI0	RSPI control register 2	SPCR2	8	8
A006 8010h	RSPI0	RSPI command register 0	SPCMD0	16	16
A006 8012h	RSPI0	RSPI command register 1	SPCMD1	16	16
A006 8014h	RSPI0	RSPI command register 2	SPCMD2	16	16
A006 8016h	RSPI0	RSPI command register 3	SPCMD3	16	16
A006 8018h	RSPI0	RSPI command register 4	SPCMD4	16	16
A006 801Ah	RSPI0	RSPI command register 5	SPCMD5	16	16
A006 801Ch	RSPI0	RSPI command register 6	SPCMD6	16	16
A006 801Eh	RSPI0	RSPI command register 7	SPCMD7	16	16
A006 8400h	RSPI1	RSPI control register	SPCR	8	8
A006 8401h	RSPI1	RSPI slave select polarity register	SSLP	8	8
A006 8402h	RSPI1	RSPI pin control register	SPPCR	8	8
A006 8403h	RSPI1	RSPI status register	SPSR	8	8
A006 8404h	RSPI1	RSPI data register	SPDR	32	16, 32
A006 8408h	RSPI1	RSPI sequence control register	SPSCR	8	8

Table 5.1 List of I/O Registers (Address Order) (40 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A006 8409h	RSPI1	RSPI sequence status register	SPSSR	8	8
A006 840Ah	RSPI1	RSPI bit rate register	SPBR	8	8
A006 840Bh	RSPI1	RSPI data control register	SPDCR	8	8
A006 840Ch	RSPI1	RSPI clock delay register	SPCKD	8	8
A006 840Dh	RSPI1	RSPI slave select negation delay register	SSLND	8	8
A006 840Eh	RSPI1	RSPI next-access delay register	SPND	8	8
A006 840Fh	RSPI1	RSPI control register 2	SPCR2	8	8
A006 8410h	RSPI1	RSPI command register 0	SPCMD0	16	16
A006 8412h	RSPI1	RSPI command register 1	SPCMD1	16	16
A006 8414h	RSPI1	RSPI command register 2	SPCMD2	16	16
A006 8416h	RSPI1	RSPI command register 3	SPCMD3	16	16
A006 8418h	RSPI1	RSPI command register 4	SPCMD4	16	16
A006 841Ah	RSPI1	RSPI command register 5	SPCMD5	16	16
A006 841Ch	RSPI1	RSPI command register 6	SPCMD6	16	16
A006 841Eh	RSPI1	RSPI command register 7	SPCMD7	16	16
A007 8010h	RSCAN	Channel 1 configuration register	RSCAN0C1CFG	32	8, 16, 32
A007 8014h	RSCAN	Channel 1 control register	RSCAN0C1CTR	32	8, 16, 32
A007 8018h	RSCAN	Channel 1 status register	RSCAN0C1STS	32	8, 16, 32
A007 801Ch	RSCAN	Channel 1 error flag register	RSCAN0C1ERFL	32	8, 16, 32
A007 8084h	RSCAN	Global configuration register	RSCAN0GCFG	32	8, 16, 32
A007 8088h	RSCAN	Global control register	RSCAN0GCTR	32	8, 16, 32
A007 808Ch	RSCAN	Global status register	RSCAN0GSTS	32	8, 16, 32
A007 8090h	RSCAN	Global error flag register	RSCAN0GERFL	32	8, 16, 32
A007 8094h	RSCAN	Global time stamp counter register	RSCAN0GTSC	32	16, 32
A007 8098h	RSCAN	Receive rule entry control register	RSCAN0GAFLECTR	32	8, 16, 32
A007 809Ch	RSCAN	Receive rule configuration register 0	RSCAN0GAFLCFG0	32	8, 16, 32
A007 80A4h	RSCAN	Receive buffer number register	RSCAN0RMNB	32	8, 16, 32
A007 80A8h	RSCAN	Receive buffer new data register 0	RSCAN0RMND0	32	8, 16, 32
A007 80B8h	RSCAN	Receive FIFO buffer configuration/control register 0	RSCAN0RFCC0	32	8, 16, 32
A007 80BCh	RSCAN	Receive FIFO buffer configuration/control register 1	RSCAN0RFCC1	32	8, 16, 32
A007 80C0h	RSCAN	Receive FIFO buffer configuration/control register 2	RSCAN0RFCC2	32	8, 16, 32
A007 80C4h	RSCAN	Receive FIFO buffer configuration/control register 3	RSCAN0RFCC3	32	8, 16, 32
A007 80C8h	RSCAN	Receive FIFO buffer configuration/control register 4	RSCAN0RFCC4	32	8, 16, 32
A007 80CCh	RSCAN	Receive FIFO buffer configuration/control register 5	RSCAN0RFCC5	32	8, 16, 32
A007 80D0h	RSCAN	Receive FIFO buffer configuration/control register 6	RSCAN0RFCC6	32	8, 16, 32
A007 80D4h	RSCAN	Receive FIFO buffer configuration/control register 7	RSCAN0RFCC7	32	8, 16, 32
A007 80D8h	RSCAN	Receive FIFO buffer status register 0	RSCAN0RFSTS0	32	8, 16, 32
A007 80DCh	RSCAN	Receive FIFO buffer status register 1	RSCAN0RFSTS1	32	8, 16, 32
A007 80E0h	RSCAN	Receive FIFO buffer status register 2	RSCAN0RFSTS2	32	8, 16, 32
A007 80E4h	RSCAN	Receive FIFO buffer status register 3	RSCAN0RFSTS3	32	8, 16, 32
A007 80E8h	RSCAN	Receive FIFO buffer status register 4	RSCAN0RFSTS4	32	8, 16, 32
A007 80ECh	RSCAN	Receive FIFO buffer status register 5	RSCAN0RFSTS5	32	8, 16, 32
A007 80F0h	RSCAN	Receive FIFO buffer status register 6	RSCAN0RFSTS6	32	8, 16, 32
A007 80F4h	RSCAN	Receive FIFO buffer status register 7	RSCAN0RFSTS7	32	8, 16, 32
A007 80F8h	RSCAN	Receive FIFO buffer pointer control register 0	RSCAN0RFPCTR0	32	8, 16, 32
A007 80FCh	RSCAN	Receive FIFO buffer pointer control register 1	RSCAN0RFPCTR1	32	8, 16, 32
A007 8100h	RSCAN	Receive FIFO buffer pointer control register 2	RSCAN0RFPCTR2	32	8, 16, 32
A007 8104h	RSCAN	Receive FIFO buffer pointer control register 3	RSCAN0RFPCTR3	32	8, 16, 32

Table 5.1 List of I/O Registers (Address Order) (41 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 8108h	RSCAN	Receive FIFO buffer pointer control register 4	RSCAN0RFPCTR4	32	8, 16, 32
A007 810Ch	RSCAN	Receive FIFO buffer pointer control register 5	RSCAN0RFPCTR5	32	8, 16, 32
A007 8110h	RSCAN	Receive FIFO buffer pointer control register 6	RSCAN0RFPCTR6	32	8, 16, 32
A007 8114h	RSCAN	Receive FIFO buffer pointer control register 7	RSCAN0RFPCTR7	32	8, 16, 32
A007 8124h	RSCAN	Transmit/receive FIFO buffer configuration/control register 3	RSCAN0CFCC3	32	8, 16, 32
A007 8128h	RSCAN	Transmit/receive FIFO buffer configuration/control register 4	RSCAN0CFCC4	32	8, 16, 32
A007 812Ch	RSCAN	Transmit/receive FIFO buffer configuration/control register 5	RSCAN0CFCC5	32	8, 16, 32
A007 8184h	RSCAN	Transmit/receive FIFO buffer status register 3	RSCAN0CFSTS3	32	8, 16, 32
A007 8188h	RSCAN	Transmit/receive FIFO buffer status register 4	RSCAN0CFSTS4	32	8, 16, 32
A007 818Ch	RSCAN	Transmit/receive FIFO buffer status register 5	RSCAN0CFSTS5	32	8, 16, 32
A007 81E4h	RSCAN	Transmit/receive FIFO buffer pointer control register 3	RSCAN0CFPCTR3	32	8, 16, 32
A007 81E8h	RSCAN	Transmit/receive FIFO buffer pointer control register 4	RSCAN0CFPCTR4	32	8, 16, 32
A007 81ECh	RSCAN	Transmit/receive FIFO buffer pointer control register 5	RSCAN0CFPCTR5	32	8, 16, 32
A007 8238h	RSCAN	FIFO empty status register	RSCAN0FESTS	32	8, 16, 32
A007 823Ch	RSCAN	FIFO full status register	RSCAN0FFSTS	32	8, 16, 32
A007 8240h	RSCAN	FIFO message lost status register	RSCAN0FMSTS	32	8, 16, 32
A007 8244h	RSCAN	Receive FIFO buffer interrupt flag status register	RSCAN0RFISTS	32	8, 16, 32
A007 8248h	RSCAN	Transmit/Receive FIFO Buffer Receive Interrupt Flag Status Register	RSCAN0CFRISTS	32	8, 16, 32
A007 824Ch	RSCAN	Transmit/Receive FIFO Buffer Transmit Interrupt Flag Status Register	RSCAN0CFTISTS	32	8, 16, 32
A007 8260h	RSCAN	Transmit buffer control register 16	RSCAN0TMC16	8	8
A007 8261h	RSCAN	Transmit buffer control register 17	RSCAN0TMC17	8	8
A007 8262h	RSCAN	Transmit buffer control register 18	RSCAN0TMC18	8	8
A007 8263h	RSCAN	Transmit buffer control register 19	RSCAN0TMC19	8	8
A007 8264h	RSCAN	Transmit buffer control register 20	RSCAN0TMC20	8	8
A007 8265h	RSCAN	Transmit buffer control register 21	RSCAN0TMC21	8	8
A007 8266h	RSCAN	Transmit buffer control register 22	RSCAN0TMC22	8	8
A007 8267h	RSCAN	Transmit buffer control register 23	RSCAN0TMC23	8	8
A007 8268h	RSCAN	Transmit buffer control register 24	RSCAN0TMC24	8	8
A007 8269h	RSCAN	Transmit buffer control register 25	RSCAN0TMC25	8	8
A007 826Ah	RSCAN	Transmit buffer control register 26	RSCAN0TMC26	8	8
A007 826Bh	RSCAN	Transmit buffer control register 27	RSCAN0TMC27	8	8
A007 826Ch	RSCAN	Transmit buffer control register 28	RSCAN0TMC28	8	8
A007 826Dh	RSCAN	Transmit buffer control register 29	RSCAN0TMC29	8	8
A007 826Eh	RSCAN	Transmit buffer control register 30	RSCAN0TMC30	8	8
A007 826Fh	RSCAN	Transmit buffer control register 31	RSCAN0TMC31	8	8
A007 82E0h	RSCAN	Transmit buffer status register 16	RSCAN0TMSTS16	8	8
A007 82E1h	RSCAN	Transmit buffer status register 17	RSCAN0TMSTS17	8	8
A007 82E2h	RSCAN	Transmit buffer status register 18	RSCAN0TMSTS18	8	8
A007 82E3h	RSCAN	Transmit buffer status register 19	RSCAN0TMSTS19	8	8
A007 82E4h	RSCAN	Transmit buffer status register 20	RSCAN0TMSTS20	8	8
A007 82E5h	RSCAN	Transmit buffer status register 21	RSCAN0TMSTS21	8	8
A007 82E6h	RSCAN	Transmit buffer status register 22	RSCAN0TMSTS22	8	8
A007 82E7h	RSCAN	Transmit buffer status register 23	RSCAN0TMSTS23	8	8
A007 82E8h	RSCAN	Transmit buffer status register 24	RSCAN0TMSTS24	8	8
A007 82E9h	RSCAN	Transmit buffer status register 25	RSCAN0TMSTS25	8	8
A007 82EAh	RSCAN	Transmit buffer status register 26	RSCAN0TMSTS26	8	8
A007 82EBh	RSCAN	Transmit buffer status register 27	RSCAN0TMSTS27	8	8
A007 82ECh	RSCAN	Transmit buffer status register 28	RSCAN0TMSTS28	8	8

Table 5.1 List of I/O Registers (Address Order) (42 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 82EDh	RSCAN	Transmit buffer status register 29	RSCAN0TMSTS29	8	8
A007 82EEh	RSCAN	Transmit buffer status register 30	RSCAN0TMSTS30	8	8
A007 82EFh	RSCAN	Transmit buffer status register 31	RSCAN0TMSTS31	8	8
A007 8350h	RSCAN	Transmit buffer transmit request status register 0	RSCAN0TMTRSTS0	32	8, 16, 32
A007 8360h	RSCAN	Transmit buffer transmit abort request status register 0	RSCAN0TMTARSTS0	32	8, 16, 32
A007 8370h	RSCAN	Transmit buffer transmit complete status register 0	RSCAN0TMTCSTS0	32	8, 16, 32
A007 8380h	RSCAN	Transmit buffer transmit abort status register 0	RSCAN0TMTASTS0	32	8, 16, 32
A007 8390h	RSCAN	Transmit buffer interrupt enable configuration register 0	RSCAN0TMIEC0	32	8, 16, 32
A007 83A4h	RSCAN	Transmit queue configuration/control register 1	RSCAN0TXQCC1	32	8, 16, 32
A007 83C4h	RSCAN	Transmit queue status register 1	RSCAN0TXQSTS1	32	8, 16, 32
A007 83E4h	RSCAN	Transmit queue pointer control register 1	RSCAN0TXQPCTR1	32	8, 16, 32
A007 8404h	RSCAN	Transmission history configuration/control register 1	RSCAN0THLCC1	32	8, 16, 32
A007 8424h	RSCAN	Transmission history status register 1	RSCAN0THLSTS1	32	8, 16, 32
A007 8444h	RSCAN	Transmission history pointer control register 1	RSCAN0THLPCTR1	32	8, 16, 32
A007 8460h	RSCAN	Global TX interrupt status register 0	RSCAN0GTINTSTS0	32	8, 16, 32
A007 8468h	RSCAN	Global test configuration register	RSCAN0GTSTCFG	32	8, 16, 32
A007 846Ch	RSCAN	Global test control register	RSCAN0GTSTCTR	32	8, 16, 32
A007 847Ch	RSCAN	Global lock key register	RSCAN0GLOCKK	32	16, 32
A007 8500h	RSCAN	Receive rule ID register 0	RSCAN0GAFLID0	32	8, 16, 32
A007 8504h	RSCAN	Receive rule mask register 0	RSCAN0GAFLM0	32	8, 16, 32
A007 8508h	RSCAN	Receive rule pointer 0 register 0	RSCAN0GAFLP00	32	8, 16, 32
A007 850Ch	RSCAN	Receive rule pointer 1 register 0	RSCAN0GAFLP10	32	8, 16, 32
A007 8510h	RSCAN	Receive rule ID register 1	RSCAN0GAFLID1	32	8, 16, 32
A007 8514h	RSCAN	Receive rule mask register 1	RSCAN0GAFLM1	32	8, 16, 32
A007 8518h	RSCAN	Receive rule pointer 0 register 1	RSCAN0GAFLP01	32	8, 16, 32
A007 851Ch	RSCAN	Receive rule pointer 1 register 1	RSCAN0GAFLP11	32	8, 16, 32
A007 8520h	RSCAN	Receive rule ID register 2	RSCAN0GAFLID2	32	8, 16, 32
A007 8524h	RSCAN	Receive rule mask register 2	RSCAN0GAFLM2	32	8, 16, 32
A007 8528h	RSCAN	Receive rule pointer 0 register 2	RSCAN0GAFLP02	32	8, 16, 32
A007 852Ch	RSCAN	Receive rule pointer 1 register 2	RSCAN0GAFLP12	32	8, 16, 32
A007 8530h	RSCAN	Receive rule ID register 3	RSCAN0GAFLID3	32	8, 16, 32
A007 8534h	RSCAN	Receive rule mask register 3	RSCAN0GAFLM3	32	8, 16, 32
A007 8538h	RSCAN	Receive rule pointer 0 register 3	RSCAN0GAFLP03	32	8, 16, 32
A007 853Ch	RSCAN	Receive rule pointer 1 register 3	RSCAN0GAFLP13	32	8, 16, 32
A007 8540h	RSCAN	Receive rule ID register 4	RSCAN0GAFLID4	32	8, 16, 32
A007 8544h	RSCAN	Receive rule mask register 4	RSCAN0GAFLM4	32	8, 16, 32
A007 8548h	RSCAN	Receive rule pointer 0 register 4	RSCAN0GAFLP04	32	8, 16, 32
A007 854Ch	RSCAN	Receive rule pointer 1 register 4	RSCAN0GAFLP14	32	8, 16, 32
A007 8550h	RSCAN	Receive rule ID register 5	RSCAN0GAFLID5	32	8, 16, 32
A007 8554h	RSCAN	Receive rule mask register 5	RSCAN0GAFLM5	32	8, 16, 32
A007 8558h	RSCAN	Receive rule pointer 0 register 5	RSCAN0GAFLP05	32	8, 16, 32
A007 855Ch	RSCAN	Receive rule pointer 1 register 5	RSCAN0GAFLP15	32	8, 16, 32
A007 8560h	RSCAN	Receive rule ID register 6	RSCAN0GAFLID6	32	8, 16, 32
A007 8564h	RSCAN	Receive rule mask register 6	RSCAN0GAFLM6	32	8, 16, 32
A007 8568h	RSCAN	Receive rule pointer 0 register 6	RSCAN0GAFLP06	32	8, 16, 32
A007 856Ch	RSCAN	Receive rule pointer 1 register 6	RSCAN0GAFLP16	32	8, 16, 32
A007 8570h	RSCAN	Receive rule ID register 7	RSCAN0GAFLID7	32	8, 16, 32
A007 8574h	RSCAN	Receive rule mask register 7	RSCAN0GAFLM7	32	8, 16, 32

Table 5.1 List of I/O Registers (Address Order) (43 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 8578h	RSCAN	Receive rule pointer 0 register 7	RSCAN0GAFLP07	32	8, 16, 32
A007 857Ch	RSCAN	Receive rule pointer 1 register 7	RSCAN0GAFLP17	32	8, 16, 32
A007 8580h	RSCAN	Receive rule ID register 8	RSCAN0GAFLID8	32	8, 16, 32
A007 8584h	RSCAN	Receive rule mask register 8	RSCAN0GAFLM8	32	8, 16, 32
A007 8588h	RSCAN	Receive rule pointer 0 register 8	RSCAN0GAFLP08	32	8, 16, 32
A007 858Ch	RSCAN	Receive rule pointer 1 register 8	RSCAN0GAFLP18	32	8, 16, 32
A007 8590h	RSCAN	Receive rule ID register 9	RSCAN0GAFLID9	32	8, 16, 32
A007 8594h	RSCAN	Receive rule mask register 9	RSCAN0GAFLM9	32	8, 16, 32
A007 8598h	RSCAN	Receive rule pointer 0 register 9	RSCAN0GAFLP09	32	8, 16, 32
A007 859Ch	RSCAN	Receive rule pointer 1 register 9	RSCAN0GAFLP19	32	8, 16, 32
A007 85A0h	RSCAN	Receive rule ID register 10	RSCAN0GAFLID10	32	8, 16, 32
A007 85A4h	RSCAN	Receive rule mask register 10	RSCAN0GAFLM10	32	8, 16, 32
A007 85A8h	RSCAN	Receive rule pointer 0 register 10	RSCAN0GAFLP010	32	8, 16, 32
A007 85ACh	RSCAN	Receive rule pointer 1 register 10	RSCAN0GAFLP110	32	8, 16, 32
A007 85B0h	RSCAN	Receive rule ID register 11	RSCAN0GAFLID11	32	8, 16, 32
A007 85B4h	RSCAN	Receive rule mask register 11	RSCAN0GAFLM11	32	8, 16, 32
A007 85B8h	RSCAN	Receive rule pointer 0 register 11	RSCAN0GAFLP011	32	8, 16, 32
A007 85BCh	RSCAN	Receive rule pointer 1 register 11	RSCAN0GAFLP111	32	8, 16, 32
A007 85C0h	RSCAN	Receive rule ID register 12	RSCAN0GAFLID12	32	8, 16, 32
A007 85C4h	RSCAN	Receive rule mask register 12	RSCAN0GAFLM12	32	8, 16, 32
A007 85C8h	RSCAN	Receive rule pointer 0 register 12	RSCAN0GAFLP012	32	8, 16, 32
A007 85CCh	RSCAN	Receive rule pointer 1 register 12	RSCAN0GAFLP112	32	8, 16, 32
A007 85D0h	RSCAN	Receive rule ID register 13	RSCAN0GAFLID13	32	8, 16, 32
A007 85D4h	RSCAN	Receive rule mask register 13	RSCAN0GAFLM13	32	8, 16, 32
A007 85D8h	RSCAN	Receive rule pointer 0 register 13	RSCAN0GAFLP013	32	8, 16, 32
A007 85DCh	RSCAN	Receive rule pointer 1 register 13	RSCAN0GAFLP113	32	8, 16, 32
A007 85E0h	RSCAN	Receive rule ID register 14	RSCAN0GAFLID14	32	8, 16, 32
A007 85E4h	RSCAN	Receive rule mask register 14	RSCAN0GAFLM14	32	8, 16, 32
A007 85E8h	RSCAN	Receive rule pointer 0 register 14	RSCAN0GAFLP014	32	8, 16, 32
A007 85ECh	RSCAN	Receive rule pointer 1 register 14	RSCAN0GAFLP114	32	8, 16, 32
A007 85F0h	RSCAN	Receive rule ID register 15	RSCAN0GAFLID15	32	8, 16, 32
A007 85F4h	RSCAN	Receive rule mask register 15	RSCAN0GAFLM15	32	8, 16, 32
A007 85F8h	RSCAN	Receive rule pointer 0 register 15	RSCAN0GAFLP015	32	8, 16, 32
A007 85FCh	RSCAN	Receive rule pointer 1 register 15	RSCAN0GAFLP115	32	8, 16, 32
A007 8700h	RSCAN	Receive buffer ID register 16	RSCAN0RMID16	32	8, 16, 32
A007 8704h	RSCAN	Receive buffer pointer register 16	RSCAN0RMPTR16	32	8, 16, 32
A007 8708h	RSCAN	Receive buffer data field 0 register 16	RSCAN0RMDF016	32	8, 16, 32
A007 870Ch	RSCAN	Receive buffer data field 1 register 16	RSCAN0RMDF116	32	8, 16, 32
A007 8710h	RSCAN	Receive buffer ID register 17	RSCAN0RMID17	32	8, 16, 32
A007 8714h	RSCAN	Receive buffer pointer register 17	RSCAN0RMPTR17	32	8, 16, 32
A007 8718h	RSCAN	Receive buffer data field 0 register 17	RSCAN0RMDF017	32	8, 16, 32
A007 871Ch	RSCAN	Receive buffer data field 1 register 17	RSCAN0RMDF117	32	8, 16, 32
A007 8720h	RSCAN	Receive buffer ID register 18	RSCAN0RMID18	32	8, 16, 32
A007 8724h	RSCAN	Receive buffer pointer register 18	RSCAN0RMPTR18	32	8, 16, 32
A007 8728h	RSCAN	Receive buffer data field 0 register 18	RSCAN0RMDF018	32	8, 16, 32
A007 872Ch	RSCAN	Receive buffer data field 1 register 18	RSCAN0RMDF118	32	8, 16, 32
A007 8730h	RSCAN	Receive buffer ID register 19	RSCAN0RMID19	32	8, 16, 32
A007 8734h	RSCAN	Receive buffer pointer register 19	RSCAN0RMPTR19	32	8, 16, 32

Table 5.1 List of I/O Registers (Address Order) (44 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 8738h	RSCAN	Receive buffer data field 0 register 19	RSCAN0RMDF019	32	8, 16, 32
A007 873Ch	RSCAN	Receive buffer data field 1 register 19	RSCAN0RMDF119	32	8, 16, 32
A007 8740h	RSCAN	Receive buffer ID register 20	RSCAN0RMID20	32	8, 16, 32
A007 8744h	RSCAN	Receive buffer pointer register 20	RSCAN0RMPTR20	32	8, 16, 32
A007 8748h	RSCAN	Receive buffer data field 0 register 20	RSCAN0RMDF020	32	8, 16, 32
A007 874Ch	RSCAN	Receive buffer data field 1 register 20	RSCAN0RMDF120	32	8, 16, 32
A007 8750h	RSCAN	Receive buffer ID register 21	RSCAN0RMID21	32	8, 16, 32
A007 8754h	RSCAN	Receive buffer pointer register 21	RSCAN0RMPTR21	32	8, 16, 32
A007 8758h	RSCAN	Receive buffer data field 0 register 21	RSCAN0RMDF021	32	8, 16, 32
A007 875Ch	RSCAN	Receive buffer data field 1 register 21	RSCAN0RMDF121	32	8, 16, 32
A007 8760h	RSCAN	Receive buffer ID register 22	RSCAN0RMID22	32	8, 16, 32
A007 8764h	RSCAN	Receive buffer pointer register 22	RSCAN0RMPTR22	32	8, 16, 32
A007 8768h	RSCAN	Receive buffer data field 0 register 22	RSCAN0RMDF022	32	8, 16, 32
A007 876Ch	RSCAN	Receive buffer data field 1 register 22	RSCAN0RMDF122	32	8, 16, 32
A007 8770h	RSCAN	Receive buffer ID register 23	RSCAN0RMID23	32	8, 16, 32
A007 8774h	RSCAN	Receive buffer pointer register 23	RSCAN0RMPTR23	32	8, 16, 32
A007 8778h	RSCAN	Receive buffer data field 0 register 23	RSCAN0RMDF023	32	8, 16, 32
A007 877Ch	RSCAN	Receive buffer data field 1 register 23	RSCAN0RMDF123	32	8, 16, 32
A007 8780h	RSCAN	Receive buffer ID register 24	RSCAN0RMID24	32	8, 16, 32
A007 8784h	RSCAN	Receive buffer pointer register 24	RSCAN0RMPTR24	32	8, 16, 32
A007 8788h	RSCAN	Receive buffer data field 0 register 24	RSCAN0RMDF024	32	8, 16, 32
A007 878Ch	RSCAN	Receive buffer data field 1 register 24	RSCAN0RMDF124	32	8, 16, 32
A007 8790h	RSCAN	Receive buffer ID register 25	RSCAN0RMID25	32	8, 16, 32
A007 8794h	RSCAN	Receive buffer pointer register 25	RSCAN0RMPTR25	32	8, 16, 32
A007 8798h	RSCAN	Receive buffer data field 0 register 25	RSCAN0RMDF025	32	8, 16, 32
A007 879Ch	RSCAN	Receive buffer data field 1 register 25	RSCAN0RMDF125	32	8, 16, 32
A007 87A0h	RSCAN	Receive buffer ID register 26	RSCAN0RMID26	32	8, 16, 32
A007 87A4h	RSCAN	Receive buffer pointer register 26	RSCAN0RMPTR26	32	8, 16, 32
A007 87A8h	RSCAN	Receive buffer data field 0 register 26	RSCAN0RMDF026	32	8, 16, 32
A007 87ACh	RSCAN	Receive buffer data field 1 register 26	RSCAN0RMDF126	32	8, 16, 32
A007 87B0h	RSCAN	Receive buffer ID register 27	RSCAN0RMID27	32	8, 16, 32
A007 87B4h	RSCAN	Receive buffer pointer register 27	RSCAN0RMPTR27	32	8, 16, 32
A007 87B8h	RSCAN	Receive buffer data field 0 register 27	RSCAN0RMDF027	32	8, 16, 32
A007 87BCh	RSCAN	Receive buffer data field 1 register 27	RSCAN0RMDF127	32	8, 16, 32
A007 87C0h	RSCAN	Receive buffer ID register 28	RSCAN0RMID28	32	8, 16, 32
A007 87C4h	RSCAN	Receive buffer pointer register 28	RSCAN0RMPTR28	32	8, 16, 32
A007 87C8h	RSCAN	Receive buffer data field 0 register 28	RSCAN0RMDF028	32	8, 16, 32
A007 87CCh	RSCAN	Receive buffer data field 1 register 28	RSCAN0RMDF128	32	8, 16, 32
A007 87D0h	RSCAN	Receive buffer ID register 29	RSCAN0RMID29	32	8, 16, 32
A007 87D4h	RSCAN	Receive buffer pointer register 29	RSCAN0RMPTR29	32	8, 16, 32
A007 87D8h	RSCAN	Receive buffer data field 0 register 29	RSCAN0RMDF029	32	8, 16, 32
A007 87DCh	RSCAN	Receive buffer data field 1 register 29	RSCAN0RMDF129	32	8, 16, 32
A007 87E0h	RSCAN	Receive buffer ID register 30	RSCAN0RMID30	32	8, 16, 32
A007 87E4h	RSCAN	Receive buffer pointer register 30	RSCAN0RMPTR30	32	8, 16, 32
A007 87E8h	RSCAN	Receive buffer data field 0 register 30	RSCAN0RMDF030	32	8, 16, 32
A007 87ECh	RSCAN	Receive buffer data field 1 register 30	RSCAN0RMDF130	32	8, 16, 32
A007 87F0h	RSCAN	Receive buffer ID register 31	RSCAN0RMID31	32	8, 16, 32
A007 87F4h	RSCAN	Receive buffer pointer register 31	RSCAN0RMPTR31	32	8, 16, 32

Table 5.1 List of I/O Registers (Address Order) (45 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 87F8h	RSCAN	Receive buffer data field 0 register 31	RSCAN0RMDF031	32	8, 16, 32
A007 87FCh	RSCAN	Receive buffer data field 1 register 31	RSCAN0RMDF131	32	8, 16, 32
A007 8E00h	RSCAN	Receive FIFO buffer access ID register 0	RSCAN0RFID0	32	8, 16, 32
A007 8E04h	RSCAN	Receive FIFO buffer access pointer register 0	RSCAN0RFPTR0	32	8, 16, 32
A007 8E08h	RSCAN	Receive FIFO buffer access data field 0 register 0	RSCAN0RFDF00	32	8, 16, 32
A007 8E0Ch	RSCAN	Receive FIFO buffer access data field 1 register 0	RSCAN0RFDF10	32	8, 16, 32
A007 8E10h	RSCAN	Receive FIFO buffer access ID register 1	RSCAN0RFID1	32	8, 16, 32
A007 8E14h	RSCAN	Receive FIFO buffer access pointer register 1	RSCAN0RFPTR1	32	8, 16, 32
A007 8E18h	RSCAN	Receive FIFO buffer access data field 0 register 1	RSCAN0RFDF01	32	8, 16, 32
A007 8E1Ch	RSCAN	Receive FIFO buffer access data field 1 register 1	RSCAN0RFDF11	32	8, 16, 32
A007 8E20h	RSCAN	Receive FIFO buffer access ID register 2	RSCAN0RFID2	32	8, 16, 32
A007 8E24h	RSCAN	Receive FIFO buffer access pointer register 2	RSCAN0RFPTR2	32	8, 16, 32
A007 8E28h	RSCAN	Receive FIFO buffer access data field 0 register 2	RSCAN0RFDF02	32	8, 16, 32
A007 8E2Ch	RSCAN	Receive FIFO buffer access data field 1 register 2	RSCAN0RFDF12	32	8, 16, 32
A007 8E30h	RSCAN	Receive FIFO buffer access ID register 3	RSCAN0RFID3	32	8, 16, 32
A007 8E34h	RSCAN	Receive FIFO buffer access pointer register 3	RSCAN0RFPTR3	32	8, 16, 32
A007 8E38h	RSCAN	Receive FIFO buffer access data field 0 register 3	RSCAN0RFDF03	32	8, 16, 32
A007 8E3Ch	RSCAN	Receive FIFO buffer access data field 1 register 3	RSCAN0RFDF13	32	8, 16, 32
A007 8E40h	RSCAN	Receive FIFO buffer access ID register 4	RSCAN0RFID4	32	8, 16, 32
A007 8E44h	RSCAN	Receive FIFO buffer access pointer register 4	RSCAN0RFPTR4	32	8, 16, 32
A007 8E48h	RSCAN	Receive FIFO buffer access data field 0 register 4	RSCAN0RFDF04	32	8, 16, 32
A007 8E4Ch	RSCAN	Receive FIFO buffer access data field 1 register 4	RSCAN0RFDF14	32	8, 16, 32
A007 8E50h	RSCAN	Receive FIFO buffer access ID register 5	RSCAN0RFID5	32	8, 16, 32
A007 8E54h	RSCAN	Receive FIFO buffer access pointer register 5	RSCAN0RFPTR5	32	8, 16, 32
A007 8E58h	RSCAN	Receive FIFO buffer access data field 0 register 5	RSCAN0RFDF05	32	8, 16, 32
A007 8E5Ch	RSCAN	Receive FIFO buffer access data field 1 register 5	RSCAN0RFDF15	32	8, 16, 32
A007 8E60h	RSCAN	Receive FIFO buffer access ID register 6	RSCAN0RFID6	32	8, 16, 32
A007 8E64h	RSCAN	Receive FIFO buffer access pointer register 6	RSCAN0RFPTR6	32	8, 16, 32
A007 8E68h	RSCAN	Receive FIFO buffer access data field 0 register 6	RSCAN0RFDF06	32	8, 16, 32
A007 8E6Ch	RSCAN	Receive FIFO buffer access data field 1 register 6	RSCAN0RFDF16	32	8, 16, 32
A007 8E70h	RSCAN	Receive FIFO buffer access ID register 7	RSCAN0RFID7	32	8, 16, 32
A007 8E74h	RSCAN	Receive FIFO buffer access pointer register 7	RSCAN0RFPTR7	32	8, 16, 32
A007 8E78h	RSCAN	Receive FIFO buffer access data field 0 register 7	RSCAN0RFDF07	32	8, 16, 32
A007 8E7Ch	RSCAN	Receive FIFO buffer access data field 1 register 7	RSCAN0RFDF17	32	8, 16, 32
A007 8EB0h	RSCAN	Transmit/receive FIFO buffer access ID register 3	RSCAN0CFID3	32	8, 16, 32
A007 8EB4h	RSCAN	Transmit/receive FIFO buffer access pointer register 3	RSCAN0CFPTR3	32	8, 16, 32
A007 8EB8h	RSCAN	Transmit/receive FIFO buffer access data field 0 register 3	RSCAN0CFDF03	32	8, 16, 32
A007 8EBCh	RSCAN	Transmit/receive FIFO buffer access data field 1 register 3	RSCAN0CFDF13	32	8, 16, 32
A007 8EC0h	RSCAN	Transmit/receive FIFO buffer access ID register 4	RSCAN0CFID4	32	8, 16, 32
A007 8EC4h	RSCAN	Transmit/receive FIFO buffer access pointer register 4	RSCAN0CFPTR4	32	8, 16, 32
A007 8EC8h	RSCAN0	Transmit/receive FIFO buffer access data field 0 register 4	RSCAN0CFDF04	32	8, 16, 32
A007 8ECCh	RSCAN	Transmit/receive FIFO buffer access data field 1 register 4	RSCAN0CFDF14	32	8, 16, 32
A007 8ED0h	RSCAN	Transmit/receive FIFO buffer access ID register 5	RSCAN0CFID5	32	8, 16, 32
A007 8ED4h	RSCAN	Transmit/receive FIFO buffer access pointer register 5	RSCAN0CFPTR5	32	8, 16, 32
A007 8ED8h	RSCAN	Transmit/receive FIFO buffer access data field 0 register 5	RSCAN0CFDF05	32	8, 16, 32
A007 8EDCh	RSCAN	Transmit/receive FIFO buffer access data field 1 register 5	RSCAN0CFDF15	32	8, 16, 32
A007 9100h	RSCAN	Transmit buffer ID register 16	RSCAN0TMID16	32	8, 16, 32
A007 9104h	RSCAN	Transmit buffer pointer register 16	RSCAN0TMPTR16	32	8, 16, 32

Table 5.1 List of I/O Registers (Address Order) (46 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 9108h	RSCAN	Transmit buffer data field 0 register 16	RSCAN0TMDF016	32	8, 16, 32
A007 910Ch	RSCAN	Transmit buffer data field 1 register 16	RSCAN0TMDF116	32	8, 16, 32
A007 9110h	RSCAN	Transmit buffer ID register 17	RSCAN0TMID17	32	8, 16, 32
A007 9114h	RSCAN	Transmit buffer pointer register 17	RSCAN0TMPTR17	32	8, 16, 32
A007 9118h	RSCAN	Transmit buffer data field 0 register 17	RSCAN0TMDF017	32	8, 16, 32
A007 911Ch	RSCAN	Transmit buffer data field 1 register 17	RSCAN0TMDF117	32	8, 16, 32
A007 9120h	RSCAN	Transmit buffer ID register 18	RSCAN0TMID18	32	8, 16, 32
A007 9124h	RSCAN	Transmit buffer pointer register 18	RSCAN0TMPTR18	32	8, 16, 32
A007 9128h	RSCAN	Transmit buffer data field 0 register 18	RSCAN0TMDF018	32	8, 16, 32
A007 912Ch	RSCAN	Transmit buffer data field 1 register 18	RSCAN0TMDF118	32	8, 16, 32
A007 9130h	RSCAN	Transmit buffer ID register 19	RSCAN0TMID19	32	8, 16, 32
A007 9134h	RSCAN	Transmit buffer pointer register 19	RSCAN0TMPTR19	32	8, 16, 32
A007 9138h	RSCAN	Transmit buffer data field 0 register 19	RSCAN0TMDF019	32	8, 16, 32
A007 913Ch	RSCAN	Transmit buffer data field 1 register 19	RSCAN0TMDF119	32	8, 16, 32
A007 9140h	RSCAN	Transmit buffer ID register 20	RSCAN0TMID20	32	8, 16, 32
A007 9144h	RSCAN	Transmit buffer pointer register 20	RSCAN0TMPTR20	32	8, 16, 32
A007 9148h	RSCAN	Transmit buffer data field 0 register 20	RSCAN0TMDF020	32	8, 16, 32
A007 914Ch	RSCAN	Transmit buffer data field 1 register 20	RSCAN0TMDF120	32	8, 16, 32
A007 9150h	RSCAN	Transmit buffer ID register 21	RSCAN0TMID21	32	8, 16, 32
A007 9154h	RSCAN	Transmit buffer pointer register 21	RSCAN0TMPTR21	32	8, 16, 32
A007 9158h	RSCAN	Transmit buffer data field 0 register 21	RSCAN0TMDF021	32	8, 16, 32
A007 915Ch	RSCAN	Transmit buffer data field 1 register 21	RSCAN0TMDF121	32	8, 16, 32
A007 9160h	RSCAN	Transmit buffer ID register 22	RSCAN0TMID22	32	8, 16, 32
A007 9164h	RSCAN	Transmit buffer pointer register 22	RSCAN0TMPTR22	32	8, 16, 32
A007 9168h	RSCAN	Transmit buffer data field 0 register 22	RSCAN0TMDF022	32	8, 16, 32
A007 916Ch	RSCAN	Transmit buffer data field 1 register 22	RSCAN0TMDF122	32	8, 16, 32
A007 9170h	RSCAN	Transmit buffer ID register 23	RSCAN0TMID23	32	8, 16, 32
A007 9174h	RSCAN	Transmit buffer pointer register 23	RSCAN0TMPTR23	32	8, 16, 32
A007 9178h	RSCAN	Transmit buffer data field 0 register 23	RSCAN0TMDF023	32	8, 16, 32
A007 917Ch	RSCAN	Transmit buffer data field 1 register 23	RSCAN0TMDF123	32	8, 16, 32
A007 9180h	RSCAN	Transmit buffer ID register 24	RSCAN0TMID24	32	8, 16, 32
A007 9184h	RSCAN	Transmit buffer pointer register 24	RSCAN0TMPTR24	32	8, 16, 32
A007 9188h	RSCAN	Transmit buffer data field 0 register 24	RSCAN0TMDF024	32	8, 16, 32
A007 918Ch	RSCAN	Transmit buffer data field 1 register 24	RSCAN0TMDF124	32	8, 16, 32
A007 9190h	RSCAN	Transmit buffer ID register 25	RSCAN0TMID25	32	8, 16, 32
A007 9194h	RSCAN	Transmit buffer pointer register 25	RSCAN0TMPTR25	32	8, 16, 32
A007 9198h	RSCAN	Transmit buffer data field 0 register 25	RSCAN0TMDF025	32	8, 16, 32
A007 919Ch	RSCAN	Transmit buffer data field 1 register 25	RSCAN0TMDF125	32	8, 16, 32
A007 91A0h	RSCAN	Transmit buffer ID register 26	RSCAN0TMID26	32	8, 16, 32
A007 91A4h	RSCAN	Transmit buffer pointer register 26	RSCAN0TMPTR26	32	8, 16, 32
A007 91A8h	RSCAN	Transmit buffer data field 0 register 26	RSCAN0TMDF026	32	8, 16, 32
A007 91ACh	RSCAN	Transmit buffer data field 1 register 26	RSCAN0TMDF126	32	8, 16, 32
A007 91B0h	RSCAN	Transmit buffer ID register 27	RSCAN0TMID27	32	8, 16, 32
A007 91B4h	RSCAN	Transmit buffer pointer register 27	RSCAN0TMPTR27	32	8, 16, 32
A007 91B8h	RSCAN	Transmit buffer data field 0 register 27	RSCAN0TMDF027	32	8, 16, 32
A007 91BCh	RSCAN	Transmit buffer data field 1 register 27	RSCAN0TMDF127	32	8, 16, 32
A007 91C0h	RSCAN	Transmit buffer ID register 28	RSCAN0TMID28	32	8, 16, 32
A007 91C4h	RSCAN	Transmit buffer pointer register 28	RSCAN0TMPTR28	32	8, 16, 32

Table 5.1 List of I/O Registers (Address Order) (47 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 91C8h	RSCAN	Transmit buffer data field 0 register 28	RSCAN0TMDF028	32	8, 16, 32
A007 91CCh	RSCAN	Transmit buffer data field 1 register 28	RSCAN0TMDF128	32	8, 16, 32
A007 91D0h	RSCAN	Transmit buffer ID register 29	RSCAN0TMID29	32	8, 16, 32
A007 91D4h	RSCAN	Transmit buffer pointer register 29	RSCAN0TMPTR29	32	8, 16, 32
A007 91D8h	RSCAN	Transmit buffer data field 0 register 29	RSCAN0TMDF029	32	8, 16, 32
A007 91DCh	RSCAN	Transmit buffer data field 1 register 29	RSCAN0TMDF129	32	8, 16, 32
A007 91E0h	RSCAN	Transmit buffer ID register 30	RSCAN0TMID30	32	8, 16, 32
A007 91E4h	RSCAN	Transmit buffer pointer register 30	RSCAN0TMPTR30	32	8, 16, 32
A007 91E8h	RSCAN	Transmit buffer data field 0 register 30	RSCAN0TMDF030	32	8, 16, 32
A007 91ECh	RSCAN	Transmit buffer data field 1 register 30	RSCAN0TMDF130	32	8, 16, 32
A007 91F0h	RSCAN	Transmit buffer ID register 31	RSCAN0TMID31	32	8, 16, 32
A007 91F4h	RSCAN	Transmit buffer pointer register 31	RSCAN0TMPTR31	32	8, 16, 32
A007 91F8h	RSCAN	Transmit buffer data field 0 register 31	RSCAN0TMDF031	32	8, 16, 32
A007 91FCh	RSCAN	Transmit buffer data field 1 register 31	RSCAN0TMDF131	32	8, 16, 32
A007 9804h	RSCAN	Transmission history access register 1	RSCAN0THLACC1	32	8, 16, 32
A007 9900h	RSCAN	RAM test page access register 0	RSCAN0RPGACC0	32	8, 16, 32
A007 9904h	RSCAN	RAM test page access register 1	RSCAN0RPGACC1	32	8, 16, 32
A007 9908h	RSCAN	RAM test page access register 2	RSCAN0RPGACC2	32	8, 16, 32
A007 990Ch	RSCAN	RAM test page access register 3	RSCAN0RPGACC3	32	8, 16, 32
A007 9910h	RSCAN	RAM test page access register 4	RSCAN0RPGACC4	32	8, 16, 32
A007 9914h	RSCAN	RAM test page access register 5	RSCAN0RPGACC5	32	8, 16, 32
A007 9918h	RSCAN	RAM test page access register 6	RSCAN0RPGACC6	32	8, 16, 32
A007 991Ch	RSCAN	RAM test page access register 7	RSCAN0RPGACC7	32	8, 16, 32
A007 9920h	RSCAN	RAM test page access register 8	RSCAN0RPGACC8	32	8, 16, 32
A007 9924h	RSCAN	RAM test page access register 9	RSCAN0RPGACC9	32	8, 16, 32
A007 9928h	RSCAN	RAM test page access register 10	RSCAN0RPGACC10	32	8, 16, 32
A007 992Ch	RSCAN	RAM test page access register 11	RSCAN0RPGACC11	32	8, 16, 32
A007 9930h	RSCAN	RAM test page access register 12	RSCAN0RPGACC12	32	8, 16, 32
A007 9934h	RSCAN	RAM test page access register 13	RSCAN0RPGACC13	32	8, 16, 32
A007 9938h	RSCAN	RAM test page access register 14	RSCAN0RPGACC14	32	8, 16, 32
A007 993Ch	RSCAN	RAM test page access register 15	RSCAN0RPGACC15	32	8, 16, 32
A007 9940h	RSCAN	RAM test page access register 16	RSCAN0RPGACC16	32	8, 16, 32
A007 9944h	RSCAN	RAM test page access register 17	RSCAN0RPGACC17	32	8, 16, 32
A007 9948h	RSCAN	RAM test page access register 18	RSCAN0RPGACC18	32	8, 16, 32
A007 994Ch	RSCAN	RAM test page access register 19	RSCAN0RPGACC19	32	8, 16, 32
A007 9950h	RSCAN	RAM test page access register 20	RSCAN0RPGACC20	32	8, 16, 32
A007 9954h	RSCAN	RAM test page access register 21	RSCAN0RPGACC21	32	8, 16, 32
A007 9958h	RSCAN	RAM test page access register 22	RSCAN0RPGACC22	32	8, 16, 32
A007 995Ch	RSCAN	RAM test page access register 23	RSCAN0RPGACC23	32	8, 16, 32
A007 9960h	RSCAN	RAM test page access register 24	RSCAN0RPGACC24	32	8, 16, 32
A007 9964h	RSCAN	RAM test page access register 25	RSCAN0RPGACC25	32	8, 16, 32
A007 9968h	RSCAN	RAM test page access register 26	RSCAN0RPGACC26	32	8, 16, 32
A007 996Ch	RSCAN	RAM test page access register 27	RSCAN0RPGACC27	32	8, 16, 32
A007 9970h	RSCAN	RAM test page access register 28	RSCAN0RPGACC28	32	8, 16, 32
A007 9974h	RSCAN	RAM test page access register 29	RSCAN0RPGACC29	32	8, 16, 32
A007 9978h	RSCAN	RAM test page access register 30	RSCAN0RPGACC30	32	8, 16, 32
A007 997Ch	RSCAN	RAM test page access register 31	RSCAN0RPGACC31	32	8, 16, 32
A007 9980h	RSCAN	RAM test page access register 32	RSCAN0RPGACC32	32	8, 16, 32

Table 5.1 List of I/O Registers (Address Order) (48 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 9984h	RSCAN	RAM test page access register 33	RSCAN0RPGACC33	32	8, 16, 32
A007 9988h	RSCAN	RAM test page access register 34	RSCAN0RPGACC34	32	8, 16, 32
A007 998Ch	RSCAN	RAM test page access register 35	RSCAN0RPGACC35	32	8, 16, 32
A007 9990h	RSCAN	RAM test page access register 36	RSCAN0RPGACC36	32	8, 16, 32
A007 9994h	RSCAN	RAM test page access register 37	RSCAN0RPGACC37	32	8, 16, 32
A007 9998h	RSCAN	RAM test page access register 38	RSCAN0RPGACC38	32	8, 16, 32
A007 999Ch	RSCAN	RAM test page access register 39	RSCAN0RPGACC39	32	8, 16, 32
A007 99A0h	RSCAN	RAM test page access register 40	RSCAN0RPGACC40	32	8, 16, 32
A007 99A4h	RSCAN	RAM test page access register 41	RSCAN0RPGACC41	32	8, 16, 32
A007 99A8h	RSCAN	RAM test page access register 42	RSCAN0RPGACC42	32	8, 16, 32
A007 99ACh	RSCAN	RAM test page access register 43	RSCAN0RPGACC43	32	8, 16, 32
A007 99B0h	RSCAN	RAM test page access register 44	RSCAN0RPGACC44	32	8, 16, 32
A007 99B4h	RSCAN	RAM test page access register 45	RSCAN0RPGACC45	32	8, 16, 32
A007 99B8h	RSCAN	RAM test page access register 46	RSCAN0RPGACC46	32	8, 16, 32
A007 99BCh	RSCAN	RAM test page access register 47	RSCAN0RPGACC47	32	8, 16, 32
A007 99C0h	RSCAN	RAM test page access register 48	RSCAN0RPGACC48	32	8, 16, 32
A007 99C4h	RSCAN	RAM test page access register 49	RSCAN0RPGACC49	32	8, 16, 32
A007 99C8h	RSCAN	RAM test page access register 50	RSCAN0RPGACC50	32	8, 16, 32
A007 99CCh	RSCAN	RAM test page access register 51	RSCAN0RPGACC51	32	8, 16, 32
A007 99D0h	RSCAN	RAM test page access register 52	RSCAN0RPGACC52	32	8, 16, 32
A007 99D4h	RSCAN	RAM test page access register 53	RSCAN0RPGACC53	32	8, 16, 32
A007 99D8h	RSCAN	RAM test page access register 54	RSCAN0RPGACC54	32	8, 16, 32
A007 99DCh	RSCAN	RAM test page access register 55	RSCAN0RPGACC55	32	8, 16, 32
A007 99E0h	RSCAN	RAM test page access register 56	RSCAN0RPGACC56	32	8, 16, 32
A007 99E4h	RSCAN	RAM test page access register 57	RSCAN0RPGACC57	32	8, 16, 32
A007 99E8h	RSCAN	RAM test page access register 58	RSCAN0RPGACC58	32	8, 16, 32
A007 99ECh	RSCAN	RAM test page access register 59	RSCAN0RPGACC59	32	8, 16, 32
A007 99F0h	RSCAN	RAM test page access register 60	RSCAN0RPGACC60	32	8, 16, 32
A007 99F4h	RSCAN	RAM test page access register 61	RSCAN0RPGACC61	32	8, 16, 32
A007 99F8h	RSCAN	RAM test page access register 62	RSCAN0RPGACC62	32	8, 16, 32
A007 99FCh	RSCAN	RAM test page access register 63	RSCAN0RPGACC63	32	8, 16, 32
A007 B000h	RSCAN	RSCAN ECC control register	ECCRCANCTL	32	8, 16, 32
A007 B010h	RSCAN	RSCAN ECC error address register 0	ECCRCANEAD0	32	8, 16, 32
A007 B014h	RSCAN	RSCAN ECC error address register 1	ECCRCANEAD1	32	8, 16, 32
A007 B018h	RSCAN	RSCAN ECC error address register 2	ECCRCANEAD2	32	8, 16, 32
A007 B01Ch	RSCAN	RSCAN ECC error address register 3	ECCRCANEAD3	32	8, 16, 32
A007 B020h	RSCAN	RSCAN ECC error address register 4	ECCRCANEAD4	32	8, 16, 32
A007 B024h	RSCAN	RSCAN ECC error address register 5	ECCRCANEAD5	32	8, 16, 32
A007 B028h	RSCAN	RSCAN ECC error address register 6	ECCRCANEAD6	32	8, 16, 32
A007 B02Ch	RSCAN	RSCAN ECC error address register 7	ECCRCANEAD7	32	8, 16, 32
A007 C000h	CRC	CRC data input register	CRCDIR	32	32
A007 C004h	CRC	CRC data output register	CRCDOR	32	32
A007 C020h	CRC	CRC control register	CRCCR	8	8
A007 D000h	ECMM	ECM master error set trigger register	ECMMESET	8	8
A007 D004h	ECMM	ECM master error clear trigger register	ECMMECLR	8	8
A007 D008h	ECMM	ECM master error source status register 0	ECMMESSTR0	32	32
A007 D00Ch	ECMM	ECM master error source status register 1	ECMMESSTR1	32	32
A007 D010h	ECMM	ECM master error source status register 2	ECMMESSTR2	32	32

Table 5.1 List of I/O Registers (Address Order) (49 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A007 D014h	ECMM	ECM master protection command register	ECMMPCMD0	32	32
A007 D040h	ECMC	ECM checker error set trigger register	ECMCESET	8	8
A007 D044h	ECMC	ECM checker error clear trigger register	ECMCECLR	8	8
A007 D048h	ECMC	ECM checker error source status register 0	ECMCESSTR0	32	32
A007 D04Ch	ECMC	ECM checker error source status register 1	ECMCESSTR1	32	32
A007 D050h	ECMC	ECM checker error source status register 2	ECMCESSTR2	32	32
A007 D054h	ECMC	ECM checker protection command register	ECMCPCMD0	32	32
A007 D080h	ECM	ECM error pulse configuration register	ECMEPCFG	8	8
A007 D084h	ECM	ECM maskable interrupt configuration register 0	ECMMICFG0	32	32
A007 D088h	ECM	ECM maskable interrupt configuration register 1	ECMMICFG1	32	32
A007 D08Ch	ECM	ECM maskable interrupt configuration register 2	ECMMICFG2	32	32
A007 D090h	ECM	ECM non-maskable interrupt configuration register 0	ECMNMICFG0	32	32
A007 D094h	ECM	ECM non-maskable interrupt configuration register 1	ECMNMICFG1	32	32
A007 D098h	ECM	ECM non-maskable interrupt configuration register 2	ECMNMICFG2	32	32
A007 D09Ch	ECM	ECM internal reset configuration register 0	ECMIRCFG0	32	32
A007 D0A0h	ECM	ECM internal reset configuration register 1	ECMIRCFG1	32	32
A007 D0A4h	ECM	ECM internal reset configuration register 2	ECMIRCFG2	32	32
A007 D0A8h	ECM	ECM error mask register 0	ECMEMK0	32	32
A007 D0ACh	ECM	ECM error mask register 1	ECMEMK1	32	32
A007 D0B0h	ECM	ECM error mask register 2	ECMEMK2	32	32
A007 D0B4h	ECM	ECM Error Source Status Clear Trigger Register 0	ECMESSTC0	32	32
A007 D0B8h	ECM	ECM Error Source Status Clear Trigger Register 1	ECMESSTC1	32	32
A007 D0BCh	ECM	ECM Error Source Status Clear Trigger Register 2	ECMESSTC2	32	32
A007 D0C0h	ECM	ECM protection command register	ECMPCMD1	32	32
A007 D0C4h	ECM	ECM protection status register	ECMPS	8	8
A007 D0C8h	ECM	ECM pseudo error trigger register 0	ECMPE0	32	32
A007 D0CCh	ECM	ECM pseudo error trigger register 1	ECMPE1	32	32
A007 D0D0h	ECM	ECM pseudo error trigger register 2	ECMPE2	32	32
A007 D0D4h	ECM	ECM delay timer control register	ECMDTMCTL	8	8
A007 D0D8h	ECM	ECM delay timer register	ECMDTMR	16	16
A007 D0DCh	ECM	ECM delay timer compare register	ECMDTMCMP	32	32
A007 D0E0h	ECM	ECM delay timer configuration register 0	ECMDTMCFG0	32	32
A007 D0E4h	ECM	ECM delay timer configuration register 1	ECMDTMCFG1	32	32
A007 D0E8h	ECM	ECM delay timer configuration register 2	ECMDTMCFG2	32	32
A007 D0ECh	ECM	ECM delay timer configuration register 3	ECMDTMCFG3	32	32
A007 D0F0h	ECM	ECM delay timer configuration register 4	ECMDTMCFG4	32	32
A007 D0F4h	ECM	ECM delay timer configuration register 5	ECMDTMCFG5	32	32
A007 D0F8h	ECM	ECM Error Output Clear Disable Configuration Register	ECMEOCCFG	32	32
A008 0000h	CMT	Compare match timer start register 0	CMSTR0	16	16
A008 0002h	CMT0	Compare match timer control register	CMCR	16	16
A008 0004h	CMT0	Compare match counter	CMCNT	16	16
A008 0006h	CMT0	Compare match constant register	CMCOR	16	16
A008 0008h	CMT1	Compare match timer control register	CMCR	16	16
A008 000Ah	CMT1	Compare match counter	CMCNT	16	16
A008 000Ch	CMT1	Compare match constant register	CMCOR	16	16
A008 0020h	CMT	Compare match timer start register 1	CMSTR1	16	16
A008 0022h	CMT2	Compare match timer control register	CMCR	16	16
A008 0024h	CMT2	Compare match counter	CMCNT	16	16

Table 5.1 List of I/O Registers (Address Order) (50 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A008 0026h	CMT2	Compare match constant register	CMCOR	16	16
A008 0028h	CMT3	Compare match timer control register	CMCR	16	16
A008 002Ah	CMT3	Compare match counter	CMCNT	16	16
A008 002Ch	CMT3	Compare match constant register	CMCOR	16	16
A008 0040h	CMT	Compare match timer start register 2	CMSTR2	16	16
A008 0042h	CMT4	Compare match timer control register	CMCR	16	16
A008 0044h	CMT4	Compare match counter	CMCNT	16	16
A008 0046h	CMT4	Compare match constant register	CMCOR	16	16
A008 0048h	CMT5	Compare match timer control register	CMCR	16	16
A008 004Ah	CMT5	Compare match counter	CMCNT	16	16
A008 004Ch	CMT5	Compare match constant register	CMCOR	16	16
A008 0300h	CMTW0	Timer start register	CMWSTR	16	16
A008 0304h	CMTW0	Timer control register	CMWCR	16	16
A008 0308h	CMTW0	Timer I/O control register	CMWIOR	16	16
A008 0310h	CMTW0	Timer counter	CMWCNT	32	32
A008 0314h	CMTW0	Compare match constant register	CMWCOR	32	32
A008 0318h	CMTW0	Input capture register 0	CMWICR0	32	32
A008 031Ch	CMTW0	Input capture register 1	CMWICR1	32	32
A008 0320h	CMTW0	Output compare register 0	CMWOCR0	32	32
A008 0324h	CMTW0	Output compare register 1	CMWOCR1	32	32
A008 0380h	CMTW1	Timer start register	CMWSTR	16	16
A008 0384h	CMTW1	Timer control register	CMWCR	16	16
A008 0388h	CMTW1	Timer I/O control register	CMWIOR	16	16
A008 0390h	CMTW1	Timer counter	CMWCNT	32	32
A008 0394h	CMTW1	Compare match constant register	CMWCOR	32	32
A008 0398h	CMTW1	Input capture register 0	CMWICR0	32	32
A008 039Ch	CMTW1	Input capture register 1	CMWICR1	32	32
A008 03A0h	CMTW1	Output compare register 0	CMWOCR0	32	32
A008 03A4h	CMTW1	Output compare register 1	CMWOCR1	32	32
A008 0400h	CMTW	Digital noise filter control register 0	NFCR0	32	32
A008 0404h	CMTW	Digital noise filter control register 1	NFCR1	32	32
A008 0410h	CMTW	ECM dynamic mode error output select register	ECDMESLR	32	32
A008 0600h	WDT0	WDT refresh register	WDTRR	8	8
A008 0602h	WDT0	WDT control register	WDTCR	16	16
A008 0604h	WDT0	WDT status register	WDTSR	16	16
A008 0700h	IWDT	IWDT refresh register	IWDTRR	8	8
A008 0702h	IWDT	IWDT control register	IWDTCR	16	16
A008 0704h	IWDT	IWDT status register	IWDTSR	16	16
A008 0706h	IWDT	IWDT reset control register	IWDTRCR	8	8
A008 0940h	RIIC1	I2C bus control register 1	ICCR1	8	8
A008 0941h	RIIC1	I2C bus control register 2	ICCR2	8	8
A008 0942h	RIIC1	I2C bus mode register 1	ICMR1	8	8
A008 0943h	RIIC1	I2C bus mode register 2	ICMR2	8	8
A008 0944h	RIIC1	I2C bus mode register 3	ICMR3	8	8
A008 0945h	RIIC1	I2C bus function enable register	ICFER	8	8
A008 0946h	RIIC1	I2C bus status enable register	ICSER	8	8
A008 0947h	RIIC1	I2C bus interrupt enable register	ICIER	8	8
A008 0948h	RIIC1	I2C bus status register 1	ICSR1	8	8

Table 5.1 List of I/O Registers (Address Order) (51 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A008 0949h	RIIC1	I²C bus status register 2	ICSR2	8	8
A008 094Ah	RIIC1	Slave address register L0	ICSARL0	8	8
A008 094Bh	RIIC1	Slave address register U0	ICSARU0	8	8
A008 094Ch	RIIC1	Slave address register L1	ICSARL1	8	8
A008 094Dh	RIIC1	Slave address register U1	ICSARU1	8	8
A008 094Eh	RIIC1	Slave address register L2	ICSARL2	8	8
A008 094Fh	RIIC1	Slave address register U2	ICSARU2	8	8
A008 0950h	RIIC1	I²C bus bitrate low register	ICBRL	8	8
A008 0951h	RIIC1	I²C bus bitrate high register	ICBRH	8	8
A008 0952h	RIIC1	I²C bus transmit data register	ICDRT	8	8
A008 0953h	RIIC1	I²C bus receive data register	ICDRR	8	8
A008 0B00h	ELC	Event link control register	ELCR	8	8
A008 0B08h	ELC	Event link setting register 7	ELSR7	8	8
A008 0B13h	ELC	Event link setting register 18	ELSR18	8	8
A008 0B14h	ELC	Event link setting register 19	ELSR19	8	8
A008 0B15h	ELC	Event link setting register 20	ELSR20	8	8
A008 0B16h	ELC	Event link setting register 21	ELSR21	8	8
A008 0B17h	ELC	Event link setting register 22	ELSR22	8	8
A008 0B18h	ELC	Event link setting register 23	ELSR23	8	8
A008 0B19h	ELC	Event link setting register 24	ELSR24	8	8
A008 0B1Ah	ELC	Event link setting register 25	ELSR25	8	8
A008 0B1Bh	ELC	Event link setting register 26	ELSR26	8	8
A008 0B1Ch	ELC	Event link setting register 27	ELSR27	8	8
A008 0B21h	ELC	Event link option setting register C	ELOPC	8	8
A008 0B23h	ELC	Port group setting register 1	PGR1	8	8
A008 0B24h	ELC	Port group setting register 2	PGR2	8	8
A008 0B25h	ELC	Port group control register 1	PGC1	8	8
A008 0B26h	ELC	Port group control register 2	PGC2	8	8
A008 0B27h	ELC	Port buffer register 1	PDBF1	8	8
A008 0B28h	ELC	Port buffer register 2	PDBF2	8	8
A008 0B29h	ELC	Event link port setting register 0	PEL0	8	8
A008 0B2Ah	ELC	Event link port setting register 1	PEL1	8	8
A008 0B2Bh	ELC	Event link port setting register 2	PEL2	8	8
A008 0B2Ch	ELC	Event link port setting register 3	PEL3	8	8
A008 0B2Dh	ELC	Event link software event generation register	ELSEGR	8	8
A008 0B31h	ELC	Event link setting register 33	ELSR33	8	8
A008 0B41h	ELC	Event link option setting register H	ELOPH	8	8
A009 0000h	CLMA0	CLMA0 control register 0	CLMA0CTL0	8	8
A009 0008h	CLMA0	CLMA0 compare register L	CLMA0CMPL	16	16
A009 000Ch	CLMA0	CLMA0 compare register H	CLMA0CMPH	16	16
A009 0010h	CLMA0	CLMA0 command register	CLMA0PCMD	8	8
A009 0014h	CLMA0	CLMA0 protection status register	CLMA0PS	8	8
A009 0020h	CLMA1	CLMA1 control register 0	CLMA1CTL0	8	8
A009 0028h	CLMA1	CLMA1 compare register L	CLMA1CMPL	16	16
A009 002Ch	CLMA1	CLMA1 compare register H	CLMA1CMPH	16	16
A009 0030h	CLMA1	CLMA1 command register	CLMA1PCMD	8	8
A009 0034h	CLMA1	CLMA1 protection status register	CLMA1PS	8	8
A009 0040h	CLMA2	CLMA2 control register 0	CLMA2CTL0	8	8

Table 5.1 List of I/O Registers (Address Order) (52 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A009 0048h	CLMA2	CLMA2 compare register L	CLMA2CMPL	16	16
A009 004Ch	CLMA2	CLMA2 compare register H	CLMA2CMPH	16	16
A009 0050h	CLMA2	CLMA2 command register	CLMA2PCMD	8	8
A009 0054h	CLMA2	CLMA2 protection status register	CLMA2PS	8	8
A009 4000h	DMA0	DMAC unit 0 source select register 0	DMA0SEL0	32	32
A009 4004h	DMA0	DMAC unit 0 source select register 1	DMA0SEL1	32	32
A009 4008h	DMA0	DMAC unit 0 source select register 2	DMA0SEL2	32	32
A009 400Ch	DMA0	DMAC unit 0 source select register 3	DMA0SEL3	32	32
A009 4010h	DMA0	DMAC unit 0 source select register 4	DMA0SEL4	32	32
A009 4014h	DMA0	DMAC unit 0 source select register 5	DMA0SEL5	32	32
A009 4018h	DMA0	DMAC unit 0 source select register 6	DMA0SEL6	32	32
A009 401Ch	DMA0	DMAC unit 0 source select register 7	DMA0SEL7	32	32
A009 4020h	DMA0	DMAC unit 0 source select register 8	DMA0SEL8	32	32
A009 4024h	DMA0	DMAC unit 0 source select register 9	DMA0SEL9	32	32
A009 4028h	DMA0	DMAC unit 0 source select register 10	DMA0SEL10	32	32
A009 402Ch	DMA0	DMAC unit 0 source select register 11	DMA0SEL11	32	32
A009 4030h	DMA0	DMAC unit 0 source select register 12	DMA0SEL12	32	32
A009 4034h	DMA0	DMAC unit 0 source select register 13	DMA0SEL13	32	32
A009 4038h	DMA0	DMAC unit 0 source select register 14	DMA0SEL14	32	32
A009 403Ch	DMA0	DMAC unit 0 source select register 15	DMA0SEL15	32	32
A009 4040h	DMA1	DMAC unit 1 source select register 0	DMA1SEL0	32	32
A009 4044h	DMA1	DMAC unit 1 source select register 1	DMA1SEL1	32	32
A009 4048h	DMA1	DMAC unit 1 source select register 2	DMA1SEL2	32	32
A009 404Ch	DMA1	DMAC unit 1 source select register 3	DMA1SEL3	32	32
A009 4050h	DMA1	DMAC unit 1 source select register 4	DMA1SEL4	32	32
A009 4054h	DMA1	DMAC unit 1 source select register 5	DMA1SEL5	32	32
A009 4058h	DMA1	DMAC unit 1 source select register 6	DMA1SEL6	32	32
A009 405Ch	DMA1	DMAC unit 1 source select register 7	DMA1SEL7	32	32
A009 4060h	DMA1	DMAC unit 1 source select register 8	DMA1SEL8	32	32
A009 4064h	DMA1	DMAC unit 1 source select register 9	DMA1SEL9	32	32
A009 4068h	DMA1	DMAC unit 1 source select register 10	DMA1SEL10	32	32
A009 406Ch	DMA1	DMAC unit 1 source select register 11	DMA1SEL11	32	32
A009 4070h	DMA1	DMAC unit 1 source select register 12	DMA1SEL12	32	32
A009 4074h	DMA1	DMAC unit 1 source select register 13	DMA1SEL13	32	32
A009 4078h	DMA1	DMAC unit 1 source select register 14	DMA1SEL14	32	32
A009 407Ch	DMA1	DMAC unit 1 source select register 15	DMA1SEL15	32	32
A009 4080h	DMAC	DMAC software start register	DMASTG	32	32
A009 4200h	ICU	IRQ control register 0	IRQCR0	32	32
A009 4204h	ICU	IRQ control register 1	IRQCR1	32	32
A009 4208h	ICU	IRQ control register 2	IRQCR2	32	32
A009 420Ch	ICU	IRQ control register 3	IRQCR3	32	32
A009 4210h	ICU	IRQ control register 4	IRQCR4	32	32
A009 4218h	ICU	IRQ control register 6	IRQCR6	32	32
A009 421Ch	ICU	IRQ control register 7	IRQCR7	32	32
A009 4224h	ICU	IRQ control register 9	IRQCR9	32	32
A009 422Ch	ICU	IRQ control register 11	IRQCR11	32	32
A009 4230h	ICU	IRQ control register 12	IRQCR12	32	32
A009 4234h	ICU	IRQ control register 13	IRQCR13	32	32

Table 5.1 List of I/O Registers (Address Order) (53 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A009 4238h	ICU	IRQ control register 14	IRQCR14	32	32
A009 4240h	ICU	IRQ pin digital noise filter enable register	IRQFLTE	32	32
A009 4244h	ICU	IRQ pin digital noise filter setting register	IRQFLTC	32	32
A009 4248h	ICU	Non-maskable interrupt status register	NMISR	32	32
A009 424Ch	ICU	Non-maskable interrupt status clear register	NMICLR	32	32
A009 4250h	ICU	NMI pin interrupt control register	NMICR	32	32
A009 4254h	ICU	NMI pin digital noise filter enable register	NMIFLTE	32	32
A009 4258h	ICU	NMI pin digital noise filter setting register	NMIFLTC	32	32
A009 425Ch	ICU	EtherPHY control register 0	EPHYCR0	32	32
A009 4260h	ICU	EtherPHY control register 1	EPHYCR1	32	32
A009 4268h	ICU	EtherPHY interrupt request pin digital noise filter enable register	EPHYFLTE	32	32
A009 426Ch	ICU	EtherPHY interrupt request pin digital noise filter setting register	EPHYFLTC	32	32
A00B 0020h	SYSTEM	System clock control register	SCKCR	32	32
A00B 0024h	SYSTEM	System clock control register 2	SCKCR2	32	32
A00B 0038h	SYSTEM	PLL1 control register 2	PLL1CR2	32	32
A00B 0040h	SYSTEM	Low-speed on-chip oscillator control register	LOCOCR	32	32
A00B 004Ch	SYSTEM	Oscillation stop detect control register	OSTDCR	32	32
A00B 0200h	SYSTEM	Reset status register 0	RSTSR0	32	32
A00B 0210h	SYSTEM	Software reset register	SWRR1	32	32
A00B 0248h	SYSTEM	Module reset control register C	MRCTLC	32	32
A00B 0300h	SYSTEM	Module stop control register A	MSTPCRA	32	32
A00B 0304h	SYSTEM	Module stop control register B	MSTPCRB	32	32
A00B 0308h	SYSTEM	Module stop control register C	MSTPCRC	32	32
A00B 0310h	SYSTEM	Module stop control register E	MSTPCRE	32	32
A00B 0314h	SYSTEM	Module stop control register F	MSTPCRF	32	32
A00B 0800h	SYSTEM	ATCM wait control register	SYTATCMWAIT	32	32
A00B 0A00h	SYSTEM	Debug interface control register	DBGIFCNT	32	32
A00B 0A80h	SYSTEM	ECM mask control register	ECMMCNT	32	32
A00B 0B00h	SYSTEM	Protect register	PRCR	32	32
A00B F000h	ESC	Ethernet system protect command register	ETSPCMD	32	32
A00B F004h	ESC	MAC select register	MACSEL	32	32
A00B F014h	ESC	Ethernet PHY LINK mode register	ETHPHYLNK	32	32
A00B F100h	ESC	EtherCAT PHY Offset Address Setting Register	CATOFFADD	32	32
A00B F104h	ESC	EtherCAT Operation Mode Setting Register	CATEMMD	32	32
A00B F10Ch	ESC	EtherCAT TXC Shift Setting Register	CATTXCSFT	32	32
A00B F118h	ESC	Ethernet peripheral reset register	ETHSFTRST	32	32
A00D 0000h	ESC	Type Register	TYPE	8	8
A00D 0001h	ESC	Revision Register	REVISION	8	8
A00D 0002h	ESC	Build Register	BUILD	16	16
A00D 0004h	ESC	FMMU Supported Register	FMMU_NUM	8	8
A00D 0005h	ESC	SyncManager Supported Register	SYNC_MANAGER	8	8
A00D 0006h	ESC	RAM Size Register	RAM_SIZE	8	8
A00D 0007h	ESC	Port Descriptor Register	PORT_DESC	8	8
A00D 0008h	ESC	ESC Features Supported Register	FEATURE	16	16
A00D 0010h	ESC	Configured Station Address Register	STATION_ADR	16	16
A00D 0012h	ESC	Configured Station Alias Register	STATION_ALIAS	16	16
A00D 0020h	ESC	Write Register Enable Register	WR_REG_ENABLE	8	8
A00D 0021h	ESC	Write Register Protection Register	WR_REG_PROTECT	8	8

Table 5.1 List of I/O Registers (Address Order) (54 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A00D 0030h	ESC	ESC Write Enable Register	ESC_WR_ENABLE	8	8
A00D 0031h	ESC	ESC Write Protection Register	ESC_WR_PROTECT	8	8
A00D 0040h	ESC	ESC Reset ECAT Register	ESC_RESET_ECAT	8	8
A00D 0041h	ESC	ESC Reset PDI Register	ESC_RESET_PDI	8	8
A00D 0100h	ESC	ESC DL Control Register	ESC_DL_CONTROL	32	32
A00D 0108h	ESC	Physical Read/Write Offset Register	PHYSICAL_RW_OFFSET	16	16
A00D 0110h	ESC	ESC DL Status Register	ESC_DL_STATUS	16	16
A00D 0120h	ESC	AL Control Register	AL_CONTROL	16	16
A00D 0130h	ESC	AL Status Register	AL_STATUS	16	16
A00D 0134h	ESC	AL Status Code Register	AL_STATUS_CODE	16	16
A00D 0138h	ESC	RUN LED Override Register	RUN_LED_OVERRIDE	8	8
A00D 0139h	ESC	ERR LED Override Register	ERR_LED_OVERRIDE	8	8
A00D 0140h	ESC	PDI Control Register	PDI_CONTROL	8	8
A00D 0141h	ESC	ESC Configuration Register	ESC_CONFIG	8	8
A00D 0150h	ESC	PDI Configuration Register	PDI_CONFIG	8	8
A00D 0151h	ESC	SYNC/LATCH PDI Configuration Register	SYNC_LATCH_CONFIG	16	16
A00D 0152h	ESC	Extended PDI Configuration Register	EXT_PDI_CONFIG	16	16
A00D 0200h	ESC	ECAT Event Mask Register	ECAT_EVENT_MASK	16	16
A00D 0204h	ESC	AL Event Mask Register	AL_EVENT_MASK	32	32
A00D 0210h	ESC	ECAT Event Request Register	ECAT_EVENT_REQ	16	16
A00D 0220h	ESC	AL Event Request Register	AL_EVENT_REQ	32	32
A00D 0300h + 0002h*n	ESC	Rx Error Counter n Register	RX_ERR_COUNTn	16	16
A00D 0308h + 0001h*n	ESC	Forwarded Rx Error Counter n Register	FWD_RX_ERR_COUNTn	8	8
A00D 030Ch	ESC	ECAT Processing Unit Error Counter Register	ECAT_PROC_ERR_COUNT	8	8
A00D 030Dh	ESC	PDI Error Counter Register	PDI_ERR_COUNT	8	8
A00D 0310h + 0001h*n	ESC	Lost Link Counter n Register	LOST_LINK_COUNTn	8	8
A00D 0400h	ESC	Watchdog Divider Register	WD_DIVIDE	16	16
A00D 0410h	ESC	Watchdog Time PDI Register	WDT_PDI	16	16
A00D 0420h	ESC	Watchdog Time Process Data Register	WDT_DATA	16	16
A00D 0440h	ESC	Watchdog Status Process Data Register	WDS_DATA	16	16
A00D 0442h	ESC	Watchdog Counter Process Data Register	WDC_DATA	8	8
A00D 0443h	ESC	Watchdog Counter PDI Register	WDC_PDI	8	8
A00D 0500h	ESC	EEPROM Configuration Register	EEP_CONF	8	8
A00D 0501h	ESC	EEPROM PDI Access State Register	EEP_STATE	8	8
A00D 0502h	ESC	EEPROM Control/Status Register	EEP_CONT_STAT	16	16
A00D 0504h	ESC	EEPROM Address Register	EEP_ADR	32	32
A00D 0508h	ESC	EEPROM Data Register	EEP_DATA	32	32
A00D 0510h	ESC	MII Management Control/Status Register	MII_CONT_STAT	16	16
A00D 0512h	ESC	PHY Address Register	PHY_ADR	8	8
A00D 0513h	ESC	PHY Register Address Register	PHY_REG_ADR	8	8
A00D 0514h	ESC	PHY Data Register	PHY_DATA	16	16
A00D 0516h	ESC	MII Management ECAT Access State Register	MII_ECAT_ACS_STAT	8	8
A00D 0517h	ESC	MII Management PDI Access State Register	MII_PDI_ACS_STAT	8	8
A00D 0600h + 0010h*m	ESC	FMMU Logical Start Address m Register	FMMUm_L_START_ADR	32	32

Table 5.1 List of I/O Registers (Address Order) (55 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A00D 0604h + 0010h*m	ESC	FMMU Length m Register	FMMUm_LEN	16	16
A00D 0606h + 0010h*m	ESC	FMMU Logical Start Bit m Register	FMMUm_L_START_BIT	8	8
A00D 0607h + 0010h*m	ESC	FMMU Logical Stop Bit m Register	FMMUm_L_STOP_BIT	8	8
A00D 0608h + 0010h*m	ESC	FMMU Physical Start Address m Register	FMMUm_P_START_ADR	16	16
A00D 060Ah + 0010h*m	ESC	FMMU Physical Start Bit m Register	FMMUm_P_START_BIT	8	8
A00D 060Bh + 0010h*m	ESC	FMMU Type m Register	FMMUm_TYPE	8	8
A00D 060Ch + 0010h*m	ESC	FMMU Activate m Register	FMMUm_ACT	8	8
A00D 0800h + 0008h*m	ESC	SyncManager Physical Start Address m Register	SMm_P_START_ADR	16	16
A00D 0802h + 0008h*m	ESC	SyncManager Length m Register	SMm_LEN	16	16
A00D 0804h + 0008h*m	ESC	SyncManager Control m Register	SMm_CONTROL	8	8
A00D 0805h + 0008h*m	ESC	SyncManager Status m Register	SMm_STATUS	8	8
A00D 0806h + 0008h*m	ESC	SyncManager Activate m Register	SMm_ACT	8	8
A00D 0807h + 0008h*m	ESC	SyncManager PDI Control m Register	SMm_PDI_CONT	8	8
A00D 0900h	ESC	Receive Time Port 0 Register	DC_RCV_TIME_PORT0	32	32
A00D 0904h	ESC	Receive Time Port 1 Register	DC_RCV_TIME_PORT1	32	32
A00D 0910h	ESC	System Time Register	DC_SYS_TIME	64	32
A00D 0918h	ESC	Receive Time ECAT Processing Unit Register	DC_RCV_TIME_UNIT	64	32
A00D 0920h	ESC	System Time Offset Register	DC_SYS_TIME_OFFSET	64	32
A00D 0928h	ESC	System Time Delay Register	DC_SYS_TIME_DELAY	32	32
A00D 092Ch	ESC	System Time Difference Register	DC_SYS_TIME_DIFF	32	32
A00D 0930h	ESC	Speed Counter Start Register	DC_SPEED_COUNT_START	16	16
A00D 0932h	ESC	Speed Counter Difference Register	DC_SPEED_COUNT_DIFF	16	16
A00D 0934h	ESC	System Time Difference Filter Depth Register	DC_SYS_TIME_DIFF_FIL_DEPTH	8	8
A00D 0935h	ESC	Speed Counter Filter Depth Register	DC_SPEED_COUNT_FIL_DEPTH	8	8
A00D 0980h	ESC	Cyclic Unit Control Register	DC_CYC_CONT	8	8
A00D 0981h	ESC	Activation Register	DC_ACT	8	8
A00D 0982h	ESC	SYNC Signal Pulse Length Register	DC_PULSE_LEN	16	16
A00D 0984h	ESC	Activation Status Register	DC_ACT_STAT	8	8
A00D 098Eh	ESC	SYNC0 Status Register	DC_SYNC0_STAT	8	8
A00D 098Fh	ESC	SYNC1 Status Register	DC_SYNC1_STAT	8	8
A00D 0990h	ESC	Start Time Cyclic Operation/Next SYNC0 Pulse Register	DC_CYC_START_TIME	64	32
A00D 0998h	ESC	Next SYNC1 Pulse Register	DC_NEXT_SYNC1_PULSE	64	32
A00D 09A0h	ESC	SYNC0 Cycle Time Register	DC_SYNC0_CYC_TIME	32	32
A00D 09A4h	ESC	SYNC1 Cycle Time Register	DC_SYNC1_CYC_TIME	32	32
A00D 09A8h	ESC	Latch 0 Control Register	DC_LATCH0_CONT	8	8
A00D 09A9h	ESC	Latch 1 Control Register	DC_LATCH1_CONT	8	8
A00D 09AEh	ESC	Latch 0 Status Register	DC_LATCH0_STAT	8	8

Table 5.1 List of I/O Registers (Address Order) (56 / 56)

Address	Module Symbol	Register Name	Register Symbol	Number of Bits	Access Size
A00D 09AFh	ESC	Latch 1 Status Register	DC_LATCH1_STAT	8	8
A00D 09B0h	ESC	Latch 0 Time Positive Edge Register	DC_LATCH0_TIME_POS	64	32
A00D 09B8h	ESC	Latch 0 Time Negative Edge Register	DC_LATCH0_TIME_NEG	64	32
A00D 09C0h	ESC	Latch 1 Time Positive Edge Register	DC_LATCH1_TIME_POS	64	32
A00D 09C8h	ESC	Latch 1 Time Negative Edge Register	DC_LATCH1_TIME_NEG	64	32
A00D 09F0h	ESC	Buffer Change Event Time Register	DC_ECAT_CNG_EV_TIME	32	32
A00D 09F8h	ESC	PDI Buffer Start Event Time Register	DC_PDI_START_EV_TIME	32	32
A00D 09FCh	ESC	PDI Buffer Change Event Time Register	DC_PDI_CNG_EV_TIME	32	32
A00D 0E00h	ESC	Product ID Register	PRODUCT_ID	64	32
A00D 0E08h	ESC	Vendor ID Register	VENDOR_ID	64	32
A00D 0F80h – A00D 0FFFh	ESC	User RAM	USER_RAM	1024	8, 16, 32
A00D 1000h – A00D 2FFFh	ESC	Process Data RAM	DATA_RAM	65536	8, 16, 32
A00F 00A0h	ESC	MIIM register	GMAC_MIIM	32	32
A00F 2100h	ESC	System protect command register	SPCMD	32	32
A00F 2110h	ESC	MDIO controller reset register	EMACRST	32	32

6. Reset

6.1 Overview

Available reset types are RES# pin reset, error control module (ECM) reset, and software reset.
Table 6.1 lists the reset names and sources.

Table 6.1 Reset Names and Sources

Reset Name	Source
RES# pin reset	Voltage input to the RES# pin is Low.
ECM reset	Reset request from the error control module (ECM)
Software reset	SWRR1 register setting

The internal states and pins are initialized by reset.
Table 6.2 lists the reset targets to be initialized for each reset type. For details on reset control during debugging, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.

Table 6.2 Targets to Be Initialized for Each Reset Type (√: To be initialized, —: No change)

	Reset Source		
Reset Target	**RES# Pin Reset**	**ECM Reset**	**Software Reset**
RES# pin reset flag (RSTSR0.TRF)	—	√	√
ECM reset detect flag (RSTSR0.ECMRF)	√	—	√
Software reset detect flag (RSTSR0.SWRF1)	√	√	—
Pin state	√	√	√
ECM ECM master error source status registers 0 to 2 ECM checker error source status registers 0 to 2 ECM error output clear disable setting register	√	—	—
Registers other than the above, and internal state	√	√	√
RSTOUT# pin output	√ (Low)*1	√ (Low)*1	√ (Low)*1
ERROROUT# pin output	√ (Low)	*2	—

Note 1. For the Low output period, see section 6.3.5, Reset Output Pin (RSTOUT#).
Note 2. Depends on the ECM setting. For details, see section 32, Error Control Module (ECM).

Table 6.3 lists the input and output pins related to the reset.

Table 6.3 Input and Output Pins Related to Reset

Pin Name	I/O	Function
RES#	Input	Reset pin. Use this pin to reset the entire LSI except the debugging circuit and TAP (Test Access Port). Since the power-on reset circuit is not incorporated in EC-1, a reset circuit must be implemented outside this LSI. For an example of configuration of the external reset circuit, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.
TRST#	Input	Test reset pin. Use this pin to reset TAP. If you design a board which enables an emulator, set the TRST# pin to Low level during the same period as the RES# pin at the time of power-on. The TRST# pin should also be controllable independently. When unused, this pin must be set to low level or connected to the same signal as the RES# pin. For details, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.
RSTOUT#	Output	Reset output pin. This pin outputs low-level signal upon occurrence of reset. For details, see section 6.3.5, Reset Output Pin (RSTOUT#). This pin can be used for resetting the external device.

Note: For details on resetting the debugging circuit, see section 10, Debugging Interface.

6.2 Register Descriptions

The reset status register 0 contains bits assigned to respective reset types to indicate reset generation sources. RSTSR0 and SWRR1 are protected by the register write protection function. To write these registers, clear the write protection bit 1 in the protect register (PRCR). For details on the register write protection, see section 11, Register Write Protection Function.

6.2.1 Reset Status Register 0 (RSTSR0)

RSTSR0 is the register that indicates reset generation sources.

Address(es): RSTSR0: A00B 0200h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	SWR1F	ECMRF	TRF	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	*1	*1	*1	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/(W)
b1	TRF	RES# Pin Reset Detect Flag	0: RES# pin reset not detected 1: RES# pin reset detected [Setting condition] • When Low is input to the RES# pin. [Clearing conditions] • When a ECM reset or software reset occurs. • When “0000 0000h” is written to RSTSR0 after RSTSR0 is read.	R/(W)*2
b2	ECMRF	ECM Reset Detect Flag	0: ECM reset not detected 1: ECM reset detected [Setting condition] • When an error predefined as a reset source by ECM setting occurs. [Clearing conditions] • When an RES# pin reset or software reset occurs. • When “0000 0000h” is written to RSTSR0 after RSTSR0 is read.	R/(W)*2
b3	SWR1F	Software Reset Detect Flag	0: Software reset not detected. 1: Software reset detected. [Setting condition] • When a software reset occurs. [Clearing conditions] • When an RES# pin reset or ECM reset occurs. • When “0000 0000h” is written to RSTSR0 after RSTSR0 is read.	R/(W)*2
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/(W)

Note 1. The value after reset depends on the reset source.
Note 2. Only 0 can be written to clear the flag.

6.2.2 Software Reset Register (SWRR1)

SWRR1 is a register that controls the software reset.

Address(es): SWRR1: A00B 0210h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SWRR1[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SWRR1[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	SWRR1[31:0]	Software Reset	When “4321 A501h” is written, a software reset occurs. These bits are read as 0000 0000h.	R/W

6.2.3 Module Reset Control Register C (MRCTLC)

MRCTLC is a register that controls the reset of peripheral modules.

Address(es): MRCTLC: A00B 0248h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MRUSBH	MRUSBF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	MRUSBF	USB (Func) Reset Control	0: USB (Func) is released from reset. 1: USB (Func) is in the reset state.	R/W
b1	MRUSBH	USB (Host) Reset Control	0: USB (Host) is released from reset. 1: USB (Host) is in the reset state.	R/W
b31 to b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

6.3 Operation

6.3.1 RES# Pin Reset

This reset occurs when a signal arrives at the RES# pin from the externally connected reset circuit. When the signal becomes Low at the RES# pin, all the ongoing processes are aborted and the LSI enters the reset state. In order to reset the LSI without fail, the RES# pin should be held at Low level for the specified time after power-on. For details on the reset configuration, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.
After the reset is released, the CPU (Cortex-R4) starts reset exception handling.
When an RES# pin reset occurs, the RSTSR0.TRF flag is set to 1.

6.3.2 ECM Reset

This reset is generated in response to a reset request from ECM (Error Control Module).
ECM receives serious errors, such as oscillation stop detection, from individual modules in the LSI, and generates reset requests corresponding to respective errors. For details on the ECM operation, see section 32, Error Control Module (ECM). When an ECM reset occurs, all the ongoing processes are aborted and the LSI enters the reset state. After the reset is released, the CPU (Cortex-R4) starts reset exception handling.
When an ECM reset occurs, the RSTSR0.ECMRF flag is set to 1.

6.3.3 Software Reset

The software reset occurs when “4321 A501h” is written to the SWRR1 register. When a software reset occurs, all the ongoing processes are aborted and the LSI enters the reset state. After the reset is released, the CPU (Cortex-R4) starts reset exception handling.
When the software reset occurs, the RSTSR0.SWR1F flag is set to 1.

6.3.4 Determination of Reset Generation Source

Reading the RSTSR0 register determines which reset source was used for reset execution. Figure 6.1 shows an example of the flow to identify a reset generation source.

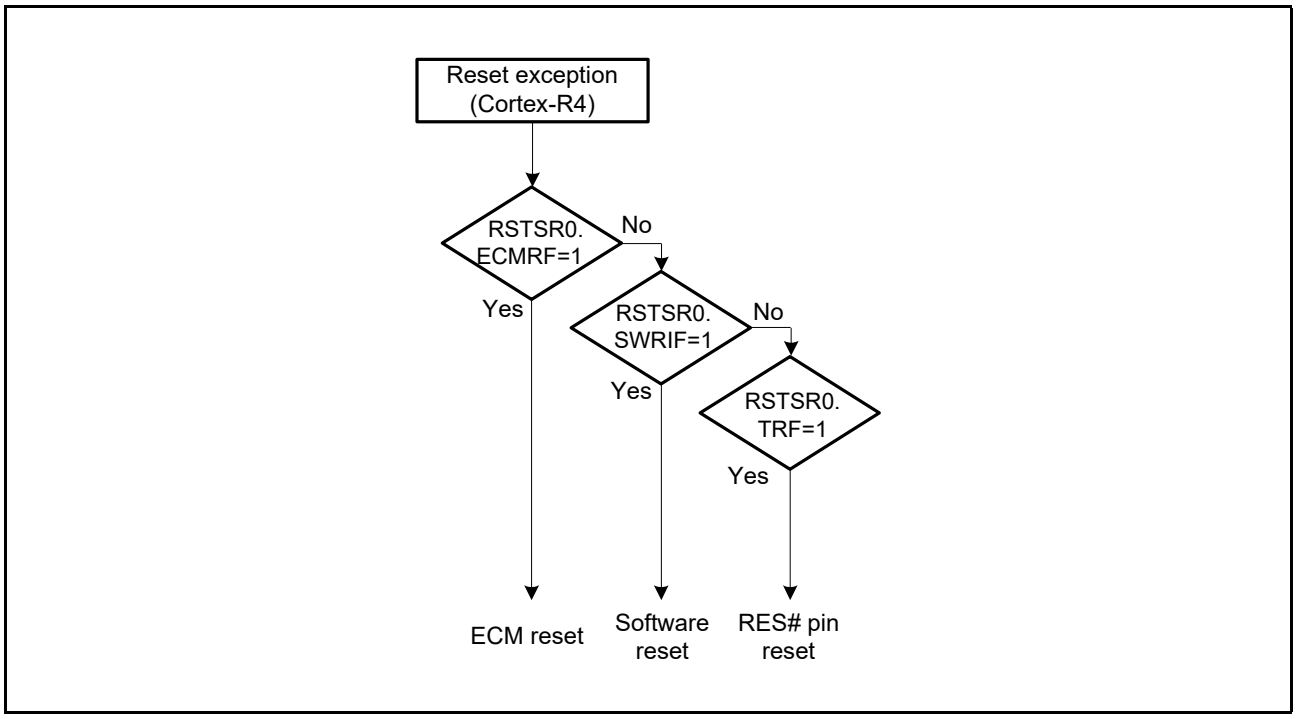


Figure 6.1 Example of Reset Generation Source Determination Flow

6.3.5 Reset Output Pin (RSTOUT#)

The reset output pin (RSTOUT#) outputs Low level upon occurrence of reset when the input level of the RES# pin is Low. It also outputs Low level upon occurrence of an ECM reset or software reset.
If the RES# pin remains Low for a specified time period and then changes to high, the reset output remains Low for 500 us (Typ.) and then changes to High. Similarly, when an ECM reset or software reset occurs, the reset output remains Low for 500 us (Typ.) and then changes to High.

6.3.6 Noise Cancellation for Reset Input

Noise cancellation using analog delay is applied to the RES# pin. This can eliminate noises within 100 ns (Min.).

6.4 Usage Note

6.4.1 Connection of Reset Output Pin (RSTOUT#)

The low level is output for a specified period of time on RSTOUT# after de-assertion of the signal on the RES# pin. This means that RSTOUT# should not be directly connected as a reset signal for a flash memory to be used in booting the LSI chip. Otherwise, release of the LSI chip itself from the reset state may precede that for the flash memory.
However, using RSTOUT# as the reset signal for external device is still possible as long as the timing requirement imposed by the output on the RSTOUT# pin is satisfied.
For details, see section 6.3.5, Reset Output Pin (RSTOUT#).

7. Clock Generation Circuit

7.1 Overview

This LSI incorporates a clock generation circuit.

Table 7.1 lists the specifications of the clock generation circuit. Figure 7.1 shows a block diagram of the clock generation circuit.

Table 7.1 Specifications of Clock Generation Circuit

Item	Specifications
Main clock oscillator	Resonator frequency: 25 MHz
	Connectable resonator or additional circuit: Ceramic resonator, crystal resonator Connection pin: EXTAL, XTAL
	Oscillation stop detection function: When an oscillation stop is detected with the main clock, the system clock source is switched to LOCO.
PLL0 circuit	Input clock source: Main clock oscillator
	Input frequency: 25 MHz
	Frequency multiplication ratio: 48 multiplication
	Output clock frequency of the PLL0 circuit: 1200 MHz
	Oscillation abnormality detection function: When the CLMA0 detects an abnormality in oscillation of the PLL0, the system clock source is switched to main clock.
PLL1 circuit	Input clock source: PLL0 clock divided by 80
	Input frequency: 15 MHz
	Frequency multiplication ratio: 80
	Output clock frequency of the PLL1 circuit: 1200 MHz
	Oscillation abnormality detection function: When the CLMA1 detects an abnormality in oscillation of the PLL1, the system clock source is switched to main clock.
Low-speed on-chip oscillator (LOCO)	Oscillation frequency: 240 kHz
	Oscillation abnormality detection function: The CLMA2 can detect an abnormality in oscillation of the LOCO.
External clock input (TCK) for JTAG	Input frequency: 50 MHz (max.)

Table 7.2 Specifications of Clock Generation Circuit (Internal Clock)

Item	Clock Source	Supplied to	Frequency
CPU clock (CPUCLK)	Selected from frequency-dividing clocks for PLL0 or PLL1	CPU (Cortex-R4)	150 MHz
System clock (ICLK)	Selected from frequency-dividing clocks for PLL0 or PLL1	DMAC, interrupt controller	150 MHz
High-speed peripheral module clock (PCLKA)	Selected from frequency-dividing clocks for PLL0 or PLL1	Peripheral module	150 MHz
Low-speed peripheral module clock (PCLKB)	Selected from frequency-dividing clocks for PLL0 or PLL1	Peripheral module	75 MHz
Low-speed peripheral module clock (PCLKD)	Frequency-dividing clock for PLL0	Peripheral modules (CRC, ECM, ELC, CMT, CMTW, and RIICa)	75 MHz
Low-speed peripheral module clock (PCLKE)	Frequency-dividing clock for PLL0	Peripheral module (WDTA)	Up to 75 MHz
High-speed serial clock (SERICLK)	Frequency-dividing clock for PLL0	RSPIa, SCIFA	150 MHz, 120 MHz
USB clock M (USBMCLK)	Frequency-dividing clock for PLL0	USB PHY	50 MHz
USB clock P (USBPCLK)	Frequency-dividing clock for PLL0	USB	30 MHz
Ethernet clock A (ETCLKA)	Frequency-dividing clock for PLL0	EtherCAT	100 MHz
Ethernet clock D (ETCLKD)	Frequency-dividing clock for PLL0	MDIO controller (MDC_CLK)	Up to 12.5 MHz
Ethernet clock E (ETCLKE)	Main clock	Ethernet PHY	25 MHz
Ethernet clock F (ETCLKF)	Frequency-dividing clock for PLL0	EtherCAT	25 MHz
Clock A for RSCAN (CANCLKA)	Frequency-dividing clock for PLL0	RSCAN	24 MHz
Clock B for RSCAN (CANCLKB)	Main clock	RSCAN	25 MHz
CLMAn sampling clock (CLMAMCLKA) (n = 1 or 0)	Main clock divided by 2	CLMA0, CLMA1	12.5 MHz
CLMA2 sampling clock (CLMAMCLKB)	Main clock divided by 256	CLMA2	97.6 kHz
CLMA2 monitor clock (CLMALCLK)	LOCO	CLMA2	240 kHz
CLMA0 monitor clock (CLMAPLCLK0)	PLL0 clock divided by 16	CLMA0	75 MHz
CLMA1 monitor clock (CLMAPLCLK1)	PLL1 clock divided by 16	CLMA1	75 MHz
IWDT clock (IWDTCLK)	LOCO clock divided by 2	IWDT	120 kHz
ECM clock (ECMCKL)	LOCO	ECM	240 kHz
JTAG clock (JTAGTCK)	TCK	JTAG	Up to 50 MHz
Trace interface clock (TCLK)	Selected from frequency-dividing clocks for PLL0 or PLL1	CoreSight TPIU	75 MHz

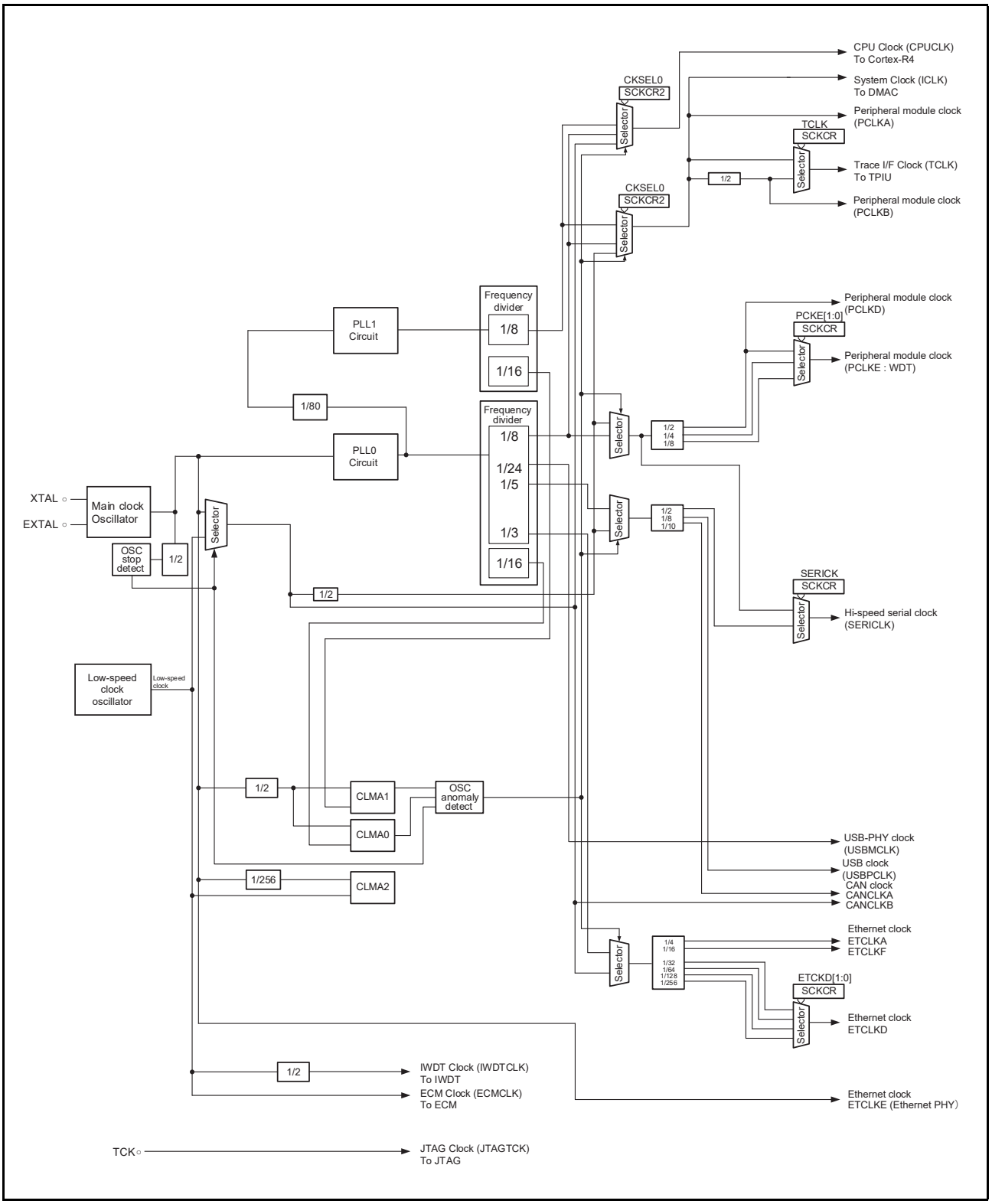


Figure 7.1 Block Diagram of Clock Generation Circuit

Table 7.3 shows the input/output pins of the clock generation circuit.

Table 7.3 Pin Configuration of Clock Generation Circuit

Pin Name	I/O	Description
XTAL	Output	These pins are used to connect a crystal resonator.
EXTAL	Input	
TCK	Input	This pin is used to input the clock for the JTAG.
CLKOUT25M0	Output	This pin is used to supply EtherPHY0 with the main clock (25 MHz) or frequency-dividing clock of PLL0 (50 MHz).
CLKOUT25M1	Output	This pin is used to supply EtherPHY1 with the main clock (25 MHz) or frequency-dividing clock of PLL0 (50 MHz).

7.2 Register Descriptions

The registers related to the clock generation circuit can be write-protected. To write to the registers, specify bit 0 of the Protect Register (PRCR) to cancel the write protection. For details, see section 11, Register Write Protection Function.

7.2.1 System Clock Control Register (SCKCR)

The SCKCR register is used to select the frequency for each of the trace interface clock (TCLK), high-speed serial clock (SERICLK), Ethernet clock (ETCLKD), and peripheral module clock (PCLKE).

Address(es): A00B 0020h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	TCLK	—	—	—	SERICK
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ETCKD[1:0]		—	—	—	—	—	—	—	—	PCKE[1:0]		—	—	—	—
Value after reset:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b5, b4	PCKE[1:0]	Peripheral Module Clock E (PCLKE) Select	Select the low-speed peripheral module clock PCLKE supplied to the WDTA. 00: 75 MHz 01: 37.5 MHz 10: 18.75 MHz Settings other than above are prohibited.	R/W
b7 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b13-b9	—	Reserved	Thes bits are read as 0. The write value should be 0.	R/W
b15, b14	ETCKD[1:0]	Ethernet Clock D (ETCLKD) Select	Select the Ethernet clock ETCLKD (MDIO controller) supplied to Ethernet MAC. 00: 12.5 MHz 01: 6.25 MHz 10: 3.125 MHz 11: 1.563 MHz	R/W
b16	SERICK	High-Speed Serial Clock (SERICLK) Select	Selects the high-speed serial clock SERICLK supplied to RSPIa and SCIFA. 0: 150 MHz 1: 120 MHz	R/W
b19 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b20	TCLK	Trace interface clock (TCLK)	Selects the clock supplied to the trace I/F clock TCLK (CoreSight TPIU). 0: Setting prohibited 1: 75 MHz	R/W
b31 to b21	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

7.2.2 System Clock Control Register 2 (SCKCR2)

The SCKCR2 register is used to select PLL0 or PLL1 as the source of clocks supplied to the system clock.

Address(es): A00B 0024h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CKSEL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CKSEL0	System Clock Source Select	Selects the source of the CPU clock (CPUCLK), system clock (ICLK), high-speed peripheral module clock (PCLKA), and low-speed peripheral module clock (PCLKB). Switching to a clock source which is not in operation is prohibited. 0: PLL0 is selected. 1: PLL1 is selected.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

7.2.3 PLL1 Control Register 2 (PLL1CR2)

The PLL1CR2 register is used to control operation of the PLL1 circuit.

Address(es): A00B 0038h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PLL1EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PLL1EN	PLL1 Operation Control	Specifies whether to run or stop PLL1. 0: PLL1 stops 1: PLL1 runs If the PLL1EN bit is set to 1, 100 us of the PLL oscillation settling time must be counted by using loop processing in the CPU or by a timer. For details about the setting procedure of PLL1, see Figure 7.2.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

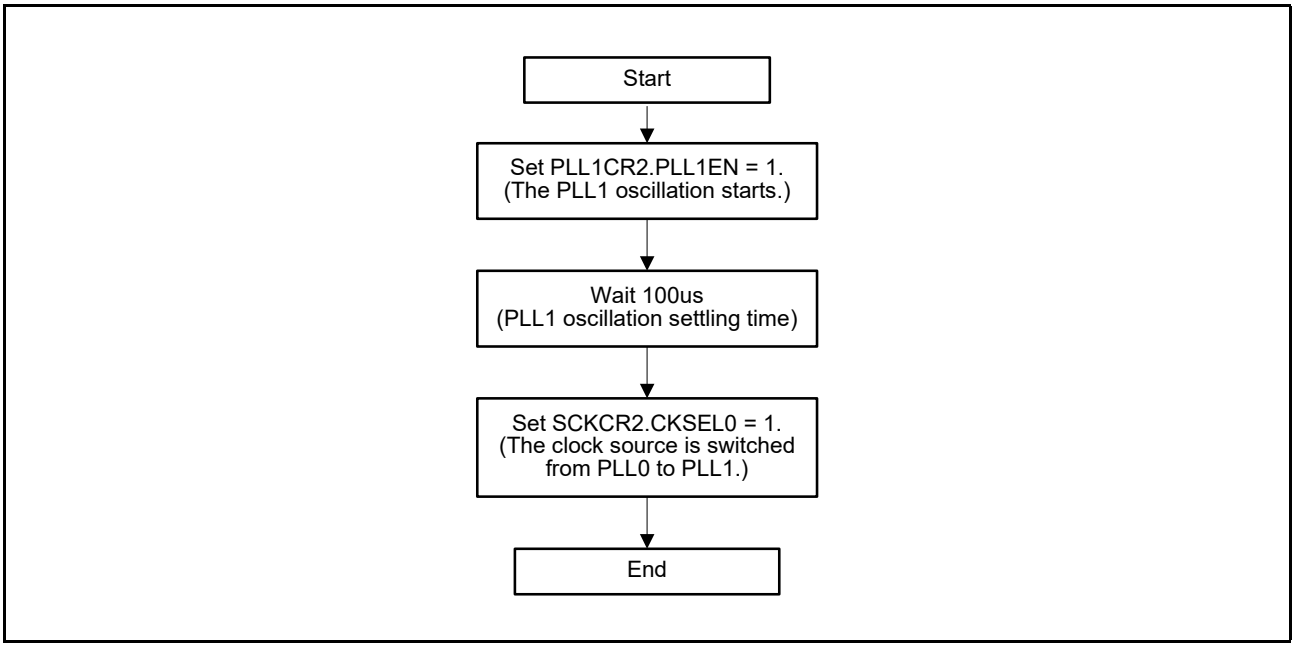


Figure 7.2 Setting Procedure of PLL1

7.2.4 Low-Speed On-Chip Oscillator Control Register (LOCOCR)

LOCOCR is used to control operation of a low-speed on-chip oscillator.

Address(es): A00B 0040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LCSTP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	LCSTP	LOCO Stop	Specify whether to run or stop the low-speed on-chip oscillator (LOCO). 0: Run 1: Stop After setting this bit to run the LOCO, start using the LOCO clock after the LOCO oscillation stabilization time (t_{LOCOWT}) has elapsed.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. To stop the LOCO and then run it again, wait at least the LOCO oscillation stabilization time (tLOCOWT) is elapsed after setting the LCSTP bit to stop the LOCO, and then set the LCSTP bit to run the LOCO again.
To set the bit to stop the LOCO, make sure that the LOCO oscillation is stable.

Note 2. Writing of 1 to the LCSTP bit (stop the LOCO) is prohibited if the oscillation stop detection function is enabled by setting the OSTDCR.OSTDE bit.

7.2.5 Oscillation Stop Detection Control Register (OSTDCR)

The OSTDCR register is used to control the oscillation stop detection function of the main clock.

Address(es): A00B 004Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OSTDE	—	—	—	—	—	—	OSTDIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	OSTDIE	Oscillation Stop Detection Interrupt Enable	Enable or disable the oscillation stop detection interrupt (OSTDI). 0: Oscillation stop detection interrupt is disabled. 1: Oscillation stop detection interrupt is enabled.	R/W
b6 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	OSTDE	Oscillation Stop Detection Function Enable	Enable or disable the oscillation stop detection function. If the oscillation stop detection function is enabled, the LCSTP bit of the low-speed on-chip oscillator control register (LOCOCR) is also set to 0, and the low-speed on-chip oscillator operation starts.*1 0: The oscillation stop detection function is disabled. 1: The oscillation stop detection function is enabled.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The low-speed on-chip oscillator cannot be stopped while the oscillation stop detection function is enabled (OSTDE bit = 1). Writing 1 to the LOCOCR.LCSTP bit is ignored.

7.3 Selecting Input to Main Clock Oscillator

Connect an oscillator to supply the clock signal to the main clock oscillator.

7.3.1 Connecting a Crystal Resonator

Figure 7.3 shows an example of connecting a crystal resonator.
A damping resistor Rd should be added, if necessary. Because the resistor values vary depending on the resonator and the oscillation drive capability, use values recommended by the resonator manufacturer. If use of an external feedback resistor (Rf) is directed by the resonator manufacturer, insert an Rf between EXTAL and XTAL by following the instruction.
When connecting a resonator to supply the clock, the frequency of the resonator should be in the frequency range of the resonator for the main clock oscillator described in Table 7.1.

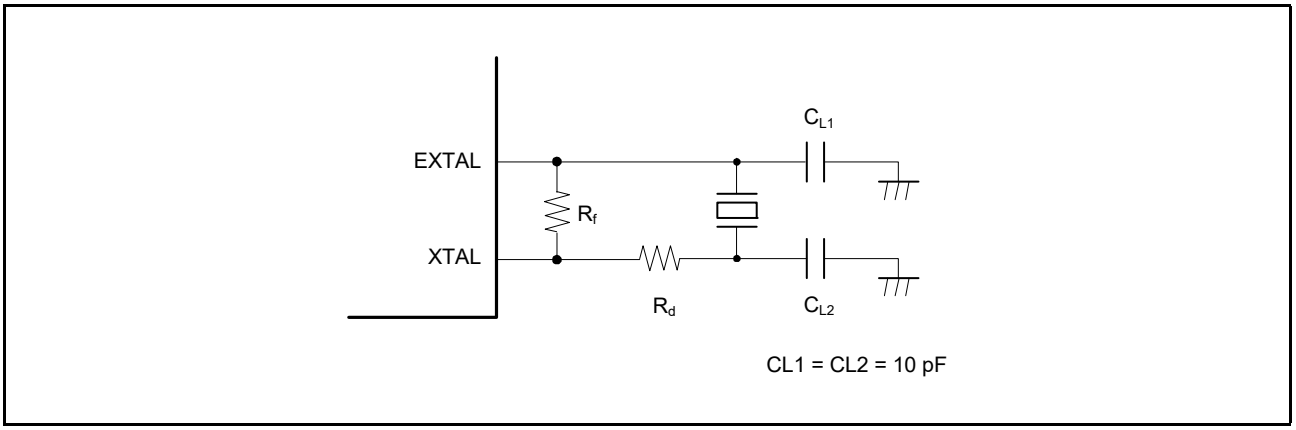


Figure 7.3 Example of Crystal Resonator Connection

Table 7.4 Damping Resistance (Reference Values)

Frequency (MHz)	25
Rd (Ω)	2.2K

Figure 7.4 shows an equivalent circuit of the crystal resonator. Use a crystal resonator that has the characteristics shown in Table 7.5.

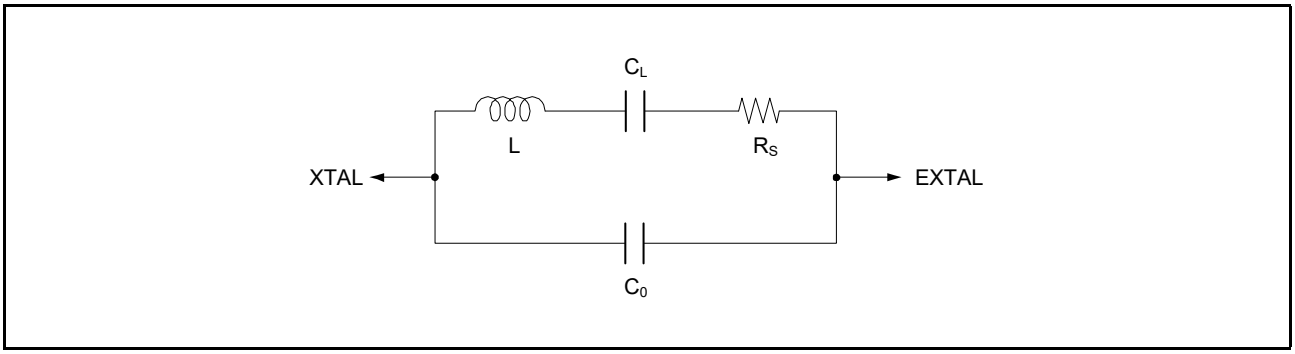


Figure 7.4 Equivalent Circuit of Crystal Resonator

Table 7.5 Crystal Resonator Characteristics (Reference Values)

Frequency (MHz)	25
R_S max (Ω)	100

7.4 Oscillation Stop Detection Function

7.4.1 Oscillation Stop Detection and Operation after Detection

The oscillation stop detection function detects the main clock oscillator stop and to supply LOCO clock pulses from the low-speed on-chip oscillator as the system clock source instead of the main clock, PLL0 clock, and PLL1 clock.
An oscillation stop detection interrupt request can be generated when an oscillation stop is detected.
This LSI detects a main clock oscillation stop when the input clock remains to be 0 or 1 for a certain period, for example, due to a malfunction of the main clock oscillator (see Table 33.26, Oscillation Stop Detection Circuit Characteristics, in section 33.6, Oscillation Stop Detection Timing).
When an oscillation stop is detected, clocks are switched to the LOCO clock.

After a reset is released, the oscillation stop detection function is disabled. To enable the oscillation stop detection function, set the OSTDCR.OSTDE bit to 1.
The clocks that are switched to the LOCO clock by the oscillation stop detection are the PLL0 clock, PLL1 clock, and clock B for RSCAN (CANCLKB).

7.4.2 Oscillation Stop Detection Interrupt

An oscillation-stop detection interrupt (OSTDI) will be generated if the oscillation-stop detection interrupt enable bit (OSTDCR.OSTDIE) is 1 (enabling interrupt generation on oscillation stop detection).
The oscillation stop detection interrupt is connected to the Error Control Module (ECM) as an error interrupt. Interrupts are disabled in the initial state after a reset release. To use oscillation stop detection interrupts, select maskable or non-maskable interrupt as the operation to be performed when the ECN detects an error interrupt. For details, see section 32, Error Control Module (ECM).

7.5 PLL Oscillation Abnormality Detection Function

The PLL oscillation abnormality detection function detects an abnormality in oscillation of the PLL0 or PLL1 by using the clock monitor circuits (CLMA0 and CLMA1) to monitor the frequency, and supplies the main clock instead of the PLL0 clock or PLL1 clock. For details about the CLMA, see section 8, Clock Monitor Circuit (CLMA).

7.6 Low-Speed On-Chip Oscillator Oscillation Abnormality Detection Function

The clock monitor circuit (CLMA2) can be used to detect an abnormality in oscillation of the low-speed on-chip oscillator. For details about the CLMA, see section 8, Clock Monitor Circuit (CLMA).

7.7 PLL Circuit

The PLL circuit has a function to multiply the frequency from the oscillator.

7.8 Internal Clock

Clock sources of internal clock signals are the main clock, LOCO clock, PLL0 clock, PLL1 clock, and the external clock for JTAG. Eleven types of internal clocks listed below are produced from these sources.

(1) Operating clock of the CPU: CPU clock (CPUCLK)
(2) Operating clock of DMAC and interrupt controller: System clock (ICLK)
(3) Operating clocks of peripheral modules: Peripheral module clocks (PCLKA, PCLKB, PCLKD, and PCLKE)
(4) Operating clock for high-speed serial clock: High-speed serial clock (SERICLK)
(5) Operating clock for USB-PHY: USB clock M (USBMCLK)
(6) Operating clock for USB: USB clock P (USBPCLK)
(7) Operating clocks for Ethernet: Ethernet clocks (ETCLKA, ETCLKD, ETCLKE, and ETCLKF)
(8) Operating clock for the CLMA module: CLMA clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1)
(9) Operating clock for the IWDT module: IWDT-dedicated clock (IWDTCLK)
(10) Operating clock for the JTAG module: JTAG clock (JTAGTCK)
(11) Operating clock for the trace interface: Trace interface clock (TCLK)

7.8.1 CPU Clock (CPUCLK)

The CPU clock (CPUCLK) is used as the operating clock of the CPU.
The operating frequency of the CPU is fixed to 150 MHz. It cannot be specified by the user.

7.8.2 System Clock (ICLK)

The system clock (ICLK) is used as the operating clock of DMAC and interrupt controller.
The frequency of ICLK is fixed to 150 MHz. It cannot be specified by the user.

7.8.3 High-Speed Peripheral Module Clock (PCLKA)

The high-speed peripheral module clock (PCLKA) is used as the operating clock for high-speed peripheral modules.
The frequency of PCLKA is fixed to 150 MHz. It cannot be specified by the user.

7.8.4 Low-Speed Peripheral Module Clock (PCLKB)

The low-speed peripheral module clock (PCLKB) is used as the operating clock for low-speed peripheral modules.
The frequency of PCLKB is fixed to 75 MHz. It cannot be specified by the user.

7.8.5 Low-Speed Peripheral Module Clocks (PCLKD and PCLKE)

The unmodulated low-speed peripheral module clocks (PCLKD and PCLKE) are operating clocks for use by low-speed peripheral modules. The frequency of PCLKD is fixed to 75 MHz. It cannot be specified by the user. The frequency of PCLKE is set by the SCKCR.PCKE[1:0] bits.

7.8.6 High-Speed Serial Clock (SERICLK)

The high-speed serial clock (SERICLK) is used as the operating clock for SCIFA and RSPIa.
The SERICLK operating frequency is specified by the SCKCR.SERICK bit.

7.8.7 USB Clock M (USBMCLK)

The USB M (USBMCLK) is used as the operating clock for USB PHY.
The frequency of USBMCLK is fixed to 50 MHz. It cannot be specified by the user.

7.8.8 USB Clock P (USBPCLK)

The clock P (USBPCLK) is used as the operating clock for USB.
The frequency of USBPCLK is fixed to 30 MHz. It cannot be specified by the user.

7.8.9 Ethernet Clocks (ETCLKA, ETCLKD, ETCLKE, and ETCLKF)

Ethernet clocks (ETHCLKA, ETCLKD, ETCLKE, and ETCLKF) are the operating clocks for use with EtherCAT.
The ETCLKD operating frequency is specified by the SCKCR.ETCKD[1:0] bits.
Frequencies of the Ethernet clocks other than ETCLKD is fixed. They cannot be specified by the user.

7.8.10 CLMA Clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1)

CLMA clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1) are used as the operating clock for the CLMA module.
CLMAMCLKA and CLMAMCLKB are obtained by frequency-dividing the main clock.
CMLAMCLK is generated by internal oscillation of the low-speed on-chip oscillator.
CMLAPLCLK0 and CLMAPLCLK1 are obtained by frequency-dividing the clocks which are generated by internal oscillation of PLL0 and PLL1 circuits.

7.8.11 IWDT Clock (IWDTCLK)

The IWDT clock (IWDTCLK) is used as the operating clock for the IWDT module.
IWDTCLK is a 1/2 clock internally generated by the low-speed on-chip oscillator.

7.8.12 ECM Clock (ECMCLK)

The ECM clock (ECMCLK) is used as the delay counter operating clock for the ECM module.
ECMCLK is generated by internal oscillation of the low-speed on-chip oscillator.

7.8.13 JTAG Clock

The JTAG-dedicated clock (JTAGTCK) is the operating clock for the JTAG.
JTAGTCK is generated by the external clock for JTAG (TCK).

7.8.14 Trace Interface Clock (TCLK)

The trace interface clock (TCLK) is used as the operating clock for the trace interface within CoreSight.
This clock is obtained by frequency-dividing the clock which is generated by internal oscillation within the PLL0 or PLL1 circuit.
This clock divided by two is output from the LSI as the trace clock for on-chip debugger (TRACECLK).

7.9 Usage Notes

7.9.1 Notes on Clock Generation Circuit

(1) Note that the operating frequency of the high-speed serial clock (SERICLK) that is supplied to the modules based on the SCKCR register settings varies before and after the frequency is changed.

(2) Do not change the PCLKE clock frequency after the WDTA counting started. To start the WDTA counting after the clock frequency is changed, confirm that the change to the frequency has been completed before you start the WDTA counting.

(3) Do not change the ETCLKD clock frequency while the MDIO controller is operating. To cancel the module-stop state after the clock frequency is changed, confirm that the change to the frequency has been completed before you cancel the module-stop state.

(4) Do not change the SERICLK clock frequency while the RSPIa (channels 0 and 1) or SCIFA (channels 0 to 4) is operating. To cancel the module-stop state after the clock frequency is changed, confirm that the change to the frequency has been completed before you cancel the module-stop state.

(5) In order to secure the processing after the clock frequency is changed, modify the pertinent clock control register to change the frequency, and then perform a dummy read of the register at least three times. Then perform the subsequent processing.

7.9.2 Notes on Resonator

Since various resonator characteristics relate closely to the user's board design, adequate evaluation is required on the user side before use, referencing the resonator connection example shown in this section. The circuit constants for the resonator depend on the resonator to be used and the stray capacitance of the mounting circuit. Therefore, the circuit constants should be determined in full consultation with the resonator manufacturer. The voltage to be applied between the resonator pins must be within the absolute maximum rating.

7.9.3 Notes on Designing the Board

When using a crystal resonator, place the resonator and its load capacitors as close to the XTAL and EXTAL pins as possible. Other signal lines should be routed away from the oscillation circuit as shown in Figure 7.5. This prevents electromagnetic induction from interfering with correct oscillation.

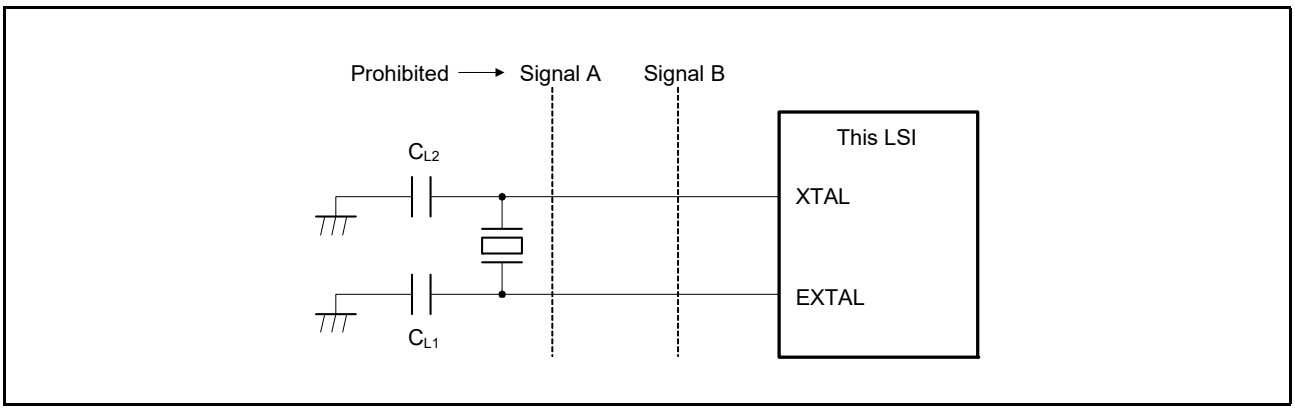


Figure 7.5 Notes on Board Design for Oscillation Circuit

8. Clock Monitor Circuit (CLMA)

A clock monitor circuit (CLMAn) (n = 2 to 0) detects a frequency error in PLL0 output, PLL1 output, or low-speed on-chip oscillator (LOCO) output, and then sends an error signal.

8.1 Overview

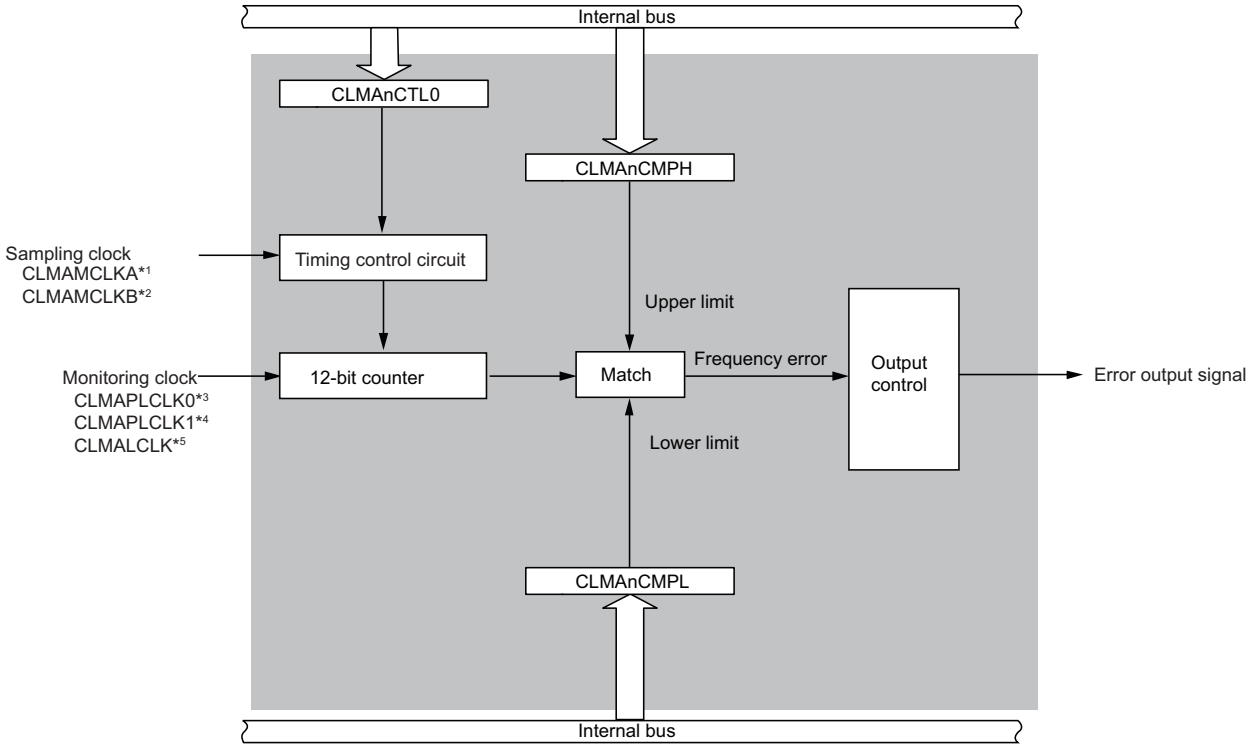
CLMAn (n = 2 to 0) can detect a frequency error in PLL0 output, PLL1 output, or on-chip oscillator (LOCO) output. During 16 cycles of the sampling clock, it counts the rising edges of the monitoring clock (PLL0 output, PLL1 output, and a frequency dividing clock LOCO output), and then compares the counter value with the compare register.

When CLMAn (n = 2 to 0) detects an error, it sends an error signal to the error control module (ECM). Besides, it switches the clock to supply the main clock instead of PLL0 or PLL1 output when it detects an error in PLL0 or PLL1 output.
For details on error signals, see section 32, Error Control Module (ECM).

Table 8.1 Specifications of CLMAn (n = 2 to 0)

Item	Specifications
Monitoring clock	The following monitoring clock frequency errors can be detected: • PLL0 output clock divided by 16 (CLMAPLCLK0, which is to be supplied to CLMA0): 75 MHz • PLL1 output clock divided by 16 (CLMAPLCLK1, which is to be supplied to CLMA1): 75 MHz • Low-speed on-chip oscillator (LOCO) output clock (CLMALCLK, which is to be supplied to CLMA2): 240 kHz
Sampling clock	The following clock frequency errors are monitored as the sampling clock: • Clock which equals to the main clock frequency divided by two (CLMAMCLKA, which is to be supplied to CLMA0): 12.5 MHz • Clock which equals to the main clock frequency divided by two (CLMAMCLKA, which is to be supplied to CLMA1): 12.5 MHz • Clock which equals to the main clock frequency divided by 256 (CLMAMCLKB, which is to be supplied to CLMA2): 97.66 kHz
Error signal output	When CLMAn detects a frequency error, it sends an error signal to the error control module (ECM). • CLMA0 oscillator-stopped detection error signal • CLMA1 oscillator-stopped detection error signal • CLMA2 oscillator-stopped detection error signal
Clock switching function when an error occurs	When an error is detected in PLL0 or PLL1 output, switches the clock to supply the main clock instead of PLL0 or PLL1 output.

Figure 8.1 is a block diagram of CLMAn (n = 2 to 0).



Note 1. Main clock frequency divided by 2 to be supplied to CLMA0 and CLMA1
Note 2. Main clock frequency divided by 256 to be supplied to CLMA2
Note 3. PLL0 output clock divided by 16 to be supplied to CLMA0
Note 4. PLL1 output clock divided by 16 to be supplied to CLMA1
Note 5. Low-speed on-chip oscillator (LOCO) output clock to be supplied to CLMA2

Figure 8.1 Block Diagram of CLMAn (n = 2 to 0)

8.2 Register Descriptions

8.2.1 CLMAn Control Registers 0 (CLMAnCTL0) (n = 2 to 0)

The CLMAnCTL0 registers control operation of clock monitor circuit CLMAn.
Writing to these registers is protected by a specific command sequence. For details, see section 8.3.1, CLMAn Operation, (1) Enabling operations.

Address(es): CLMA0CTL0: A009 0000h
CLMA1CTL0: A009 0020h
CLMA2CTL0: A009 0040h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CLMAn CLME
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CLMAnCLME	Clock Monitor Enable n	Enables or disables operation of clock monitor circuit CLMAn (n = 2 to 0). 0: Disables CLMAn operation. 1: Enables CLMAn operation.	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should be 0.	R/(W)

Note 1. Once the CLMAnCLME bit is set to 1, it cannot be cleared by any operation other than reset.

8.2.2 CLMAn Compare Registers L (CLMAnCMPL) (n = 2 to 0)

The CLMAnCMPL registers set the lower limit for comparing frequency domains.
Values can be written to these registers when the CLMAnCLME bit is set to 0. When the CLMAnCLME bit is set to 1, writing to these registers has no effect.

Address(es): CLMA0CMPL: A009 0008h
CLMA1CMPL: A009 0028h
CLMA2CMPL: A009 0048h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CLMAnCMPL[11:0]											
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Descriptions	R/W
b11 to b0	CLMAnCMPL [11:0]	Clock Monitor Compare L	Specify the lower-limit threshold for the frequency domain.*1 • For details, see section 8.3.2, (2) Method of calculating threshold values, CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]. Recommended value: $f_{CLMATMON\ (min)} / f_{CLMATSMP\ (max)} \times 16 - 1$ $f_{CLMATMON}$: Monitoring clock frequency $f_{CLMATSMP}$: Sampling clock frequency • Minimum value: 0001h	R/W
b15 to b12	—	Reserved	These bits are always read as 0. The write value should be 0.	R/(W)

Note 1. To set the CLMAnCMPL registers, the following conditions must be met:
- 1 ≤ CLMAnCMPL
- CLMAnCMPL + 3 ≤ CLMAnCMPH

8.2.3 CLMAn Compare Registers H (CLMAnCMPH) (n = 2 to 0)

The CLMAnCMPH registers set the upper limit for comparing frequency domains.
Values can be written to the CLMAnCMPH registers when the CLMAnCLME bit is set to 0. When the CLMAnCLME bit is set to 1, writing to these registers has no effect.

Address(es): CLMA0CMPH: A009 000Ch
CLMA1CMPH: A009 002Ch
CLMA2CMPH: A009 004Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CLMAnCMPH[11:0]											
Value after reset:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b11 to b0	CLMAnCMPH[11:0]	Clock Monitor Compare H	Specify the upper-limit threshold for the frequency domain*1. • For details, see section 8.3.2, (2) Method of calculating threshold values, CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]. Recommended value: $f_{CLMATMON\ (max)} / f_{CLMATSMP\ (min)} \times 16 + 1$ $f_{CLMATMON}$: Monitoring clock frequency $f_{CLMATSMP}$: Sampling clock frequency • Minimum value: CLMAnCMPL+0003h	R/W
b15 to b12	—	Reserved	These bits are always read as 0. The write value should be 0.	R/(W)

Note 1. To set the CLMAnCMPH registers, the following conditions must be met:
- 1 ≤ CLMAnCMPL
- CLMAnCMPL + 3 ≤ CLMAnCMPH

8.2.4 CLMAn Command Registers (CLMAnPCMD) (n = 2 to 0)

The CLMAnPCMD registers control writing to the protected registers.
For details, see section 8.3.1, CLMAn Operation, (1) Enabling operations.

Address(es): CLMA0PCMD: A009 0010h
CLMA1PCMD: A009 0030h
CLMA2PCMD: A009 0050h

	b7	b6	b5	b4	b3	b2	b1	b0
	CLMAnPR[7:0]							
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CLMAnPR[7:0]	CLMAn Protect Key Code	Write a specific command sequence.	W

8.2.5 CLMAn Protection Status Registers (CLMAnPS) (n = 2 to 0)

The CLMAnPS registers indicate whether writing to the protected register is performed correctly. If writing is not performed correctly, a protection error occurs, and the CLMAnPS.CLMAnPRERR bit is set to 1.
For details, see section 8.3.1, CLMAn Operation, (1) Enabling operations.

Address(es): CLMA0PS: A009 0014h
CLMA1PS: A009 0034h
CLMA2PS: A009 0054h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	CLMAn PRERR
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CLMAnPRERR	CLMAn Error	0: No protection error occurs. 1: A protection error occurred.	R
b7 to b1	—	Reserved	These bits are always read as 0.	R

8.3 Operation

8.3.1 CLMAn Operation

(1) Enabling operations

Setting the CLMAnCTL0.CLMAnCLME bit to 1 starts monitoring the monitoring clock (PLL0 output divided by 16, PLL1 output divided by 16, and LOCO output) by using clock monitor circuit CLMAn (n = 2 to 0) output. To write 1 in the CLMAnCTL0.CLMAnCLME bit, follow the following command sequence:

1. Write A5h to the CLMAnPCMD register.
2. Writing to CLMAnCTL0 is performed by the following sequence:
   - Write the target setting value (01h).
   - Write the reversed value of the target (FEh).
   - Write the target value (01h) again.
3. Read CLMAnCTL0.
   When the value of CLMAnCTL0 shows 01h, the operation of CLMAn is enabled.
   If it shows another value, check the value of CLMAn protection status register (CLMAnPS).

When CLMAnPS = 01h, the command sequence is not performed correctly. Execute the sequence again from step 1 to perform writing.

(2) Stopping operations

When a monitoring clock stops due to register operation, the corresponding clock monitor circuit (CLMAn) is disabled automatically. After that, when the monitoring clock starts oscillating again and stabilizes, the clock monitor circuit (CLMAn) resumes operation.

8.3.2 Detecting Error Clock Frequency

(1) Detection Method

1. During 16 cycles of the sampling clock, CLMAn counts the rising edge of the monitoring clock (PLL0 output divided by 16, PLL1 output divided by 16, and LOCO output), and compares the counter value with the set threshold (n = 2 to 0).
   - CLMAnCMPL.CLMAnCMPL[11:0] set the lower-limit threshold of the frequency domain.
   - CLMAnCMPH.CLMAnCMPH[11:0] set the upper-limit threshold of the frequency domain.
2. If the monitoring clock stops, or shows a value lower than the expected frequency, the counter value shows a value lower than the setting of CLMAnCMPL.CLMAnCMPL[11:0].
3. If the monitoring clock shows a value higher than the expected frequency, the counter value shows a value higher than the setting of CLMAnCMPH.CLMAnCMPH[11:0].

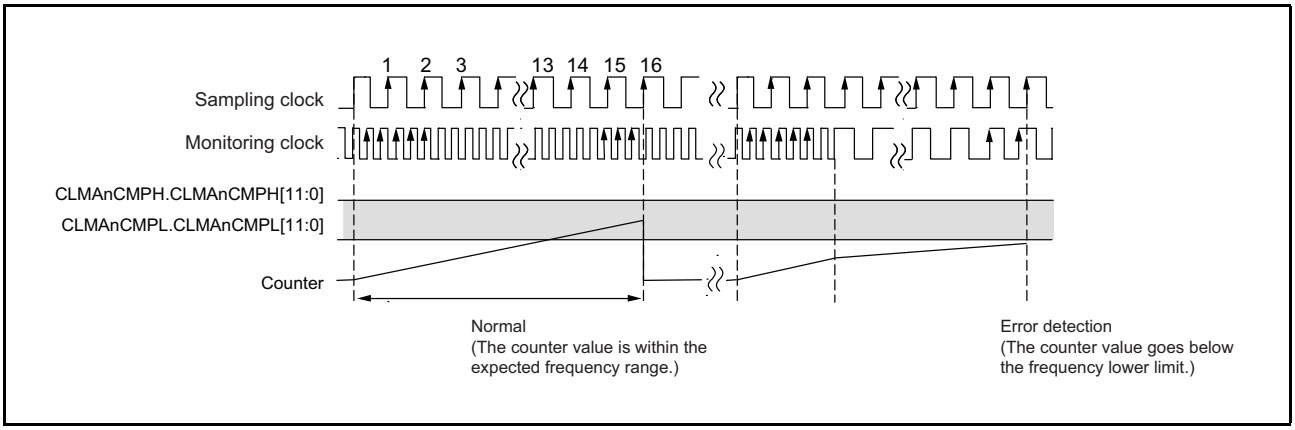


Figure 8.2 Example When the Monitoring Clock Shows a Value Lower Than the Expected Frequency

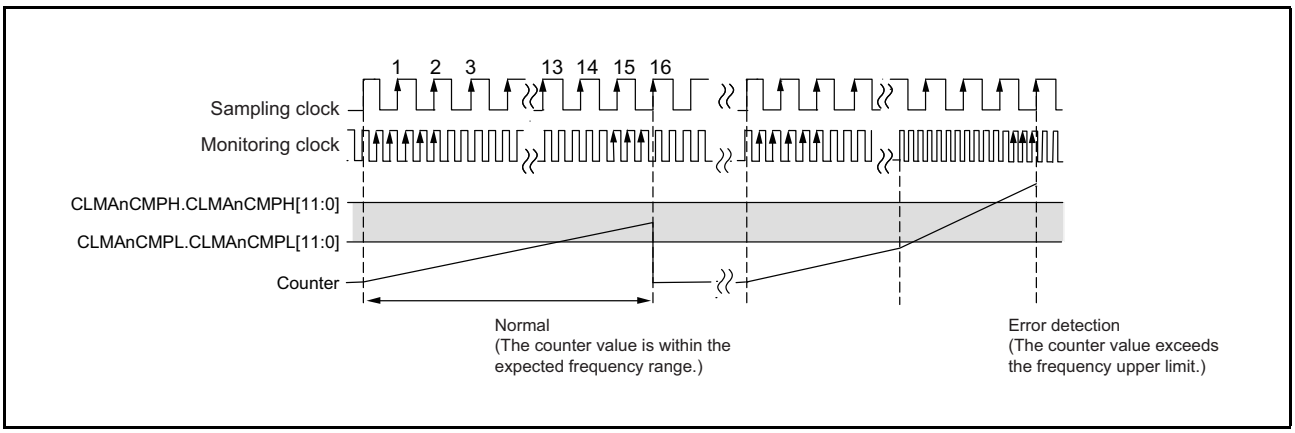


Figure 8.3 Example When the Monitoring Clock Shows a Value Higher Than the Expected Frequency

Note 1. No error is detected if the monitoring clock frequency changes within a sampling interval, and it stays within a valid counter value eventually. A monitoring clock error is detected after a sampling interval (16 cycles of the sampling clock).

(2) Method of calculating threshold values, CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]

For compare registers CLMAnCMPL and CLMAnCMPH, set the minimum and maximum values of the number of cycles (the number of rising edges) for the monitoring clock which is assumed to be normal within 16 cycles of the sampling clock (main clock frequency dividing clock).

$f_{CLMATSMP}$ indicates the sampling clock frequency, $f_{CLMATMON}$ indicates the monitoring clock frequency, and N indicates the number of cycles (rising edges) of the monitoring clock, which is expected within 16 cycles of the sampling clock.

$$\frac{16}{f_{CLMATSMP}} = \frac{N}{f_{CLMATMON}}$$

$$N = \frac{f_{CLMATMON}}{f_{CLMATSMP}} \times 16$$

Considering allowable frequency deviations for the monitoring clock and the sampling clock, use the following formula to calculate the threshold:

Lower-limit threshold

$$= N_{min}$$

$$= \frac{f_{CLMATMON(min)}}{f_{CLMATSMP(max)}} \times 16 - 1$$

Upper-limit threshold

$$= N_{max}$$

$$= \frac{f_{CLMATMON(max)}}{f_{CLMATSMP(min)}} \times 16 + 1$$

Note: Set the threshold within the following range:
CLMAnCMPL ≥ 0001_H
CLMAnCMPH ≥ CLMAnCMPL + 0003_H

Example: For CLMA0

For example, when the sampling clock equals to the main clock frequency divided by two, $f_{CLMATSMP}$ = 12.5 MHz (±5%), or the monitoring clock equals to PLL0 output divided by 16, $f_{CLMATMON}$ = 75 MHz (±5%), the recommended threshold is calculated as follows:

$$N_{min} = f_{CLMATMON}(min) / f_{CLMATSMP}(max) = 71.25 / 13.125 \times 16 - 1$$
$$= 85.86$$

CLMAnCMPL = 86 = 0056h

$$N_{max} = f_{CLMATMON}(max) / f_{CLMATSMP}(min) = 78.75 / 11.875 \times 16 + 1$$
$$= 107.11$$

CLMAnCMPH = 107 = 006Bh

8.3.3 Detecting Error Clock Frequency

When the monitoring clock frequency (PLL0 output divided by 16, PLL1 output divided by 16, and LOCO output) is higher than the upper-limit threshold, or lower than the lower-limit threshold, any of the following signals is sent to the error control module (ECM):

- CLMA0 oscillator-stopped detection error signal
- CLMA1 oscillator-stopped detection error signal
- CLMA2 oscillator-stopped detection error signal

For details on error signals, see section 32, Error Control Module (ECM).

8.4 Notes on Using CLMAn

Do not use a clock for which CLMAn detected an error. If you use the clock, operation of the device is not guaranteed.

9. Low-Power Consumption Function

9.1 Overview

This LSI has several functions for reducing power consumption, including the Cortex-R4 standby function and the module stop function that stops functions independently for each peripheral module. Power consumption by using clock control such as CLKOUT25Mn output control (n = 1, 0) is also possible.
Table 9.1 lists the specifications of low power consumption functions. Table 9.2 describes how to stop each peripheral module and exit the stop state.

Table 9.1 Specifications of Low Power Consumption Function

Item	Specifications
Low power consumption mode	Standby mode (Cortex-R4)
Module-stop function	Functions can be stopped independently for each peripheral module.
CLKOUT25Mn (n = 1, 0) output control function	CLKOUT25Mn (n = 1, 0) clock output or stop (held at the low level) can be selected.

Table 9.2 Stopping Peripheral Modules and Exiting Module-Stop State (1 / 2)

Module	How to Stop Operation and Exit the Stop State	Initial State*1
Cortex-R4	Stop condition: Execution of the Wait For Interrupt (WFI) instruction Condition for release from the stop-state: Interrupt	Operating
Internal bus	Always operating	Operating
Tightly Coupled Memory (ATCM or BTCM)	Operate only during access	Operate only during access
Interrupt controller	Operation is always enabled.	Operating
Error control module (ECM)	Operation is always enabled.	Operating
Compare match timer (CMT)	Set the control register to enter or exit the module-stop state.	Stop
Compare match timer W (CMTW)	Set the control register to enter or exit the module-stop state.	Stop
EtherCAT slave controller (ESC)	Set the control register to exit the module-stop state.*2	Stop
MDIO controller	Set the control register to exit the module-stop state.*2	Stop
Serial peripheral interface (RSPIa)	Set the control register to enter or exit the module-stop state.	Stop
Serial communication interface with FIFO (SCIFA)	Set the control register to enter or exit the module-stop state.	Stop
I^2C bus interface (RIICa)	Set the control register to enter or exit the module-stop state.	Stop
CAN module (RSCAN)	Set the control register to enter or exit the module-stop state.	Stop
Clock monitor circuit (CLMA)	Set the control register to enter or exit the module-stop state.	Stop
CRC operation part (CRC)	Set the control register to enter or exit the module-stop state.	Stop
SPI multi I/O bus controller (SPIBSC)	Set the control register to enter or exit the module-stop state.	Stop*3
Event link controller (ELC)	Set the control register to enter or exit the module-stop state.	Stop
USB	Set the control register to stop the module or exit the module stop state.	Stop
Direct memory access controller (DMAC)	Set the control register to stop the module or exit the module stop state.	Operating
I/O port	Operation is always enabled.	Operating

Table 9.2 Stopping Peripheral Modules and Exiting Module-Stop State (2 / 2)

Module	How to Stop Operation and Exit the Stop State	Initial State*1
CoreSight	Set the control register to stop the module or exit the module stop state.	Operating
Watchdog timer (WDTA)	Operation is always enabled.	Operating
Independent watchdog timer (IWDTa)	Operation is always enabled.	Operating

Note 1. Each module is returned to the initial state by means of the RES# pin reset, error control module (ECM) reset, or software reset.
Note 2. This module cannot be stopped by specifying the control register. For details, see section 9.2.2, Module Stop Control Register B (MSTPCRB).
Note 3. The state of the SPI multi I/O bus controller (SPIBSC) after booting processvaries depending on the operation mode that is selected after release from the reset state. For details, see section 3, Operating Modes.

9.2 Register Descriptions

The registers are applicable to the register write protection function. For writing these registers, clear write protection for the target register by setting b1 of the Protect Register (PRCR). For details, see section 11, Register Write Protection Function.

9.2.1 Module Stop Control Register A (MSTPCRA)

The MSTPCRA register controls the module stop state.
For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0300h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	MSTP CRA4	MSTP CRA3	MSTP CRA2	MSTP CRA1	MSTP CRA0
Value after reset:	0	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	MSTPCRA0	CMTW Unit 1 Module Stop	Target module: CMTW unit 1 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b1	MSTPCRA1	CMTW Unit 0 Module Stop	Target module: CMTW unit 0 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b2	MSTPCRA2	CMT Unit 2 Module Stop	Target module: CMT unit 2 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b3	MSTPCRA3	CMT Unit 1 Module Stop	Target module: CMT unit 1 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b4	MSTPCRA4	CMT Unit 0 Module Stop	Target module: CMT unit 0 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b9 to b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b10	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b11	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b31 to b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

9.2.2 Module Stop Control Register B (MSTPCRB)

The MSTPCRB register controls the module stop state.

For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0304h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	MSTPCRB19	MSTPCRB18	—	MSTPCRB16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	MSTPCRB15	—	MSTPCRB13	MSTPCRB12	—	—	MSTPCRB9	MSTPCRB8	MSTPCRB7	MSTPCRB6	MSTPCRB5	—	—	MSTPCRB2	MSTPCRB1	—
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	MSTPCRB1	RSCAN Module Stop	Target module: RSCAN 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b2	MSTPCRB2	RIICa Channel 1 Module Stop	Target module: RIICa channel 1 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b4, b3	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b5	MSTPCRB5	SCIFA Channel 4 Module Stop	Target module: SCIFA channel 4 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b6	MSTPCRB6	SCIFA Channel 3 Module Stop	Target module: SCIFA channel 3 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b7	MSTPCRB7	SCIFA Channel 2 Module Stop	Target module: SCIFA channel 2 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b8	MSTPCRB8	SCIFA Channel 1 Module Stop	Target module: SCIFA channel 1 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b9	MSTPCRB9	SCIFA Channel 0 Module Stop	Target module: SCIFA channel 0 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b11, b10	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b12	MSTPCRB12	RSPIa Channel 1 Module Stop	Target module: RSPIa channel 1 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b13	MSTPCRB13	RSPIa Channel 0 Module Stop	Target module: RSPIa channel 0 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b14	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b15*1	MSTPCRB15	ESC Module Stop	Target module: ESC 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b16*1	MSTPCRB16	MDIO Controller Module Stop	Target module: MDIO controller 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W

Bit	Symbol	Bit Name	Description	R/W
b17	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b18*1	MSTPCRB18	MDIO Contoller Module Stop	Target module: MDIO controller 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b19*1	MSTPCRB19	CLKOUT25Mn (Ether PHY Clock Output) Stop (n = 1, 0)	0: Clock output from the CLKOUT25Mn pin is permitted 1: Clock output from the CLKOUT25Mn pin is stopped (held at low level)	R/W
b31 to b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. EtherCAT-related functions are stopped in the initial state. To use the functions, specify the MSTPCRB.MSTPCRB15, MSTPCRB16, MSTPCRB18, and MSTPCRB19 bits for releasing the module from the module-stop state. Note, however, that once the modules have been released from the module-stop state, they cannnot be set to the module-stop state again. Operation is not guaranteed after release from the module-stop state is initiated the second time. After the modules are returned to the initial state (stop state) by a reset, they can be released from the module-stop state again. In addition, the MSTPCRB16 and MSTPCRB18 bits need to be set to 0 when the module-stop state of the MDIO controller is released.

9.2.3 Module Stop Control Register C (MSTPCRC)

The MSTPCRC register controls the module stop state.

For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0308h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	MSTP CRC14	MSTP CRC13	MSTP CRC12	MSTP CRC11	—	MSTP CRC9	—	—	MSTP CRC6	—	—	—	—	MSTP CRC1	—
Value after reset:	0	1	1	1	1	1	1*1	1	0	1	1	1	1	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	MSTPCRC1	USB Module Stop	Target module: USB 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b5 to b2	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b6	MSTPCRC6	ELC Module Stop	Target module: ELC 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b8	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b9	MSTPCRC9	SPIBSC Module Stop	Target module: SPIBSC 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b10	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b11	MSTPCRC11	CRC Module Stop	Target module: CRC 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b12	MSTPCRC12	CLMA Unit 2 Module Stop	Target module: CLMA unit 2 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b13	MSTPCRC13	CLMA Unit 1 Module Stop	Target module: CLMA unit 1 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b14	MSTPCRC14	CLMA Unit 0 Module Stop	Target module: CLMA unit 0 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The state of the SPI multi I/O bus controller (SPIBSC) after booting process varies depending on the settings in the loader parameters. The initial value of this bit also varies. For details, see section 3.4.2, Parameters for the Loader.

9.2.4 Module Stop Control Register E (MSTPCRE)

The MSTPCRE register controls the module stop state.
For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0310h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	MSTP CRE5	MSTP CRE4	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b3 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	MSTPCRE4	DMAC Unit 1 Module Stop	Target module: DMAC unit 1 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b5	MSTPCRE5	DMAC Unit 0 Module Stop	Target module: DMAC unit 0 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b31 to b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

9.2.5 Module Stop Control Register F (MSTPCRF)

The MSTPCRF register controls the module stop state.
For release from the module-stop state, see section 9.3.1, Module-Stop Function.

Address(es): A00B 0314h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSTP CRF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MSTPCRF0	CoreSight Module Stop	Target module: CoreSight 0: Release from the module-stop state 1: Transition to the module stop state is made	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

9.3 Operation

9.3.1 Module-Stop Function

The module-stop function can stop operation of each module of on-chip peripheral functions.
When the MSTPmi bit (m = A to C, E, and F, i = 31 to 0) in the MSTPCRA to MSTPCRC, MSTPCRE, and MSTPCRF registers is set to 1, the specified module stops operating and enters the module-stop state.
Clearing the MSTPmi bit (m = A to C, E, and F, i = 31 to 0) to 0 leads to release from the module-stop state.
Follow the procedure given below when releasing any of the peripheral modules listed in Table 9.3 from the module-stop state. This procedure is not required for peripheral modules not listed in Table 9.3. However, release from the module-stop state should be in accord with the procedure for initialization described in the section for the given peripheral module.

[Procedures]

(1) Clear the corresponding bit in the relevant module stop control register (MSTPCRm; m = A to C, E, F) then immediately dummy-read the MSTPCRm register once.
(2) Dummy-read any register of the peripheral module which is being released from the module-stop state. After that, all registers of that peripheral module will be accessible.

Remarks: The MPU having set the access control attribute for the peripheral I/O register region to ‘Strongly-ordered’ or ‘Device’ is a prerequisite for the procedure.

<Example code>

```
volatile unsigned long dummy;                  //  Declared volatile to prevent optimization being applied

  SYSTEM.MSTPCRA.BIT.MSTPCRA0 = 0;             //  Setting to release CMTW unit 1 from the module-stop state
  dummy = SYSTEM.MSTPCRA.BIT.MSTPCRA0;         //  Step 1: Dummy-read the MSTPCRm register.

  dummy = CMTW1.CMWIOR.WORD;                   //  Step 2: Dummy-read any register of CMTW unit 1.
  CMTW1.CMWIOR.WORD = 0x81;                    //  The first setting for CMTW unit 1 (value is an example)
```

For details about the initial state after a release from the reset state, see Table 9.2, Stopping Peripheral Modules and Exiting Module-Stop State.

Note: Directly after a module is set to the module-stop state, writing might still be possible to the control register of that module.

Table 9.3 Peripheral Functions Requiring the above Procedure for Release from the Module-Stop State (1 / 2)

Peripheral Functions	Corresponding Module-Stop Control Register
CMTW unit 1	MSTPCRA register, MSTPCRA0 bit
CMTW unit 0	MSTPCRA register, MSTPCRA1 bit
CMT unit 2	MSTPCRA register, MSTPCRA2 bit
CMT unit 1	MSTPCRA register, MSTPCRA3 bit
CMT unit 0	MSTPCRA register, MSTPCRA4 bit
RSCAN	MSTPCRB register, MSTPCRB1 bit
RIICa channel 1	MSTPCRB register, MSTPCRB2 bit
SCIFA channel 4	MSTPCRB register, MSTPCRB5 bit
SCIFA channel 3	MSTPCRB register, MSTPCRB6 bit
SCIFA channel 2	MSTPCRB register, MSTPCRB7 bit

Table 9.3 Peripheral Functions Requiring the above Procedure for Release from the Module-Stop State (2 / 2)

Peripheral Functions	Corresponding Module-Stop Control Register
SCIFA channel 1	MSTPCRB register, MSTPCRB8 bit
SCIFA channel 0	MSTPCRB register, MSTPCRB9 bit
RSPIa channel 1	MSTPCRB register, MSTPCRB12 bit
RSPIa channel 0	MSTPCRB register, MSTPCRB13 bit
ELC	MSTPCRC register, MSTPCRC6 bit
SPIBSC	MSTPCRC register, MSTPCRC9 bit
CRC	MSTPCRC register, MSTPCRC11 bit
CLMA unit 2	MSTPCRC register, MSTPCRC12 bit
CLMA unit 1	MSTPCRC register, MSTPCRC13 bit
CLMA unit 0	MSTPCRC register, MSTPCRC14 bit
DMAC unit 1	MSTPCRE register, MSTPCRE4 bit
DMAC unit 0	MSTPCRE register, MSTPCRE5 bit

9.3.2 Cortex-R4 Standby Mode

9.3.2.1 Transition to Cortex-R4 Standby Mode

Cortex-R4 enters standby mode by execution of a WFI instruction. In standby mode, Cortex-R4 stops operating, thus reducing power consumption. For details, see the technical reference manual provided by Arm. For details, see the technical reference manual provided by Arm.

9.3.2.2 Release from Cortex-R4 Standby Mode

Release from Cortex-R4 standby mode is initiated by any interrupt, the RES# pin reset, an ECM reset, or a software reset.

- Release triggered by an interrupt signal
  Generation of an interrupt in a CPU triggers release from standby mode or sleep mode of the CPU and the interrupt exception handling starts. Release is triggered by a non-maskable interrupt or a maskable interrupt that meets the following conditions:

  (1) The interrupt request is permitted by using the interrupt enable register.
  (2) Nothing has been assigned to DMAC by using the DMAC source selection register.

- Release by a reset
  After the RES# pin reset, ECM reset, or software reset is cleared, Cortex-R4 starts the reset exception handling. For details about resets, see section 6, Reset.

9.4 Usage Notes

9.4.1 I/O Port State

To reduce I/O power consumption, pin processing based on I/O control is required. For details, see section 16, I/O Ports.

9.4.2 Module-Stop State of DMAC

Do not set a module-stop state while DMAC is operating. Make sure that the DMAC is inactive before you attempt to set the module-stop state.
For details, see section 14, DMA Controller (DMACAa).

9.4.3 On-Chip Peripheral Module Interrupts in Module Stop State

Peripheral modules cannot interrupt in a module-stop state. Therefore, if the module-stop state is set during interrupt processing of the module or during DMA transfer by the DMAC, a CPU interrupt source or a DMAC or DTC startup source cannot be cleared. For this reason, disable the relevant interrupts before setting the module-stop state.

9.4.4 USB Low Power Consumption

For details about how to clear the USB module-stop state and how to enter the USB in low-power consumption mode, see section 23, USB2.0HS Host Module (USBh), and section 24, USB 2.0 HS Function Module (USBf).

9.4.5 Low Power Consumption for EtherCAT-Related Functions

EtherCAT-related functions are stopped in the initial state. To use the functions, specify the MSTPCRB.MSTPCRB15, MSTPCRB16, MSTPCRB18, and MSTPCRB19 bits for release from the module-stop state. Note, however, that after release from the module-stop state, the module stop state cannot be set again. Operation is not guaranteed after release from the module-stop state is initiated the second time. After the modules are returned to the initial state (stop state) by a reset, they can be released from the module-stop state again.

9.4.6 Write Protection Function

The module stop control registers (MSTPCRA to MSTPCRC, MSTPCRE, and MSTPCRF) can be write-protected. To write to these registers, specify bit 1 of the Protect Register (PRCR) to unlock the write protection. For details, see section 11, Register Write Protection Function.

10. Debugging Interface

This LSI has an internal debugging interface which adopts the CoreSight architecture. This LSI supports debugging functions, such as downloading, running, and breaking a program, and trace function, which outputs execution history of programs.

10.1 Overview

This LSI supports JTAG and SWD interfaces as interfaces for debugging, and trace port and SWV interfaces as interfaces for trace.
This LSI has TAP controllers for boundary scan and for CoreSight debugging, which can be selected by the input level of the BSCANP pin. To use debugging function, set the input level of the BSCANP pin to Low.
For details on boundary scan, see section 31, Boundary Scan.
Table 10.1 lists the specifications of CoreSight, and Figure 10.1 is a block diagram of CoreSight. Moreover, Table 10.4 and Table 10.5 indicate the CoreSight address map. For details on CoreSight, see Arm's technical reference manual.

Table 10.1 CoreSight Specifications

Item	Specifications
Debugging function	• JTAG interface • SWD (Serial Wire Debug) interface
Trace function	• Trace port interface 8 bits × 75 Mbps (37.5 MHz, DDR) trace data pin output Embedded Trace Buffer (ETB) 4 KB • SWV (Serial Wire Viewer) interface

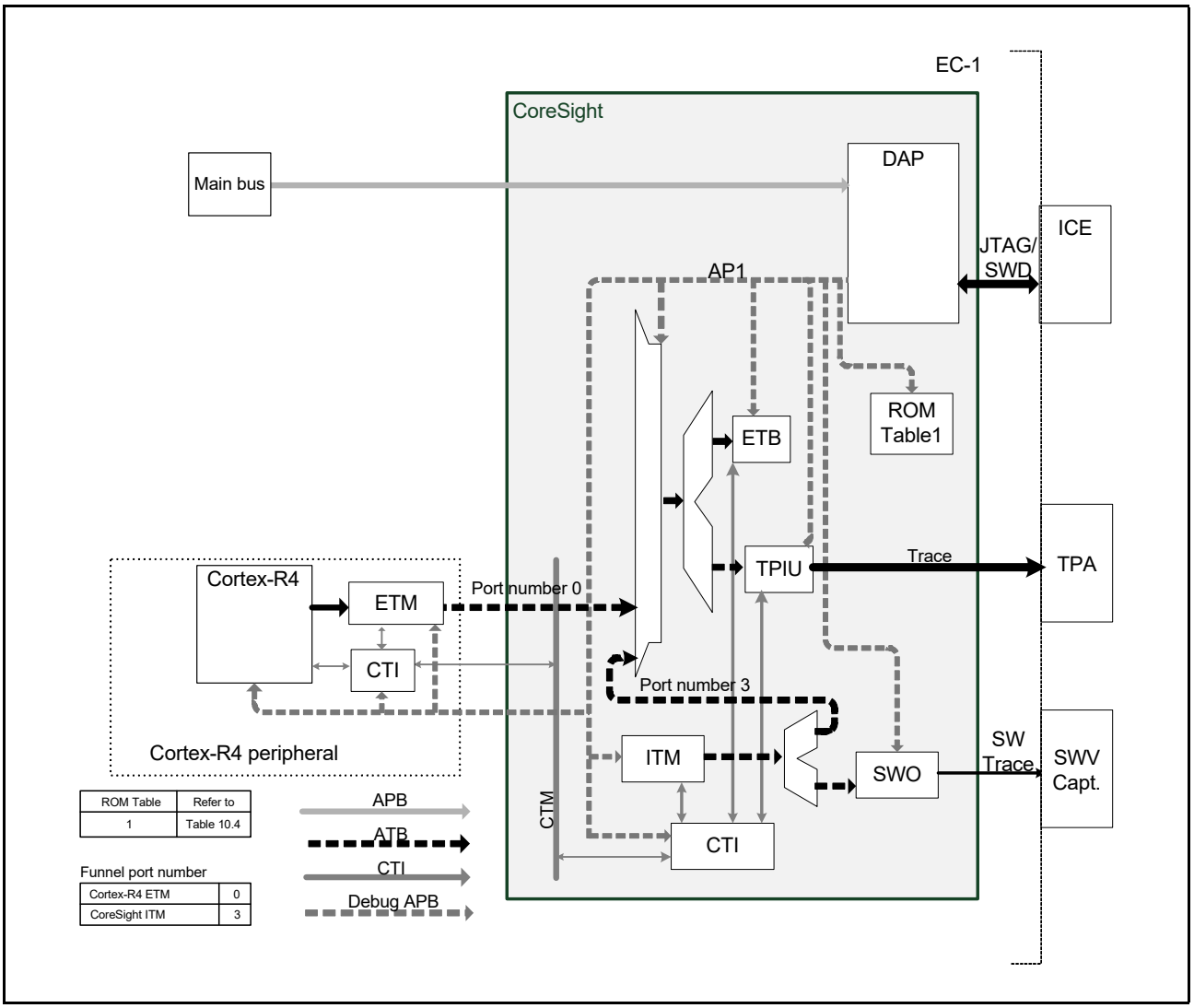


Figure 10.1 Block Diagram of CoreSight

Table 10.2 CTI Trigger Input and Output (CoreSight)

CTI Input Pin	Source Device	Signal	CTI Output Pin	Destination Device	Signal
CTITRIGIN[0]	—	—	CTITRIGOUT[0]	ETB	FLUSHIN
CTITRIGIN[1]	—	—	CTITRIGOUT[1]	ETB	TRIGIN
CTITRIGIN[2]	ETB	FULL	CTITRIGOUT[2]	TPIU	FLUSHIN
CTITRIGIN[3]	ETB	ACQCOMP	CTITRIGOUT[3]	TPIU	TRIGIN
CTITRIGIN[4]	ITM	TRIGOUT	CTITRIGOUT[4]	—	—
CTITRIGIN[5]	—	—	CTITRIGOUT[5]	—	—
CTITRIGIN[6]	—	—	CTITRIGOUT[6]	—	—
CTITRIGIN[7]	—	—	CTITRIGOUT[7]	—	—

Table 10.3 CTI Trigger Input and Output (Cortex-R4)

CTI Input Pin	Source Device	Signal	CTI Output Pin	Destination Device	Signal
CTITRIGIN[0]	Cortex-R4	DBGTRIGGER	CTITRIGOUT[0]	Cortex-R4	EDBGRQ
CTITRIGIN[1]	Cortex-R4	nPMUIRQ	CTITRIGOUT[1]	ETM-R4	EXTIN[0]
CTITRIGIN[2]	ETM-R4	EXOUT[0]	CTITRIGOUT[2]	ETM-R4	EXTIN[1]
CTITRIGIN[3]	ETM-R4	EXOUT[1]	CTITRIGOUT[3]	VIC	TRIGINT
CTITRIGIN[4]	Cortex-R4	COMMRX	CTITRIGOUT[4]	—	—
CTITRIGIN[5]	Cortex-R4	COMMTX	CTITRIGOUT[5]	—	—
CTITRIGIN[6]	ETM-R4	TRIGGER	CTITRIGOUT[6]	—	—
CTITRIGIN[7]	—	—	CTITRIGOUT[7]	Cortex-R4	DBGRESTART

Table 10.4 CoreSight Address Map (Debug-APB)

Cortex-R4 CPU View	Debugger View *1 (AP = 1)	Module
H'E8000000 to H'E8000FFF	H'00000000 to H'00000FFF	CoreSight/DAP ROM
H'E8001000 to H'E8001FFF	H'00001000 to H'00001FFF	CoreSight/ETB
H'E8002000 to H'E8002FFF	H'00002000 to H'00002FFF	CoreSight/CTI
H'E8003000 to H'E8003FFF	H'00003000 to H'00003FFF	CoreSight/TPIU
H'E8004000 to H'E8004FFF	H'00004000 to H'00004FFF	CoreSight/Funnel
H'E8005000 to H'E8005FFF	H'00005000 to H'00005FFF	CoreSight/ITM
H'E8006000 to H'E8006FFF	H'00006000 to H'00006FFF	CoreSight/SWO
H'E8007000 to H'E8007FFF	H'00007000 to H'00007FFF	—
H'E8008000 to H'E8008FFF	H'00008000 to H'00008FFF	Cortex-R4/CPU
H'E8009000 to H'E8009FFF	H'00009000 to H'00009FFF	Cortex-R4/CTI
H'E800A000 to H'E800AFFF	H'0000A000 to H'0000AFFF	Cortex-R4/ETM-R4
H'E800B000 to H'E800BFFF	H'0000B000 to H'0000BFFF	—
H'E800C000 to H'E800CFFF	H'0000C000 to H'0000CFFF	—
H'E800D000 to H'E800DFFF	H'0000D000 to H'0000DFFF	—
H'E800E000 to H'E800EFFF	H'0000E000 to H'0000EFFF	—
H'E800F000 to H'E800FFFF	H'0000F000 to H'0000FFFF	—

Note 1. When A31 (the most significant bit of the address) is set to 1, access without releasing the access lock becomes possible.

Table 10.5 CoreSight Address Map (1 / 2)

Cortex-R4 CPU View	Module
E800 0000h to E800 0FFFh	CoreSight / DAP ROM
E800 1000h to E800 1FFFh	CoreSight / ETB
E800 2000h to E800 2FFFh	CoreSight / CTI
E800 3000h to E800 3FFFh	CoreSight / TPIU
E800 4000h to E800 4FFFh	CoreSight / Funnel
E800 5000h to E800 5FFFh	CoreSight / ITM
E800 6000h to E800 6FFFh	CoreSight / SWO
E800 7000h to E800 7FFFh	—
E800 8000h to E800 8FFFh	Cortex-R4 / CPU
E800 9000h to E800 9FFFh	Cortex-R4 / CTI
E800 A000h to E800 AFFFh	Cortex-R4 / ETM-R4

Table 10.5 CoreSight Address Map (2 / 2)

Cortex-R4 CPU View	Module
E800 B000h to E800 BFFFh	—
E800 C000h to E800 CFFFh	—
E800 D000h to E800 DFFFh	—
E800 E000h to E800 EFFFh	—
E800 F000h to E800 FFFFh	—

Table 10.6 lists the input/output pins of the debugging interface.

Table 10.6 Configuration of Pins for the Debugging Interface

Name	Pin Name	I/O	Functions
Test Clock	TCK	Input	Data is serially supplied from the Test Data Input (TDI) pin to this module, synchronized with this clock, and output from the Test Data Output (TDO) pin. In SWD mode, this pin works as the SWDCLK pin.
Test Mode Select	TMS	Input/Output	Changing the level of this signal, by synchronizing with TCK, will determine the status of the TAP (Test Access Port) control circuit. The protocol conforms to the JTAG standard (IEEE Std.1149.1). In SWD mode, this pin works as the SWDIO pin.
Test Reset	TRST#*1	Input	This pin accepts input asynchronously with TCK. When the level of this pin is Low, TAP (Test Access Port) is reset. When the levels of the TRST# pin and RES# pin are both Low, TAP and the debugging circuit are reset.
Test Data Input	TDI	Input	Changing the level of this pin, by synchronizing with TCK, will send data to this module. This pin can also be used as a general-purpose port. The initial function is TDI.
Test Data Output	TDO	Output	Reading the level of this pin, by synchronizing with TCK, will read data from this module. This pin can also be selected as the output pin of SWV. This pin can also be used as a general-purpose port. The initial function is TDO.
Trace Clock Output	TRACECLK	Output	This pin is an output pin of the clock used for synchronizing trace data.
Trace Enable Output	TRACECTL	Output	This pin is an output pin of the enable signal for trace control. This pin can also be selected as the output pin of SWV.
Trace Data Output	TRACEDATA7 to TRACEDATA0	Output	This pin is an output pin of trace data. TRACEDATA0 can be selected as an output pin of SWV.
Boundary Scan Setting	BSCANP	Input	Input the high level when boundary scan test is performed. Input Low when debugging is performed by CoreSight. For details on boundary scan, see section 31, Boundary Scan.

Note 1. When you design a board on which an emulator can be used, set the TRST# pin to Low while the RES# pin is asserted at power up, and configure the board so that control is available by the TRST# pin only. When this pin is not used, fix it to Low, or connect it so that the same signal as that on the RES# pin is to be input. For details, see section 10.3.5, Reset Configuration and the Method of Connecting with the Emulator.

10.2 Register Descriptions

10.2.1 Debugging Interface Control Register (DBGIFCNT)

The DBGIFCNT register controls the pins used by the debugging interface.

Address(es): A00B 0A00h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SWVSEL[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	SWVSEL[1:0]	SWV Output Select	Selects the pin for SWV (Serial Wire Viewer) output. b1 b0 0 0: SWV output is not output. 0 1: SWV output is output from the TDO pin. 1 0: SWV output is output from the TRACEDATA0 pin. 1 1: SWV output is output from the TRACECTL pin.	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/(W)

10.3 Operation

10.3.1 JTAG Interface

The JTAG interface uses five signals (TCK, TMS, TDO, TDI, and TRST#) to communicate with the host machine (PC) via the emulator. Figure 10.2 shows an example connection, which includes the RES# pin connection.

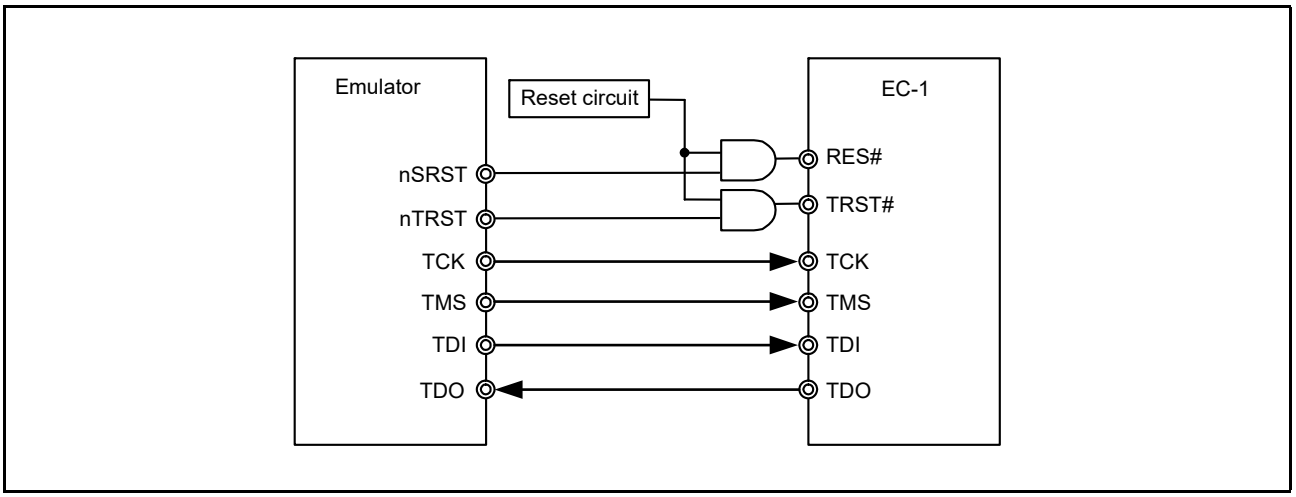


Figure 10.2 Example Connection of the JTAG Interface

10.3.2 SWD Interface

The SWD (Serial Wire Debug) interface uses two signals (SWCLK (TCK) and SWDIO (TMS)) to communicate with the host machine (PC) via the emulator. Figure 10.3 shows an example connection, which includes the RES# pin connection.

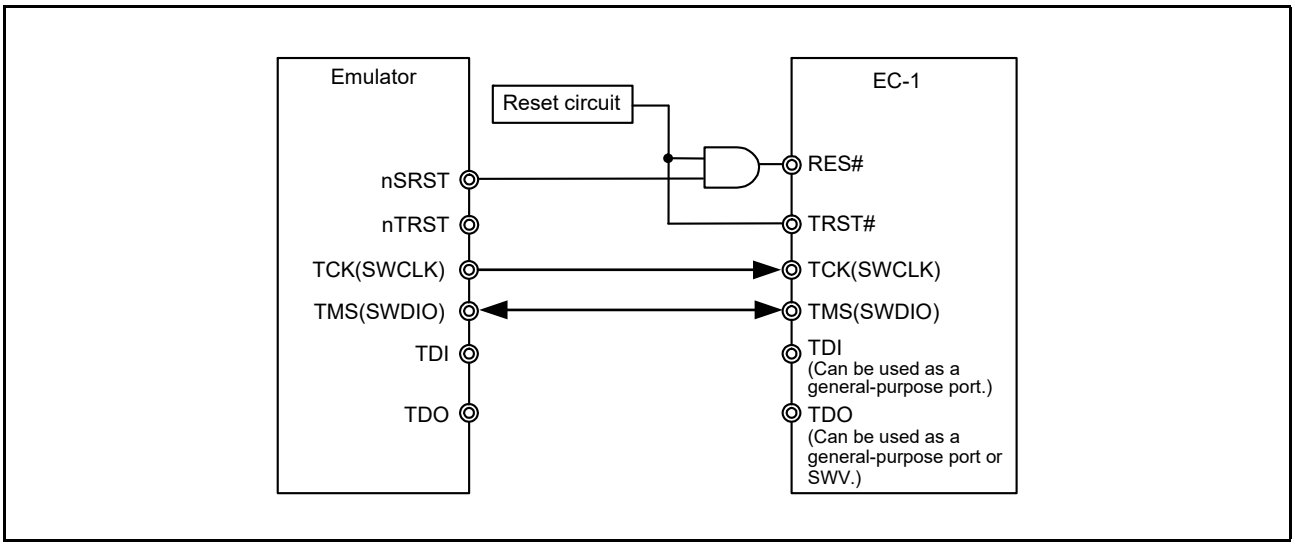


Figure 10.3 Example Connection of the SWD Interface

When the SWD interface is used for debugging, the TDI and TDO pins can be used as general-purpose ports. When you use these pins as general-purpose ports, perform pin settings by referring to section 17, Multi-Function Pin Controller (MPC).

Note: In the initial status of this LSI, the debugging interface is in JTAG mode. If you use the TDI and TDO pins as general-purpose ports and the emulator connection for debugging, switch the mode to SWD (Serial Wire Debug) mode by the control from the debugger, and then start debugging.

10.3.3 Trace Port Interface

The trace port interface uses ten signals (TRACECLK, TRACECTL, and TRACEDATA7 to TRACEDATA0) to output trace information. Information about branch instructions of the executed program (obtained by the ETM (Embedded Trace Macrocell) trace) is output from the trace port interface. After the debugger complements the information, you can know the branch source and destination at the time a branch occurred. For details on trace information, see the manual of each emulator vendor.

Only DDR clocking mode is supported for the synchronization relationship between the TRACECLK pin and TRACEDATA pin.

The maximum number of available TRACEDATA pins is 8. If the number of TRACEDATA pins is smaller than 8, the pins with smaller numbers are used (from TRACEDATA0). Set whether to connect the TRACECTL pin to TPA (Trace Port Analyzer), according to the specifications of the trace data transfer format of the TPA.

As the output frequency of the TRACECLK pin, 37.5 MHz (obtained by dividing the trace I/F clock (TCLK) by 2) can be set. For details, see section 7, Clock Generation Circuit.

In the initial status, different functions are assigned to the TRACECLK, TRACEDATA0 to TRACEDATA7, and TRACECTL pins. Perform pin settings by referring to section 17, Multi-Function Pin Controller (MPC).

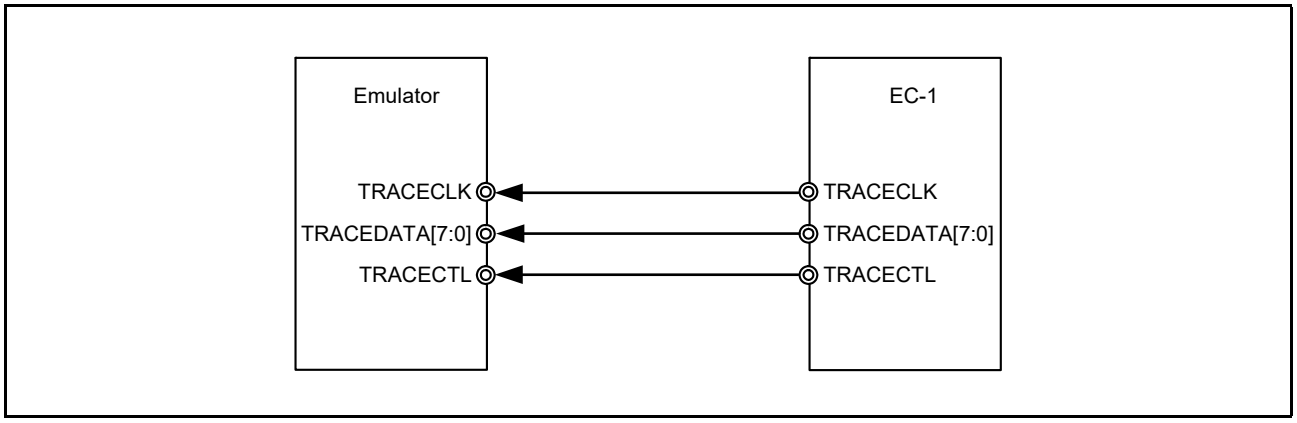


Figure 10.4 Example Connection of the Trace Port Interface

10.3.4 SWV Interface

The SWV (Serial Wire Viewer) interface is used to output trace information from the pin (TDO (SWV), TRACEDATA0 (SWV), or TRACECTL (SWV)) set by the DBGIFCNT register. When the JTAG interface is used, TDO (SWV) cannot be used. The SWV trace is a function that samples specified data at the specified sampling-cycle interval. For details on trace information, see the manual of each emulator vendor.

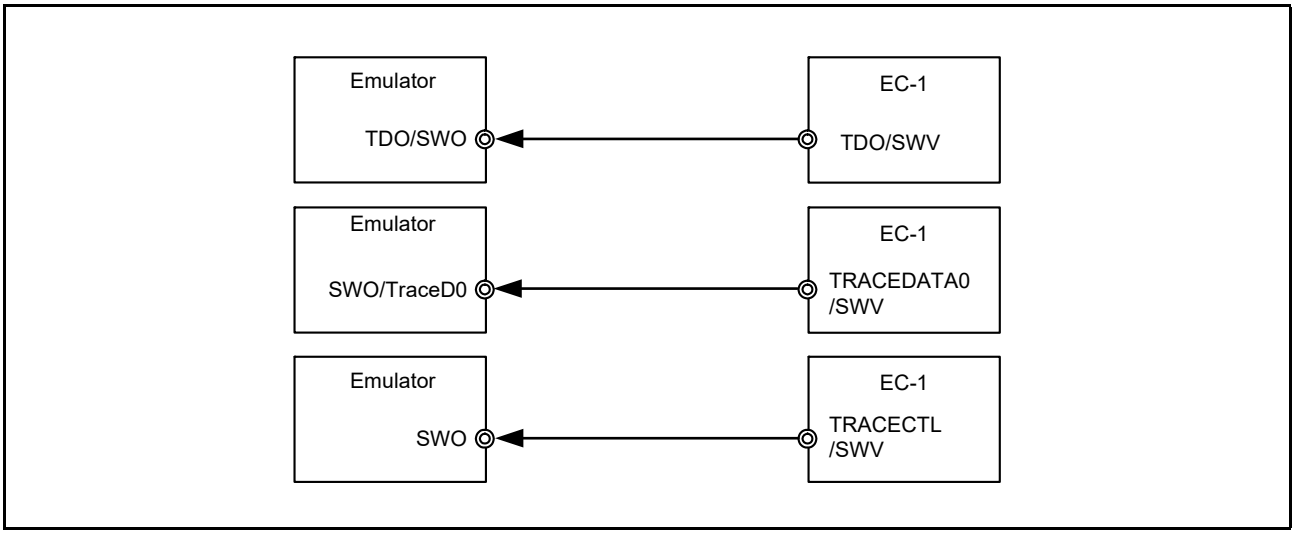


Figure 10.5 Example Connection of the SWV Interface

10.3.5 Reset Configuration and the Method of Connecting with the Emulator

When you design a board on which the emulator can be used, set the TRST# pin to Low while the RES# pin is asserted at power up. Also, configure the board so that control is available by the TRST# pin only.
When debugging is performed, if the initial level of the RES# and TRST# pins are both Low, the CPU and debugging section become the reset status. Then, setting the TRST# pin to High while the RES# pin is kept to Low will enable the debugging setting before CPU startup.
When the emulator is not connected, fix the TRST# pin to Low, or let the signal same as that on the RES# pin input to the TRST# pin.

Note: When debugging is performed by CoreSight, input Low to the BSCANP pin to disable the boundary scan function.

10.3.5.1 Example Connection of the Emulator That Cannot Drive the nTRST Output to High

Figure 10.6 shows an example of connection circuit when an emulator that cannot drive the nTRST output to High is used. The TRST# pin is pulled up, and is asserted to Low by the emulator. To perform debugging settings before CPU startup, follow the timing chart for when the emulator is connected (see Figure 10.6).

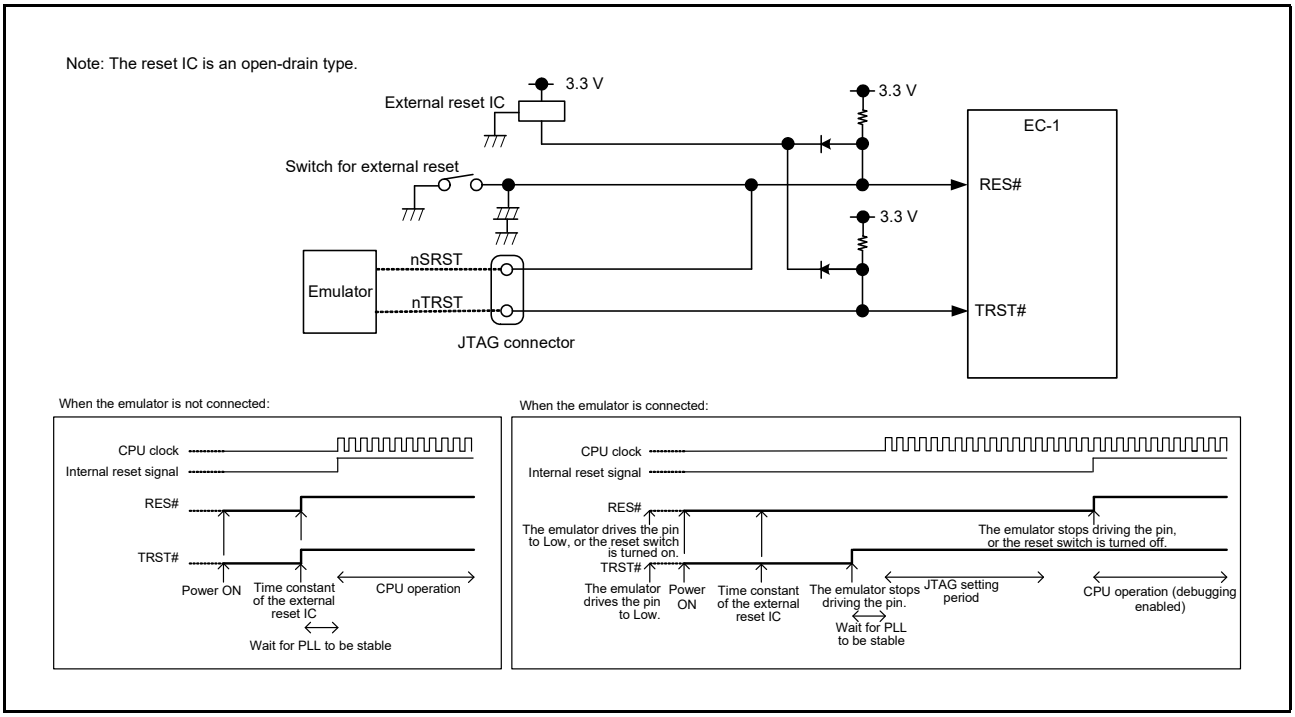


Figure 10.6 Example of Connection Circuit of an Emulator That Cannot Drive the nTRST Output to High

10.3.5.2 Example Connection of the Emulator That Can Drive the nTRST Output to High

Figure 10.7 shows an example of connection circuit when an emulator that can drive the nTRST output to High is used. The TRST# pin (High or Low) is controlled by the emulator. To perform debugging settings before CPU startup, follow the timing chart when the emulator is connected (see Figure 10.7).

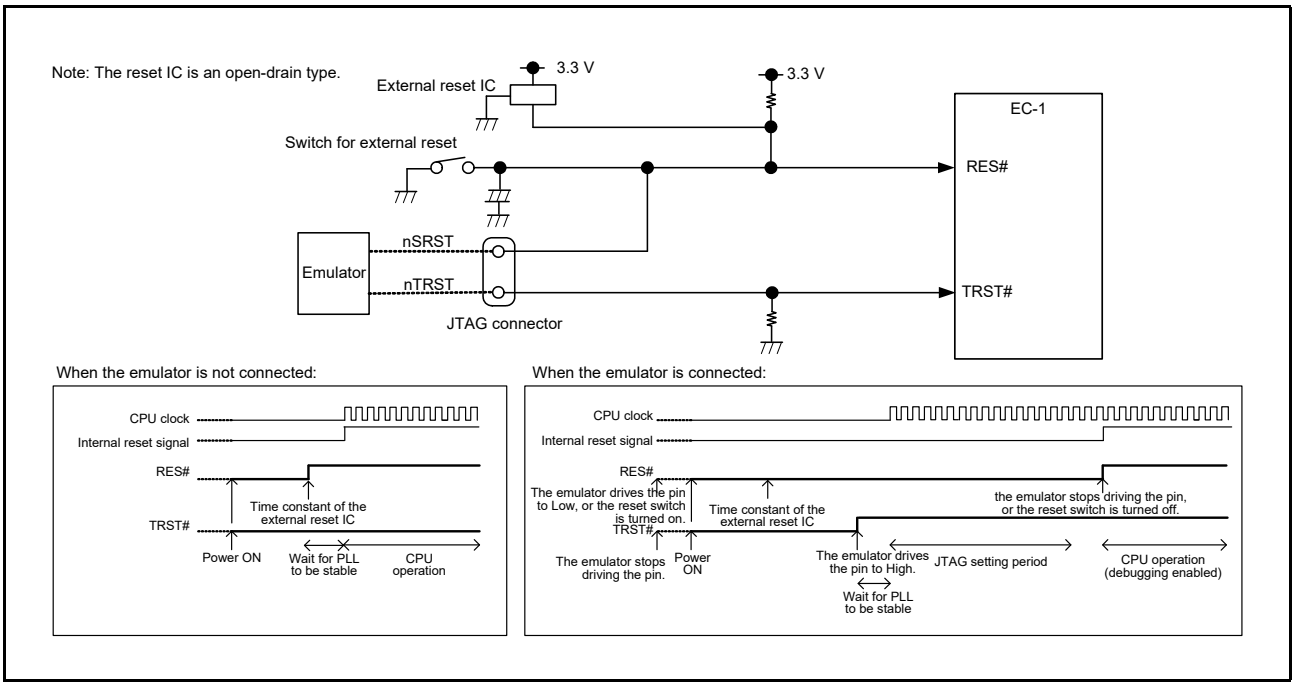


Figure 10.7 Example of Connection Circuit of an Emulator That Can Drive the nTRST Output to High

10.3.6 Handling of JTAG Pins When No Emulator Is Connected

If no emulator is connected, pin handling is required according to Table 10.7.

Table 10.7 Handling of JTAG Pins When No Emulator is Connected

Pin Name	Handling
TCK	Pull down the pin.
TMS	Pull up the pin.
TDI	Pull up the pin (except when the port is used as a general-purpose port).
TDO	Open the pin (except when the port is used as a general-purpose port).
TRST#	Pull down the pin, or let the signal same as that on the RES# pin input.

10.3.7 Noise Reduction of the TRST# Pin

Analog delay noise reduction is performed for the TRST# pin. This measure against noise can remove noise that is within 100 ns at minimum.

10.3.8 Available Trace Functions

Table 10.8 lists the trace functions that are available via the respective debugging ports (trace port interface, SWV, and SWD or JTAG).

Table 10.8 Available Trace Functions

CPU Core	Debugging Port	Trace Functions
Cortex-R4 (CR4)	Trace Port interface	Full instruction tracing through the ETM of the Cortex-R4 Software tracing through the ITM among the CoreSight macrocells
	SWV	Only software tracing through the ITM among the CoreSight macrocells
	SWD/JTAG	The same information as for the trace port interface can be acquired via the ETB.

When using software tracing through the ITM among the CoreSight macrocells, access the ITM by software from the CPU. For the address range of the ITM within the CoreSight registers, see Table 10.5.

10.4 Usage Note

10.4.1 Access to the Main Bus

Access to the main bus by the DAP is via the Cortex-R4.

11. Register Write Protection Function

11.1 Overview

The register write protection function protects important registers from being overwritten in case a program runs out of control. The registers to be protected are set with the protect register (PRCR).

Table 11.1 shows the correspondence between the PRCR register bits and the registers to be protected.

Table 11.1 Correspondence between PRCR Register Bits and Registers to be Protected

PRCR Register	Registers to be Protected
PRC0 bit	• Registers related to the clock generation circuit: SCKCR, SCKCR2, PLL1CR2, LOCOCR, OSTDCR
PRC1 bit	• Registers related to the low power consumption functions: MSTPCRA, MSTPCRB, MSTPCRC, MSTPCRE, MSTPCRF • Reset-related registers: RSTSR0, SWRR1, MRCTLC
PRC3 bit	• ATCM wait control register SYTATCMWAIT

11.2 Register Descriptions

11.2.1 Protect Register (PRCR)

The PRCR register controls writing to the protected registers.

Address(es): A00B 0B00h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRKEY[7:0]								—	—	—	—	PRC3	—	PRC1	PRC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PRC0	Protect 0	Enables writing to the registers related to the clock generation circuit. 0: Write disabled 1: Write enabled	R/W
b1	PRC1	Protect 1	Enables writing to the registers related to low-power consumption functions and reset. 0: Write disabled 1: Write enabled	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	PRC3	Protect 3	Enables writing to the ATCM wait control register. 0: Write disabled 1: Write enabled	R/W
b7 to 4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b8	PRKEY[7:0]	PRC Key Code	These bits control permission and prohibition of writing to the PRCR register. To modify the PRCR register, write A5h to PRKEY[7:0]. When a value other than A5h is written to these bits, writing to the PRCR register has no effect.	R/(W)*1
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The value written to these bits is not retained. These bits are always read as 00h.

PRCi Bits (Protect i) (i = 0, 1, 3)

These bits enable or disable writing to the corresponding registers to be protected.

Setting the PRCi bits to 1 and 0 enables and disables writing to the corresponding registers to be protected, respectively.

12. Interrupt Controller (ICUA)

12.1 Overview

As an interrupt controller, the vector interrupt controller (VIC) for the Cortex-R4 is provided. The interrupt controller accepts interrupt requests from peripheral modules and external pins including those from the Ethernet PHY. An interrupt accepted by the interrupt controller is set either as an interrupt for conveying to the CPU (the Cortex-R4) or as an activating trigger signal for the DMACAa.

Table 12.1 lists interrupt specifications, and Figure 12.1 is a block diagram of the interrupt controller.

Table 12.1 Specifications of Interrupt Controller

Item		Description
Interrupts	Interrupt contact destinations	• Cortex-R4 • Two DMACAa units (unit 0: 16ch., unit 1: 16ch.)
	Peripheral function interrupts	Interrupts from peripheral modules*1 Interrupt detection: Edge detection/level detection
	External pin interrupts	Interrupts from pins Ethernet PHY0 and 1, IRQ0 to 4, 6, 7, 9, 11 to 14 Number of sources: 14 Interrupt detection: Low level/falling edge/rising edge/rising and falling edges. One of these detection methods can be set for each source. Digital noise filter function: Supported
	Interrupt priority level	For interrupts to the CPU, the priority level is set in 16 levels by the register.*2
	DMACAa control	According to the interrupt source, DMACAa can be activated. Switches interrupts from peripheral modules to DMA transfer completion interrupts.*3
Non-maskable interrupts	NMI pin interrupts	Interrupts from the NMI pin Interrupt detection: Falling edge/rising edge Digital noise filter function: Supported
	For Cortex-R4	The following sources can be allocated as the non-maskable high-speed interrupt (FIQ) source. • Non-maskable interrupts from ECM (Error Control Module) • Non-maskable interrupts from the NMI pin
Restoration from the standby mode		Restoration due to non-maskable interrupt and all unmasked interrupt sources

Note 1. According to interrupt contact destinations, interrupt sources differ. For details on activation sources, see Table 12.3, Cortex-R4/DMACAa Interrupt Vector Table.

Note 2. The 16 priority levels are valid for all sources of CR4 (VIC) vector numbers 1 to 255. Interrupt sources for vector numbers CR4 (VIC) 256 and later have priority lower than sources for vector numbers 1 to 255. For details, see section 12.4.5.1, Restrictions on VIC Priority Levels.

Note 3. When an interrupt signal is selected as the source for activating the DMACAa, generation of the interrupt signal activates the DMACAa but branching to interrupt handling does not proceed at this time. The DMACAa generates a transfer completion interrupt when it completes the data transfer. For details, see section 12.3.1, Selecting Interrupt Request Destinations.

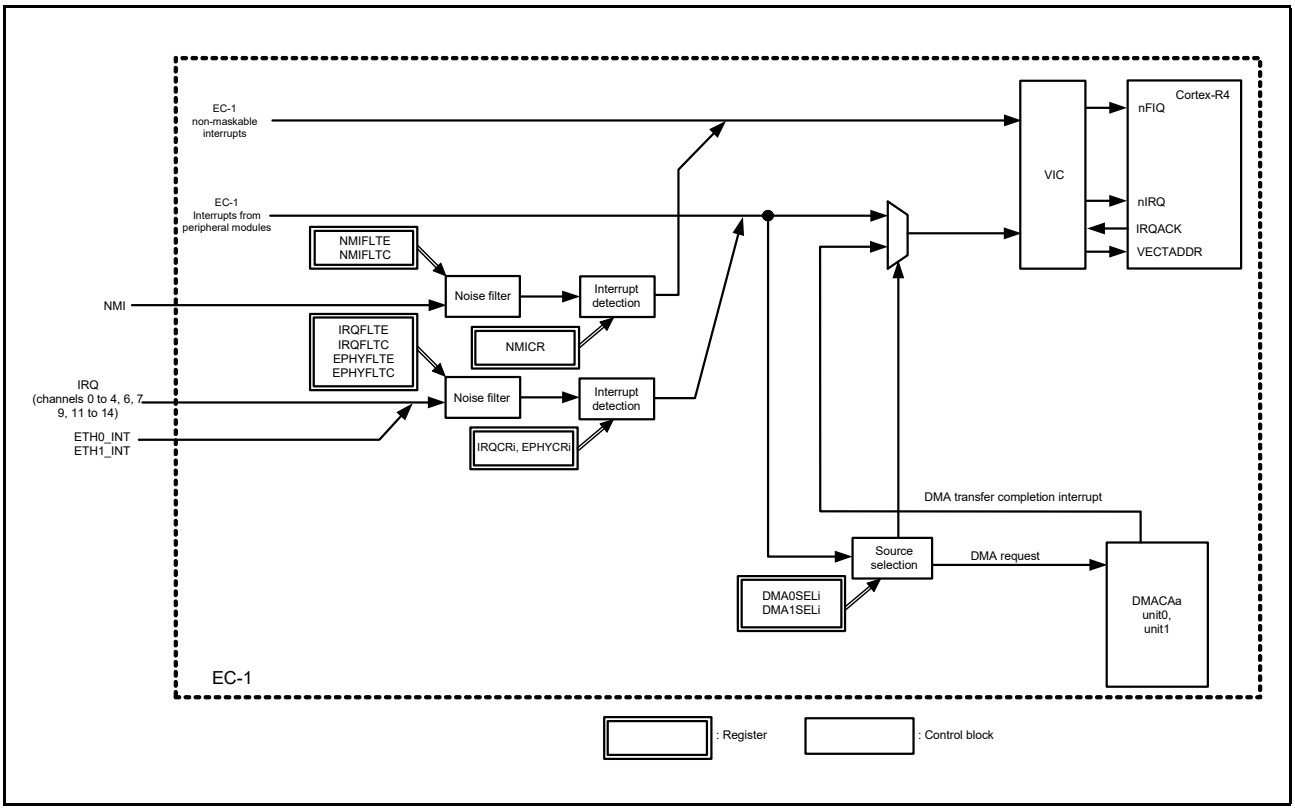


Figure 12.1 Block Diagram of Interrupt Controller

Table 12.2 lists input/output pins that are used by interrupt controllers.

Table 12.2 Input/output Pins for Interrupt Controllers

Pin Name	I/O	Description
NMI	Input	Non-maskable interrupt request pin
IRQ0 to 4, 6, 7, 9, 11 to 14	Input	Maskable interrupt request pin
ETH0_INT	Input	Ethernet PHY0 interrupt request pin
ETH1_INT	Input	Ethernet PHY1 interrupt request pin

12.2 Register Descriptions

12.2.1 IRQ Control Register i (IRQCRi) (i = 0 to 4, 6, 7, 9, 11 to 14)

The IRQCRi register sets the method for detecting the external pin interrupt source (IRQ0 to 4, 6, 7, 9, 11 to 14).

Address(es): ICU.IRQCR0 A009 4200h, ICU.IRQCR1 A009 4204h, ICU.IRQCR2 A009 4208h, ICU.IRQCR3 A009 420Ch, ICU.IRQCR4 A009 4210h, ICU.IRQCR6 A009 4218h, ICU.IRQCR7 A009 421Ch, ICU.IRQCR9 A009 4224h, ICU.IRQCR11 A009 422Ch, ICU.IRQCR12 A009 4230h, ICU.IRQCR13 A009 4234h, ICU.IRQCR14 A009 4238h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	IRQMD[1:0]		—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b3, b2	IRQMD[1:0]	IRQ Detection Sense Select	b3 b2 0 0: Low level 0 1: Falling edge 1 0: Rising edge 1 1: Rising and falling edges	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

IRQMD[1:0] Bits (IRQ Detection Sense Select)

These bits set the method for detecting the external pin interrupt source (IRQ0 to 4, 6, 7, 9, 11 to 14).
For details on the methods for detecting external pin interrupt sources, see section 12.3.3, External Pin Interrupts.

Note: The same detection method should be set to the PLSn register of the VIC.

12.2.2 IRQ Pin Digital Noise Filter Enable Register (IRQFLTE)

The IRQFLTE register sets whether to use the digital noise filter for the external pin interrupt sources (IRQ0 to 4, 6, 7, 9, 11 to 14).

Address(es): ICU.IRQFLTE A009 4240h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	FLTEN14	FLTEN13	FLTEN12	FLTEN11	—	FLTEN9	—	FLTEN7	FLTEN6	—	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FLTEN0	IRQ0 Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b1	FLTEN1	IRQ1 Digital Noise Filter Enable		R/W
b2	FLTEN2	IRQ2 Digital Noise Filter Enable		R/W
b3	FLTEN3	IRQ3 Digital Noise Filter Enable		R/W
b4	FLTEN4	IRQ4 Digital Noise Filter Enable		R/W
b5	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6	FLTEN6	IRQ6 Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b7	FLTEN7	IRQ7 Digital Noise Filter Enable		R/W
b8	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b9	FLTEN9	IRQ9 Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b10	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b11	FLTEN11	IRQ11 Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b12	FLTEN12	IRQ12 Digital Noise Filter Enable		R/W
b13	FLTEN13	IRQ13 Digital Noise Filter Enable		R/W
b14	FLTEN14	IRQ14 Digital Noise Filter Enable		R/W
b31 to b15	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

FLTENi Bit (IRQi Digital Noise Filter Enable) (i = 0 to 4, 6, 7, 9, 11 to 14)

This bit enables the digital noise filter used for the external pin interrupt source (IRQ0 4, 6, 7, 9, 11 to 14).
When this bit is set to 1, the digital noise filter is enabled. When it is cleared to 0, the digital noise filter function is disabled.
The IRQi pin level is sampled at the sampling clock cycle specified with the IRQFLTC.FCLKSELi[1:0] bits. When the sampled level matches three times, the output level from the digital filter changes.
For details on the digital noise filter, see section 12.3.2, Digital Noise Filter.

12.2.3 IRQ Pin Digital Noise Filter Setting Register (IRQFLTC)

The IRQFLTC register sets the digital noise filter sampling clock for external pin interrupt request pins (IRQ0 to 4, 6, 7, 9, 11 to 14).

Address(es): ICU.IRQFLTC A009 4244h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	FCLKSEL14 [1:0]		FCLKSEL13 [1:0]		FCLKSEL12 [1:0]		FCLKSEL11 [1:0]		—	—	FCLKSEL9 [1:0]		—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FCLKSEL7 [1:0]		FCLKSEL6 [1:0]		—	—	FCLKSEL4 [1:0]		FCLKSEL3 [1:0]		FCLKSEL2 [1:0]		FCLKSEL1 [1:0]		FCLKSEL0 [1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	FCLKSEL0[1:0]	IRQ0 Digital Noise Filter Sampling Clock Setting	Odd b Even b 0 0: PCLKB 0 1: PCLKB/8 1 0: PCLKB/32 1 1: PCLKB/64	R/W
b3, b2	FCLKSEL1[1:0]	IRQ1 Digital Noise Filter Sampling Clock Setting		R/W
b5, b4	FCLKSEL2[1:0]	IRQ2 Digital Noise Filter Sampling Clock Setting		R/W
b7, b6	FCLKSEL3[1:0]	IRQ3 Digital Noise Filter Sampling Clock Setting		R/W
b9, b8	FCLKSEL4[1:0]	IRQ4 Digital Noise Filter Sampling Clock Setting		R/W
b11, b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b13, b12	FCLKSEL6[1:0]	IRQ6 Digital Noise Filter Sampling Clock Setting	Odd b Even b 0 0: PCLKB 0 1: PCLKB/8 1 0: PCLKB/32 1 1: PCLKB/64	R/W
b15, b14	FCLKSEL7[1:0]	IRQ9 Digital Noise Filter Sampling Clock Setting		R/W
b17, b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b19, b18	FCLKSEL9[1:0]	IRQ9 Digital Noise Filter Sampling Clock Setting	Odd b Even b 0 0: PCLKB 0 1: PCLKB/8 1 0: PCLKB/32 1 1: PCLKB/64	R/W
b21, b20	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b23, b22	FCLKSEL11[1:0]	IRQ11 Digital Noise Filter Sampling Clock Setting	Odd b Even b 0 0: PCLKB 0 1: PCLKB/8 1 0: PCLKB/32 1 1: PCLKB/64	R/W
b25, b24	FCLKSEL12[1:0]	IRQ12 Digital Noise Filter Sampling Clock Setting		R/W
b27, b26	FCLKSEL13[1:0]	IRQ13 Digital Noise Filter Sampling Clock Setting		R/W
b29, b28	FCLKSEL14[1:0]	IRQ14 Digital Noise Filter Sampling Clock Setting		R/W
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

FCLKSELi[1:0] Bits (IRQi Digital Noise Filter Sampling Clock Setting) (i = 0 to 4, 6, 7, 9, 11 to 14)

These bits select the digital noise filter sampling clock for external pin interrupt request pins (IRQ0 to 4, 6, 7, 9, 11 to 14).

The sampling clock cycle can be selected from the PCLKB (every cycle), PCKLB/8 (once every eight cycles), PCKLB/32 (once every 32 cycles), and PCKLB/64 (once every 64 cycles).

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter. Note that if the digital noise filter is disabled, PCLKB of which interrupts are to be sampled does not stop.

12.2.4 Non-maskable Interrupt Status Register (NMISR)

The NMISR register monitors the status of non-maskable interrupt sources. Writing to this register is ignored.
For information on non-maskable interrupt requests from ECM, read ECMm error source status register m (ECMmESSTRm, m = 0 to 2) for ECM, and check the error source.
Before ending non-maskable interrupt handler processing, read the NMISR register, and check the occurrence status of other non-maskable interrupts. Make sure all bits of the NMISR register are cleared to 0 before ending the interrupt handler processing.

Address(es): ICU.NMISR A009 4248h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EC MST	NMIST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NMIST	NMI Status Flag	0: NMI pin interrupt is not requested. 1: NMI pin interrupt is requested.	R
b1	ECMST	NMI Error Status Flag	0: ECM non-maskable interrupt is not requested. 1: ECM non-maskable interrupt is requested.	R
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R

NMIST Flag (NMI Status Flag)

This flag indicates whether NMI pin interrupts are requested.
The NMIST flag is read only, and it can be cleared to 0 with the NMICLR.NMICLR bit.
[Setting condition]

- When the edge set for the NMICR.NMIMD bit is input for the NMI pin

[Clearing condition]

- When 1 is written to the NMICLR.NMICLR bit

ECMST Flag (ECM Error Status Flag)

This flag indicates whether ECM non-maskable interrupts are requested.
The ECMST flag is read only, and it can be cleared to 0 by the NMICLR.ECMCLR bit.
[Setting condition]

- When ECM non-maskable interrupts are generated

[Clearing condition]

- When 1 is written to the NMICLR.ECMCLR bit

12.2.5 Non-maskable Interrupt Status Clear Register (NMICLR)

The NMICLR register clears NMI or ECM non-maskable interrupt requests.

Address(es): ICU.NMICLR A009 424Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECMCLR	NMICLR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NMICLR	NMI Clear	This bit is read as 0. Writing 1 clears the NMISR.NMIST flag. Writing 0 to this bit is disabled.	R/(W) *1
b1	ECMCLR	ECM Clear	This bit is read as 0. Writing 1 clears the NMISR.ECMST flag. Writing 0 to this bit is disabled.	R/(W) *1
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only 1 can be written.

12.2.6 NMI Pin Interrupt Control Register (NMICR)

The NMICR register sets the method for detecting the NMI pin interrupt.

Address(es): ICU.NMICR A009 4250h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	NMIMD	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b3	NMIMD	NMI Detection Sense Select	0: Falling edge 1: Rising edge	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NMIMD Bit (NMI Detection Setting)

This bit sets the method for detecting the NMI pin interrupt.

12.2.7 NMI Pin Digital Noise Filter Enable Register (NMIFLTE)

The NMIFLTE register sets whether to use the digital noise filter for NMI pin interrupts.

Address(es): ICU.NMIFLTE A009 4254h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NFLTEN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NFLTEN	NMI Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NFLTEN Bit (NMI Digital Noise Filter Enable)

This bit enables the digital noise filter used for NMI pin interrupts.

When this bit is set to 1, the digital noise filter is enabled. When it is cleared to 0, the digital noise filter function is disabled.

The NMI pin level is sampled at the sampling clock cycle specified with the NMIFLTC.NFCLKSEL[1:0] bits. When the sampled level matches three times, the output level from the digital noise filter changes.

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter.

12.2.8 NMI Pin Digital Noise Filter Setting Register (NMIFLTC)

The NMIFLTC register sets the digital noise filter sampling clock for the NMI pin interrupt.

Address(es): ICU.NMIFLTC A009 4258h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NFCLKSEL [1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	NFCLKSEL[1:0]	NMI Digital Noise Filter Sampling Clock Setting	b1 b0 0 0: PCLKB 0 1: PCLKB/8 1 0: PCLKB/32 1 1: PCLKB/64	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NFCLKSEL[1:0] Bits (NMI Digital Noise Filter Sampling Clock Setting)

These bits set the digital noise filter sampling clock of the NMI pin interrupt.

The sampling clock cycle can be selected from the PCLKB (every cycle), PCKLB/8 (once every eight cycles), PCKLB/32 (once every 32 cycles), and PCKLB/64 (once every 64 cycles).

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter. Note that if the digital noise filter is disabled, PCLKB of which interrupts are to be sampled does not stop.

12.2.9 Ethernet PHY Control Register i (EPHYCRi) (i = 0, 1)

The EPHYCRi register sets the method for detecting the Ethernet PHY interrupt source (ETH0_INT/ETH1_INT).

Address(es): ICU.EPHYCR0 A009 425Ch, ICU.EPHYCR1 A009 4260h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	EPHYMD [1:0]		—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b3, b2	EPHYMD[1:0]	Ethernet PHY interrupt Detection Setting	b3 b2 0 0: Low level 0 1: Falling edge 1 0: Rising edge 1 1: Rising and falling edges	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

EPHYMD[1:0] Bits (Ethernet PHY Interrupt Detection Setting)

These bits set the method for detecting the Ethernet PHY interrupt source (ETH0_INT/ETH1_INT). For details on the detection method of Ethernet PHY interrupt request, see section 12.3.3, External Pin Interrupts.

Note: Set the PLS register in the VIC to the same method of detection as is set in this register.

12.2.10 Ethernet PHY Interrupt Request Pin Digital Noise Filter Enable Register (EPHYFLTE)

The EPHYFLTE register enables the digital noise filter used for the Ethernet PHY interrupt source (ETH0_INT/ETH1_INT).

Address(es): ICU.EPHYFLTE A009 4268h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EFLTEN1	EFLTEN0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	EFLTEN0	Ethernet PHY0 Interrupt Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b1	EFLTEN1	Ethernet PHY1 Interrupt Digital Noise Filter Enable	0: Digital noise filter is disabled. 1: Digital noise filter is enabled.	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

EFLTENi Bit (Ethernet PHY Interrupt Digital Noise Filter Enable) (i = 0, 1)

This bit enables the digital noise filter used for Ethernet PHY interrupt source (ETH0_INT/ETH1_INT).

When this bit is set to 1, the digital noise filter is enabled. When it is cleared to 0, the digital noise filter function is disabled.

The ETH0_INT/ETH1_INT pin level is sampled at the sampling clock cycle specified with the EPHYFLTE.FLTENi[1:0] bits. When the sampled level matches three times, the output level from the digital noise filter changes.

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter.

12.2.11 Ethernet PHY Interrupt Request Pin Digital Noise Filter Setting Register (EPHYFLTC)

The EPHYFLTC register sets the digital noise filter sampling clock for the Ethernet PHY interrupt request pin (ETH0_INT/ETH1_INT).

Address(es): ICU.EPHYFLTC A009 426Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	EFCLKSEL1 [1:0]		EFCLKSEL0 [1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	EFCLKSEL0[1:0]	Ethernet PHY0 Interrupts Digital Noise Filter Sampling Clock Setting	Odd b Even b 0 0: PCLKB 0 1: PCLKB/8 1 0: PCLKB/32 1 1: PCLKB/64	R/W
b3, b2	EFCLKSEL1[1:0]	Ethernet PHY1 Interrupts Digital Noise Filter Sampling Clock Setting		R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

EFCLKSELi[1:0] Bits (Ethernet PHYi Interrupt Digital Noise Filter Sampling Clock Setting) (i = 0, 1)

These bits select the digital noise filter sampling clock for Ethernet PHY interrupt request pin (ETH0_INT/ETH1_INT). The sampling clock cycle can be selected from the PCLKB (every cycle), PCKLB/8 (once every eight cycles), PCKLB/32 (once every 32 cycles), and PCKLB/64 (once every 64 cycles).

For details on the digital noise filter, see section 12.3.2, Digital Noise Filter. Note that if the digital noise filter is disabled, PCLKB of which interrupts are to be sampled does not stop.

12.3 Operation

12.3.1 Selecting Interrupt Request Destinations

Table 12.3, Cortex-R4/DMACAa Interrupt Vector Table, is a list of the requesting sources and indicates the source for which the CPU or DMACAa is selectable as the destination. When the CPU is selected as the destination, the processing currently in progress branches to the interrupt handling routine in response to the interrupt request. When the DMACAa is selected as the destination, DMA transfer starts in response to the interrupt request signal. In this case, a DMA transfer completion interrupt is generated on completion of the transfer. Do not select interrupt request destinations that do not have the letter Y in the given request destination column in Table 12.3, Cortex-R4/DMACAa Interrupt Vector Table.

Figure 12.2 shows the flow of selecting an interrupt source when vector number m is allocated to channel N of DMACAa unit 0 as the DMA source. Vector numbers selected in the DMA source select register are not connected to an interrupt controller (VIC), but they are connected as DMA transfer requests to the corresponding channels of DMACAa. On completion of the DMA transfer, the transfer completion interrupt signal for the given channel of the DMACAa is connected as the trigger for interrupt handling by the routine indicated by vector number m for the VIC.

For example, when interrupt vector number 21 (compare match interrupt_ch.0 of CMT unit 0) is selected for IFC0[7:0] of the DMACAa unit 0 source select register 0 (DMA0SEL0), if this interrupt occurs, DMA transfer is requested on channel 0 of DMACAa unit 0. After DMA transfer, if a DMA transfer completion interrupt is generated, the DMA transfer completion request for channel 0 of DMACAa unit 0 is connected to the same interrupt vector number (21) for VIC.

If vector number m is not selected by the source select register, interrupts from external pins and peripheral modules are connected to the interrupt controller VIC for the CPU (Figure 12.3).

Note 1. When the DMACAa is selected as the destination for an interrupt request with vector number m, the DMA transfer completion interrupt signal is conveyed to the interrupt controller as the interrupt for vector number m on completion of the DMA transfer. This means that the detection type for an interrupt with vector number m whose destination is set as the DMACAa should always be edge-sensing regardless of the vector number.

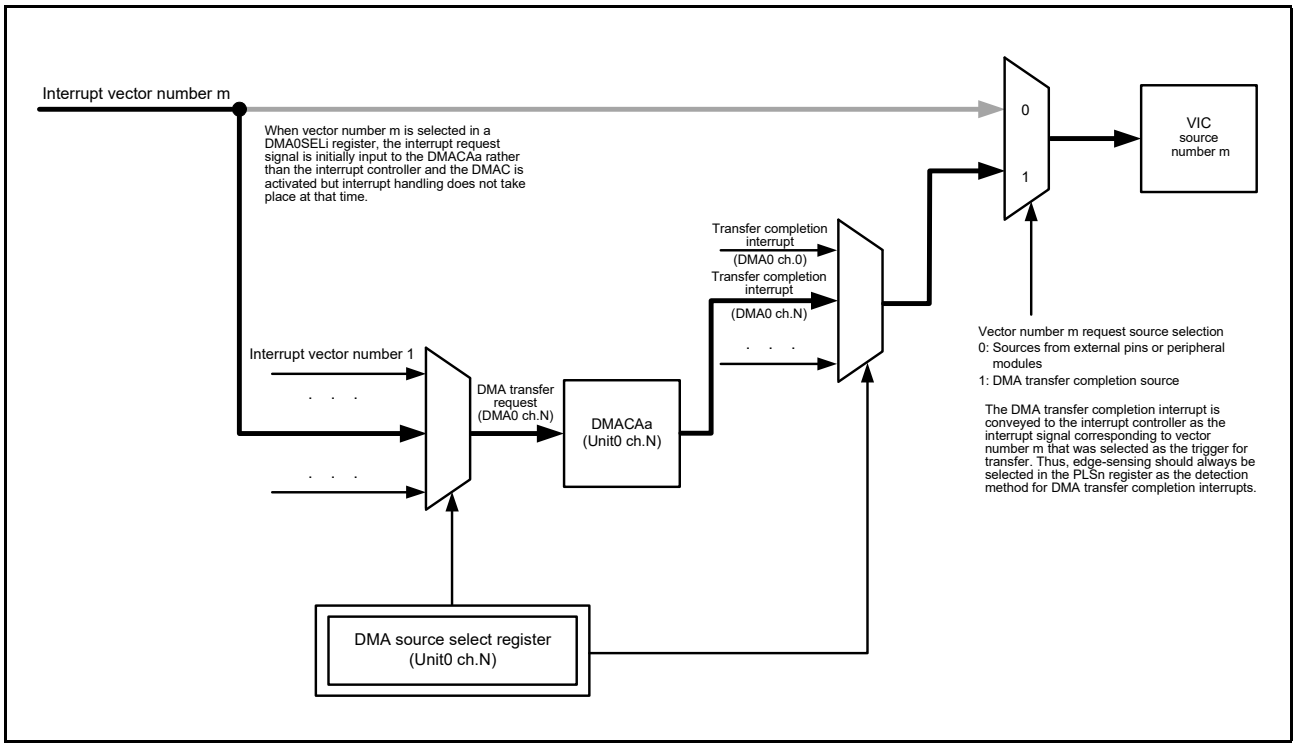


Figure 12.2 DMACAa as the Interrupt Request Destination

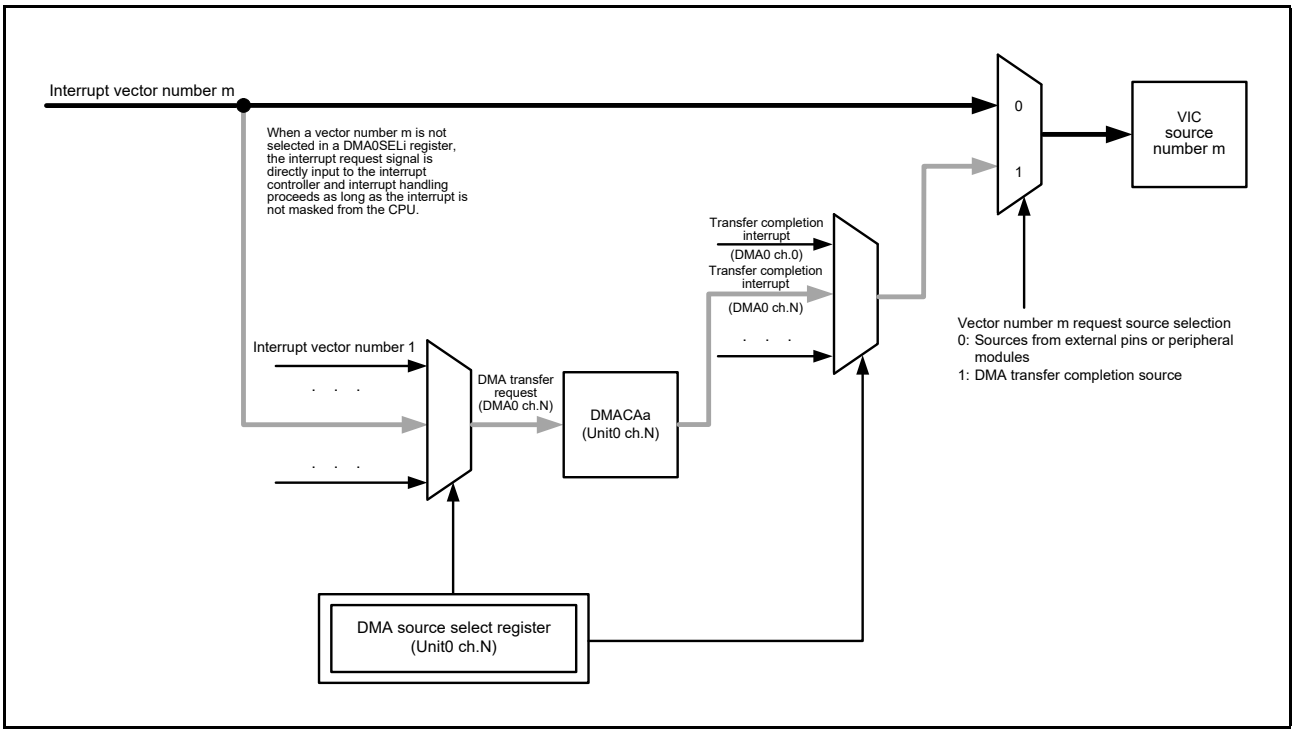


Figure 12.3 CPU (Interrupt Controller) as the Interrupt Request Destination

12.3.2 Digital Noise Filter

The digital noise filter function is provided for the external interrupt request IRQi pins (i = 0 to 4, 6, 7, 9, 11 to 14), NMI pin interrupts, and Ethernet PHY interrupt ETHn_INT pins (n = 0, 1).
The digital noise filter samples input signals at the filter sampling clock (PCLKB), and removes the pulses of which length is less than three sampling times.
To use the digital noise filter for the IRQi pins (IRQ0 to 4, 6, 7, 9, 11 to 14), set the sampling clock cycle (PCLKB, PCLKB/8, PCLKB/32, or PCLKB/64) with the IRQFLTC.FCLKSELi[1:0] bits (IRQ0 to 4, 6, 7, 9, 11 to 14), and set the IRQFLTE.FLTENi bits (IRQ0 to 4, 6, 7, 9, 11 to 14) to 1.
To use the digital noise filter for the NMI pin interrupt, set the sampling clock cycle (PCLKB, PCLKB/8, PCLKB/32, or PCLKB/64) with the NMICR.NFCLKSEL[1:0] bits, and set the NMICR.NFLTEN bit to 1.
To use the digital noise filter for the Ethernet PHY interrupt ETHn_INT pins (n = 0, 1), set the sampling clock cycle (PCLKB, PCLKB/8, PCLKB/32, or PCLKB/64) with the EPHYFLTC.FCLKSELn[1:0] bits (n = 0, 1), and set the EPHYFLTE.FLTENn bits (n = 0, 1) to 1.
Figure 12.4 shows an example of digital noise filter operation.

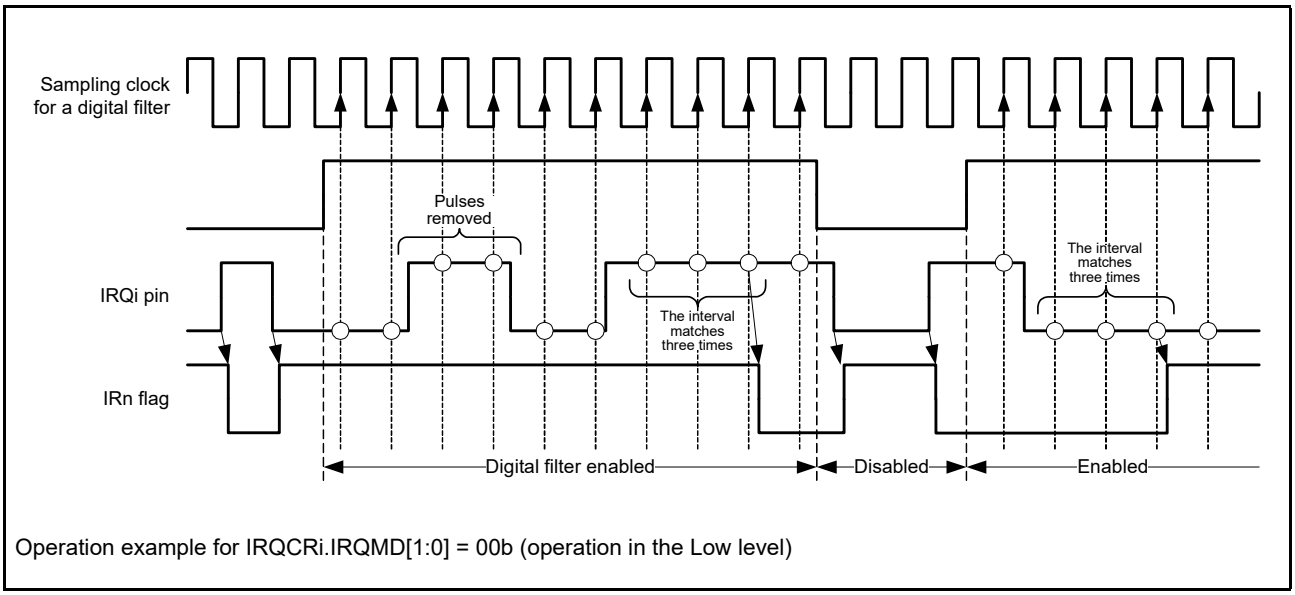


Figure 12.4 Digital Noise Filter Operation Example

12.3.3 External Pin Interrupts

The procedure for connecting an external pin interrupt to Cortex-R4 is shown below. For details on VIC, see section 12.4, Cortex-R4 Vector Interrupt Controller (VIC).
To use the external pins at their falling edges or rising and falling edges, see section 12.5.1, Using “Falling-Edge” or “Rising and Falling Edges” Detection with the External Pin Interrupts.

[For IRQ pins]

1. Clear the applicable IENn bit to 0 (set the IECn bit).
2. Clear the IRQFLTE.FLTENi bit to 0.*1
3. Set the digital noise filter sampling clock with the IRQFLTC.FCLKSEL[1:0] bits.*1
4. Set the Pmn I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).
5. Set the I/O port (PmnPFS.ISEL bit).
6. Set the method of detection with the IRQCRi.IRQMD[1:0] bits.
7. Set the IRQFLTE.FLTENi bit to 1.*1
8. Set the applicable PICn register to 1 (when an edge is detected).
9. Set the applicable IENn bit to 1.

Note 1. This setting is only required when the digital noise filter is to be used.

[For ETH0_INT/ETH1_INT]

1. Set the applicable IENn bit to 0 (set the IECn bit).
2. Clear the EPHYFLTE.EFLTENi bit to 0.*1
3. Set the digital noise filter sampling clock with the EPHYFLTC.EFCLKSEL[1:0] bits.*1
4. Set the Pmn I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).
5. Set the I/O port (the PmnPFS.PSEL[5:0] bits and PMR register).
6. Set the method of detection with the EPHYCRi.EPHYMD[1:0] bits.
7. Set the EPHYFLTE.EFLTENi bit to 1.*1
8. Set the applicable PICn register to 1 (when an edge is detected).
9. Set the applicable IENn bit to 1.

Note 1. This setting is only required when the digital noise filter is to be used.

12.3.4 NMI Pin Interrupts

Pins that can be used as NMI pins serve as general I/O ports after a reset. To use them as NMI pins, the following procedure is required. Note that setting these pins to serve as general I/O ports after setting them to serve as the NMI pins is prohibited.
To use the NMI pins at their falling edges, see section 12.5.2, Using Falling-Edge Detection with the NMI Pin.

1. Set the NMIFLTE.NFLTEN bit to 0.*1
2. Set the sampling clock of the digital noise filter with the NMIFLTC.NFCLKSEL[1:0] bits.*1
3. Set edge detection with the NMICR.NMIMD bit.
4. Set the NMICLR.NMICLR bit to 1 and clear the NMISR.NMIST flag to 0.
5. Set the NMIFLTE.NFLTEN bit to 1.*1
6. Set the P35 direction control bit in the port direction register (PDR) of the I/O port to 10b (input).
7. Set the I/O port (P35PFS.ISEL bit) and confirm the setting.

Note 1. This setting is only required when the digital noise filter is to be used.

12.4 Cortex-R4 Vector Interrupt Controller (VIC)

12.4.1 Overview

The EC-1 has the vector interrupt controller (VIC) to control interrupts for Cortex-R4. Non-maskable interrupt requests from the NMI pin or ECM are treated as FIQ interrupts and are always accepted at high-speed. Interrupts from external pins other than the NMI pin and those from on-chip peripheral modules are accepted as IRQ interrupts (maskable interrupts). The vector addresses of the individual IRQ interrupt sources are stored in the interrupt address storage registers (VADn). When an IRQ interrupt occurs, the interrupt controller provides the Cortex-R4 with the address in VADn as the destination for branching, so the program counter directly branches to the address set in VADn.

12.4.2 Register Descriptions

12.4.2.1 IRQ Status Register n (IRQSn) (n = 0 to 9)

The IRQSn (n = 0 to 9) register indicates the interrupt status after IRQ interrupt mask. This register is enabled when an interrupt is enabled (IENn = 1). Interrupt status is not reflected when an interrupt is disabled (IENn = 0).
This register can only be read in 32-bit units.
Before completing the level interrupt, use the register to make sure no interrupt is requested. (See section 12.4.4.3, (2) IRQ Interrupt (Level interrupt)).

- IRQS0

Address(es): VIC.IRQS0 A001 0000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ31	IRQ30	IRQ29	IRQ28	IRQ27	IRQ26	IRQ25	IRQ24	IRQ23	IRQ22	IRQ21	IRQ20	IRQ19	IRQ18	IRQ17	IRQ16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ15	IRQ14	IRQ13	IRQ12	IRQ11	IRQ10	IRQ9	IRQ8	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R
b31 to b1	IRQ[31:1]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS1

Address(es): VIC.IRQS1 A001 0004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ63	IRQ62	IRQ61	IRQ60	IRQ59	IRQ58	IRQ57	IRQ56	IRQ55	IRQ54	IRQ53	IRQ52	IRQ51	IRQ50	IRQ49	IRQ48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ47	IRQ46	IRQ45	IRQ44	IRQ43	IRQ42	IRQ41	IRQ40	IRQ39	IRQ38	IRQ37	IRQ36	IRQ35	IRQ34	IRQ33	IRQ32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[63:32]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

IRQi Flag (Interrupt Status Flag) (i = 1 to 63)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

- IRQS2

Address(es): VIC.IRQS2 A001 0008h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ95	IRQ94	IRQ93	IRQ92	IRQ91	IRQ90	IRQ89	IRQ88	IRQ87	IRQ86	IRQ85	IRQ84	IRQ83	IRQ82	IRQ81	IRQ80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ79	IRQ78	IRQ77	IRQ76	IRQ75	IRQ74	IRQ73	IRQ72	IRQ71	IRQ70	IRQ69	IRQ68	IRQ67	IRQ66	IRQ65	IRQ64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[95:64]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS3

Address(es): VIC.IRQS3 A001 000Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ127	IRQ126	IRQ125	IRQ124	IRQ123	IRQ122	IRQ121	IRQ120	IRQ119	IRQ118	IRQ117	IRQ116	IRQ115	IRQ114	IRQ113	IRQ112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ111	IRQ110	IRQ109	IRQ108	IRQ107	IRQ106	IRQ105	IRQ104	IRQ103	IRQ102	IRQ101	IRQ100	IRQ99	IRQ98	IRQ97	IRQ96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[127:96]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

IRQi Flag (Interrupt Status Flag) (i = 64 to 127)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

- IRQS4

Address(es): VIC.IRQS4 A001 0010h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ159	IRQ158	IRQ157	IRQ156	IRQ155	IRQ154	IRQ153	IRQ152	IRQ151	IRQ150	IRQ149	IRQ148	IRQ147	IRQ146	IRQ145	IRQ144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ143	IRQ142	IRQ141	IRQ140	IRQ139	IRQ138	IRQ137	IRQ136	IRQ135	IRQ134	IRQ133	IRQ132	IRQ131	IRQ130	IRQ129	IRQ128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[159:128]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS5

Address(es): VIC.IRQS5 A001 0014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ191	IRQ190	IRQ189	IRQ188	IRQ187	IRQ186	IRQ185	IRQ184	IRQ183	IRQ182	IRQ181	IRQ180	IRQ179	IRQ178	IRQ177	IRQ176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ175	IRQ174	IRQ173	IRQ172	IRQ171	IRQ170	IRQ169	IRQ168	IRQ167	IRQ166	IRQ165	IRQ164	IRQ163	IRQ162	IRQ161	IRQ160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[191:160]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

IRQi Flag (Interrupt Status Flag) (i = 128 to 191)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

- IRQS6

Address(es): VIC.IRQS6 A001 0018h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ223	IRQ222	IRQ221	IRQ220	IRQ219	IRQ218	IRQ217	IRQ216	IRQ215	IRQ214	IRQ213	IRQ212	IRQ211	IRQ210	IRQ209	IRQ208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ207	IRQ206	IRQ205	IRQ204	IRQ203	IRQ202	IRQ201	IRQ200	IRQ199	IRQ198	IRQ197	IRQ196	IRQ195	IRQ194	IRQ193	IRQ192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[223:192]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS7

Address(es): VIC.IRQS7 A001 001Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ255	IRQ254	IRQ253	IRQ252	IRQ251	IRQ250	IRQ249	IRQ248	IRQ247	IRQ246	IRQ245	IRQ244	IRQ243	IRQ242	IRQ241	IRQ240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ239	IRQ238	IRQ237	IRQ236	IRQ235	IRQ234	IRQ233	IRQ232	IRQ231	IRQ230	IRQ229	IRQ228	IRQ227	IRQ226	IRQ225	IRQ224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[255:224]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

IRQi Flag (Interrupt Status Flag) (i = 192 to 255)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

- IRQS8

Address(es): VIC.IRQS8 A001 1000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IRQ287	IRQ286	IRQ285	IRQ284	IRQ283	IRQ282	IRQ281	IRQ280	IRQ279	IRQ278	IRQ277	IRQ276	IRQ275	IRQ274	IRQ273	IRQ272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IRQ271	IRQ270	IRQ269	IRQ268	IRQ267	IRQ266	IRQ265	IRQ264	IRQ263	IRQ262	IRQ261	IRQ260	IRQ259	IRQ258	IRQ257	IRQ256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IRQ[287:256]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R

- IRQS9

Address(es): VIC.IRQS9 A001 1004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IRQ300	IRQ299	IRQ298	IRQ297	IRQ296	IRQ295	IRQ294	IRQ293	IRQ292	IRQ291	IRQ290	IRQ289	IRQ288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b12 to b0	IRQ[300:288]	Interrupt Status Flag	0: IRQ interrupt is not requested. 1: IRQ interrupt is requested.	R
b31 to b13	—	Reserved	These bits are read as 0.	R

IRQi Flag (Interrupt Status Flag) (i = 256 to 300)

This flag indicates the interrupt status after interrupt mask by the IEN registers.

12.4.2.2 Interrupt Input Status Register n (RAISn) (n = 0 to 9)

The RAISn (n = 0 to 9) register indicates the interrupt input status before IRQ (maskable) interrupt mask. The interrupt status is reflected to this register regardless of the IENn register setting (interrupt enabled or disabled).
This register can only be read in 32-bit units. The states of an interrupt source can be confirmed while the interrupt is disabled (the corresponding bit in IENn is 0) by, for example, polling the source bit (see section 12.4.4.6, Handling IRQ Interrupt Source Conditions by Polling).

- RAIS0

Address(es): VIC.RAIS0 A001 0040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI31	RAI30	RAI29	RAI28	RAI27	RAI26	RAI25	RAI24	RAI23	RAI22	RAI21	RAI20	RAI19	RAI18	RAI17	RAI16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI15	RAI14	RAI13	RAI12	RAI11	RAI10	RAI9	RAI8	RAI7	RAI6	RAI5	RAI4	RAI3	RAI2	RAI1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R
b31 to b1	RAI[31:1]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS1

Address(es): VIC.RAIS1 A001 0044h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI63	RAI62	RAI61	RAI60	RAI59	RAI58	RAI57	RAI56	RAI55	RAI54	RAI53	RAI52	RAI51	RAI50	RAI49	RAI48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI47	RAI46	RAI45	RAI44	RAI43	RAI42	RAI41	RAI40	RAI39	RAI38	RAI37	RAI36	RAI35	RAI34	RAI33	RAI32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[63:32]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

RAIi Flag (Interrupt Input Status Flag) (i = 1 to 63)

This flag indicates the interrupt request input status before interrupt mask.

- RAIS2

Address(es): VIC.RAIS2 A001 0048h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI95	RAI94	RAI93	RAI92	RAI91	RAI90	RAI89	RAI88	RAI87	RAI86	RAI85	RAI84	RAI83	RAI82	RAI81	RAI80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI79	RAI78	RAI77	RAI76	RAI75	RAI74	RAI73	RAI72	RAI71	RAI70	RAI69	RAI68	RAI67	RAI66	RAI65	RAI64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[95:64]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS3

Address(es): VIC.RAIS3 A001 004Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI127	RAI126	RAI125	RAI124	RAI123	RAI122	RAI121	RAI120	RAI119	RAI118	RAI117	RAI116	RAI115	RAI114	RAI113	RAI112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI111	RAI110	RAI109	RAI108	RAI107	RAI106	RAI105	RAI104	RAI103	RAI102	RAI101	RAI100	RAI99	RAI98	RAI97	RAI96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[127:96]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

RAIi Flag (Interrupt Input Status Flag) (i = 64 to 127)

This flag indicates the interrupt request input status before interrupt mask.

- RAIS4

Address(es): VIC.RAIS4 A001 0050h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI159	RAI158	RAI157	RAI156	RAI155	RAI154	RAI153	RAI152	RAI151	RAI150	RAI149	RAI148	RAI147	RAI146	RAI145	RAI144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI143	RAI142	RAI141	RAI140	RAI139	RAI138	RAI137	RAI136	RAI135	RAI134	RAI133	RAI132	RAI131	RAI130	RAI129	RAI128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[159:128]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS5

Address(es): VIC.RAIS5 A001 0054h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI191	RAI190	RAI189	RAI188	RAI187	RAI186	RAI185	RAI184	RAI183	RAI182	RAI181	RAI180	RAI179	RAI178	RAI177	RAI176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI175	RAI174	RAI173	RAI172	RAI171	RAI170	RAI169	RAI168	RAI167	RAI166	RAI165	RAI164	RAI163	RAI162	RAI161	RAI160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[191:160]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

RAIi Flag (Interrupt Input Status Flag) (i = 128 to 191)

This flag indicates the interrupt request input status before interrupt mask.

- RAIS6

Address(es): VIC.RAIS6 A001 0058h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI223	RAI222	RAI221	RAI220	RAI219	RAI218	RAI217	RAI216	RAI215	RAI214	RAI213	RAI212	RAI211	RAI210	RAI209	RAI208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI207	RAI206	RAI205	RAI204	RAI203	RAI202	RAI201	RAI200	RAI199	RAI198	RAI197	RAI196	RAI195	RAI194	RAI193	RAI192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[223:192]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS7

Address(es): VIC.RAIS7 A001 005Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI255	RAI254	RAI253	RAI252	RAI251	RAI250	RAI249	RAI248	RAI247	RAI246	RAI245	RAI244	RAI243	RAI242	RAI241	RAI240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI239	RAI238	RAI237	RAI236	RAI235	RAI234	RAI233	RAI232	RAI231	RAI230	RAI229	RAI228	RAI227	RAI226	RAI225	RAI224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[255:224]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

RAIi Flag (Interrupt Input Status Flag) (i = 192 to 255)

This flag indicates the interrupt request input status before interrupt mask.

- RAIS8

Address(es): VIC.RAIS8 A001 1040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RAI287	RAI286	RAI285	RAI284	RAI283	RAI282	RAI281	RAI280	RAI279	RAI278	RAI277	RAI276	RAI275	RAI274	RAI273	RAI272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RAI271	RAI270	RAI269	RAI268	RAI267	RAI266	RAI265	RAI264	RAI263	RAI262	RAI261	RAI260	RAI259	RAI258	RAI257	RAI256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RAI[287:256]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R

- RAIS9

Address(es): VIC.RAIS9 A001 1044h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	RAI300	RAI299	RAI298	RAI297	RAI296	RAI295	RAI294	RAI293	RAI292	RAI291	RAI290	RAI289	RAI288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b12 to b0	RAI[300:288]	Interrupt Input Status Flag	0: Interrupt is not requested. 1: Interrupt is requested.	R
b31 to b13	—	Reserved	These bits are read as 0.	R

RAIi Flag (Interrupt Input Status Flag) (i = 256 to 300)

This flag indicates the interrupt request input status before interrupt mask.

12.4.2.3 Interrupt Enable Register n (IENn) (n = 0 to 9)

The IENn (n = 0 to 9) register enables or masks IRQ interrupts. When it is reset, all interrupt requests are masked. When a bit of this register is set to 1, it cannot be cleared to 0. To clear the bit to 0, use interrupt enable clear register n (IECn).
This register can only be read and written in 32-bit units.

- IEN0

Address(es): VIC.IEN0 A001 0080h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN31	IEN30	IEN29	IEN28	IEN27	IEN26	IEN25	IEN24	IEN23	IEN22	IEN21	IEN20	IEN19	IEN18	IEN17	IEN16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN15	IEN14	IEN13	IEN12	IEN11	IEN10	IEN9	IEN8	IEN7	IEN6	IEN5	IEN4	IEN3	IEN2	IEN1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R/W
b31 to b1	IEN[31:1]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN1

Address(es): VIC.IEN1 A001 0084h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN63	IEN62	IEN61	IEN60	IEN59	IEN58	IEN57	IEN56	IEN55	IEN54	IEN53	IEN52	IEN51	IEN50	IEN49	IEN48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN47	IEN46	IEN45	IEN44	IEN43	IEN42	IEN41	IEN40	IEN39	IEN38	IEN37	IEN36	IEN35	IEN34	IEN33	IEN32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[63:32]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

IENi Bit (Interrupt Request Enable) (i = 1 to 63)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

- IEN2

Address(es): VIC.IEN2 A001 0088h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN95	IEN94	IEN93	IEN92	IEN91	IEN90	IEN89	IEN88	IEN87	IEN86	IEN85	IEN84	IEN83	IEN82	IEN81	IEN80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN79	IEN78	IEN77	IEN76	IEN75	IEN74	IEN73	IEN72	IEN71	IEN70	IEN69	IEN68	IEN67	IEN66	IEN65	IEN64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[95:64]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN3

Address(es): VIC.IEN3 A001 008Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN127	IEN126	IEN125	IEN124	IEN123	IEN122	IEN121	IEN120	IEN119	IEN118	IEN117	IEN116	IEN115	IEN114	IEN113	IEN112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN111	IEN110	IEN109	IEN108	IEN107	IEN106	IEN105	IEN104	IEN103	IEN102	IEN101	IEN100	IEN99	IEN98	IEN97	IEN96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[127:96]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

IENi Bit (Interrupt Request Enable) (i = 64 to 127)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

- IEN4

Address(es): VIC.IEN4 A001 0090h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN159	IEN158	IEN157	IEN156	IEN155	IEN154	IEN153	IEN152	IEN151	IEN150	IEN149	IEN148	IEN147	IEN146	IEN145	IEN144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN143	IEN142	IEN141	IEN140	IEN139	IEN138	IEN137	IEN136	IEN135	IEN134	IEN133	IEN132	IEN131	IEN130	IEN129	IEN128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[159:128]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN5

Address(es): VIC.IEN5 A001 0094h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN191	IEN190	IEN189	IEN188	IEN187	IEN186	IEN185	IEN184	IEN183	IEN182	IEN181	IEN180	IEN179	IEN178	IEN177	IEN176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN175	IEN174	IEN173	IEN172	IEN171	IEN170	IEN169	IEN168	IEN167	IEN166	IEN165	IEN164	IEN163	IEN162	IEN161	IEN160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[191:160]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

IENi Bit (Interrupt Request Enable) (i = 128 to 191)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

- IEN6

Address(es): VIC.IEN6 A001 0098h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN223	IEN222	IEN221	IEN220	IEN219	IEN218	IEN217	IEN216	IEN215	IEN214	IEN213	IEN212	IEN211	IEN210	IEN209	IEN208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN207	IEN206	IEN205	IEN204	IEN203	IEN202	IEN201	IEN200	IEN199	IEN198	IEN197	IEN196	IEN195	IEN194	IEN193	IEN192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[223:192]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN7

Address(es): VIC.IEN7 A001 009Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN255	IEN254	IEN253	IEN252	IEN251	IEN250	IEN249	IEN248	IEN247	IEN246	IEN245	IEN244	IEN243	IEN242	IEN241	IEN240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN239	IEN238	IEN237	IEN236	IEN235	IEN234	IEN233	IEN232	IEN231	IEN230	IEN229	IEN228	IEN227	IEN226	IEN225	IEN224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[255:224]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

IENi Bit (Interrupt Request Enable) (i = 192 to 255)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

- IEN8

Address(es): VIC.IEN8 A001 1080h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEN287	IEN286	IEN285	IEN284	IEN283	IEN282	IEN281	IEN280	IEN279	IEN278	IEN277	IEN276	IEN275	IEN274	IEN273	IEN272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEN271	IEN270	IEN269	IEN268	IEN267	IEN266	IEN265	IEN264	IEN263	IEN262	IEN261	IEN260	IEN259	IEN258	IEN257	IEN256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEN[287:256]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W

- IEN9

Address(es): VIC.IEN9 A001 1084h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IEN300	IEN299	IEN298	IEN297	IEN296	IEN295	IEN294	IEN293	IEN292	IEN291	IEN290	IEN289	IEN288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b12 to b0	IEN[300:288]	Interrupt Request Enable	0: Interrupt is masked (disabled). 1: Interrupt is enabled.	R/W
b31 to b13	—	Reserved	These bits are always read as 0. When written, always write 0.	R/W

IENi Bit (Interrupt Request Enable) (i = 256 to 300)

This bit specifies interrupt request enable settings. When the bit is enabled, it cannot be masked with the IENn register. Perform interrupt mask with the IECn register.

12.4.2.4 Interrupt Enable Clear Register n (IECn) (n = 0 to 9)

The IECn (n = 0 to 9) register clears a bit of the IENn register, and masks (disables) the applicable interrupt request. This register can only be written in 32-bit units.
If the value of the IECn register is to be changed, do so while interrupts are disabled. To disable interrupts, set the I bit in the CPSR register of the ARM CPU to 1.

- IEC0

Address(es): VIC.IEC0 A001 00A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC31	IEC30	IEC29	IEC28	IEC27	IEC26	IEC25	IEC24	IEC23	IEC22	IEC21	IEC20	IEC19	IEC18	IEC17	IEC16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC15	IEC14	IEC13	IEC12	IEC11	IEC10	IEC9	IEC8	IEC7	IEC6	IEC5	IEC4	IEC3	IEC2	IEC1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	The write value should be 0.	W
b31 to b1	IEC[31:1]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC1

Address(es): VIC.IEC1 A001 00A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC63	IEC62	IEC61	IEC60	IEC59	IEC58	IEC57	IEC56	IEC55	IEC54	IEC53	IEC52	IEC51	IEC50	IEC49	IEC48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC47	IEC46	IEC45	IEC44	IEC43	IEC42	IEC41	IEC40	IEC39	IEC38	IEC37	IEC36	IEC35	IEC34	IEC33	IEC32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[63:32]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

IECi Bit (Interrupt Request Clear) (i = 1 to 63)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

- IEC2

Address(es): VIC.IEC2 A001 00A8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC95	IEC94	IEC93	IEC92	IEC91	IEC90	IEC89	IEC88	IEC87	IEC86	IEC85	IEC84	IEC83	IEC82	IEC81	IEC80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC79	IEC78	IEC77	IEC76	IEC75	IEC74	IEC73	IEC72	IEC71	IEC70	IEC69	IEC68	IEC67	IEC66	IEC65	IEC64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[95:64]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC3

Address(es): VIC.IEC3 A001 00ACh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC127	IEC126	IEC125	IEC124	IEC123	IEC122	IEC121	IEC120	IEC119	IEC118	IEC117	IEC116	IEC115	IEC114	IEC113	IEC112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC111	IEC110	IEC109	IEC108	IEC107	IEC106	IEC105	IEC104	IEC103	IEC102	IEC101	IEC100	IEC99	IEC98	IEC97	IEC96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[127:96]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

IECi Bit (Interrupt Request Clear) (i = 64 to 127)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

- IEC4

Address(es): VIC.IEC4 A001 00B0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC159	IEC158	IEC157	IEC156	IEC155	IEC154	IEC153	IEC152	IEC151	IEC150	IEC149	IEC148	IEC147	IEC146	IEC145	IEC144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC143	IEC142	IEC141	IEC140	IEC139	IEC138	IEC137	IEC136	IEC135	IEC134	IEC133	IEC132	IEC131	IEC130	IEC129	IEC128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[159:128]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC5

Address(es): VIC.IEC5 A001 00B4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC191	IEC190	IEC189	IEC188	IEC187	IEC186	IEC185	IEC184	IEC183	IEC182	IEC181	IEC180	IEC179	IEC178	IEC177	IEC176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC175	IEC174	IEC173	IEC172	IEC171	IEC170	IEC169	IEC168	IEC167	IEC166	IEC165	IEC164	IEC163	IEC162	IEC161	IEC160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[191:160]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

IECi Bit (Interrupt Request Clear) (i = 128 to 191)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

- IEC6

Address(es): VIC.IEC6 A001 00B8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC223	IEC222	IEC221	IEC220	IEC219	IEC218	IEC217	IEC216	IEC215	IEC214	IEC213	IEC212	IEC211	IEC210	IEC209	IEC208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC207	IEC206	IEC205	IEC204	IEC203	IEC202	IEC201	IEC200	IEC199	IEC198	IEC197	IEC196	IEC195	IEC194	IEC193	IEC192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[223:192]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC7

Address(es): VIC.IEC7 A001 00BCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC255	IEC254	IEC253	IEC252	IEC251	IEC250	IEC249	IEC248	IEC247	IEC246	IEC245	IEC244	IEC243	IEC242	IEC241	IEC240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC239	IEC238	IEC237	IEC236	IEC235	IEC234	IEC233	IEC232	IEC231	IEC230	IEC229	IEC228	IEC227	IEC226	IEC225	IEC224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[255:224]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

IECi Bit (Interrupt Request Clear) (i = 192 to 255)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

- IEC8

Address(es): VIC.IEC8 A001 10A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	IEC287	IEC286	IEC285	IEC284	IEC283	IEC282	IEC281	IEC280	IEC279	IEC278	IEC277	IEC276	IEC275	IEC274	IEC273	IEC272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IEC271	IEC270	IEC269	IEC268	IEC267	IEC266	IEC265	IEC264	IEC263	IEC262	IEC261	IEC260	IEC259	IEC258	IEC257	IEC256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	IEC[287:256]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W

- IEC9

Address(es): VIC.IEC9 A001 10A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IEC300	IEC299	IEC298	IEC297	IEC296	IEC295	IEC294	IEC293	IEC292	IEC291	IEC290	IEC289	IEC288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b12 to b0	IEC[300:288]	Interrupt Request Clear	0: Nothing is changed. 1: The applicable bit of the IEN register of which interrupt is masked (disabled) is cleared to 0.	W
b31 to b13	—	Reserved	The write value should be 0.	W

IECi Bit (Interrupt Request Clear) (i = 256 to 300)

This bit specifies settings for masking (disabling) interrupt requests. When 1 is set to a bit, the same bit of the IEN register is cleared to 0, and the interrupt request is masked (disabled).

12.4.2.5 Interrupt Detection Type Selection Register n (PLSn) (n = 0 to 9)

The PLSn (n = 0 to 9) register detects the edge or level for each interrupt input.
This register can only be read and written in 32-bit units.

- PLS0

Address(es): VIC.PLS0 A001 0100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS31	PLS30	PLS29	PLS28	PLS27	PLS26	PLS25	PLS24	PLS23	PLS22	PLS21	PLS20	PLS19	PLS18	PLS17	PLS16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS15	PLS14	PLS13	PLS12	PLS11	PLS10	PLS9	PLS8	PLS7	PLS6	PLS5	PLS4	PLS3	PLS2	PLS1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b31 to b1	PLS[31:1]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS1

Address(es): VIC.PLS1 A001 0104h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS63	PLS62	PLS61	PLS60	PLS59	PLS58	PLS57	PLS56	PLS55	PLS54	PLS53	PLS52	PLS51	PLS50	PLS49	PLS48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS47	PLS46	PLS45	PLS44	PLS43	PLS42	PLS41	PLS40	PLS39	PLS38	PLS37	PLS36	PLS35	PLS34	PLS33	PLS32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[63:32]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 1 to 63)

This bit selects the interrupt input detection type.
PLS[63:1] corresponds to vector numbers 63 to 1.

- PLS2

Address(es): VIC.PLS2 A001 0108h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS95	PLS94	PLS93	PLS92	PLS91	PLS90	PLS89	PLS88	PLS87	PLS86	PLS85	PLS84	PLS83	PLS82	PLS81	PLS80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS79	PLS78	PLS77	PLS76	PLS75	PLS74	PLS73	PLS72	PLS71	PLS70	PLS69	PLS68	PLS67	PLS66	PLS65	PLS64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[95:64]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS3

Address(es): VIC.PLS3 A001 010Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS127	PLS126	PLS125	PLS124	PLS123	PLS122	PLS121	PLS120	PLS119	PLS118	PLS117	PLS116	PLS115	PLS114	PLS113	PLS112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS111	PLS110	PLS109	PLS108	PLS107	PLS106	PLS105	PLS104	PLS103	PLS102	PLS101	PLS100	PLS99	PLS98	PLS97	PLS96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[127:96]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 64 to 127)

This bit selects the interrupt input detection type.

PLS[127:64] corresponds to vector numbers 127 to 64.

- PLS4

Address(es): VIC.PLS4 A001 0110h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS159	PLS158	PLS157	PLS156	PLS155	PLS154	PLS153	PLS152	PLS151	PLS150	PLS149	PLS148	PLS147	PLS146	PLS145	PLS144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS143	PLS142	PLS141	PLS140	PLS139	PLS138	PLS137	PLS136	PLS135	PLS134	PLS133	PLS132	PLS131	PLS130	PLS129	PLS128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[159:128]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS5

Address(es): VIC.PLS5 A001 0114h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS191	PLS190	PLS189	PLS188	PLS187	PLS186	PLS185	PLS184	PLS183	PLS182	PLS181	PLS180	PLS179	PLS178	PLS177	PLS176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS175	PLS174	PLS173	PLS172	PLS171	PLS170	PLS169	PLS168	PLS167	PLS166	PLS165	PLS164	PLS163	PLS162	PLS161	PLS160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[191:160]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 128 to 191)

This bit selects the interrupt input detection type.

PLS[191:128] corresponds to vector numbers 191 to 28.

- PLS6

Address(es): VIC.PLS6 A001 0118h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS223	PLS222	PLS221	PLS220	PLS219	PLS218	PLS217	PLS216	PLS215	PLS214	PLS213	PLS212	PLS211	PLS210	PLS209	PLS208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS207	PLS206	PLS205	PLS204	PLS203	PLS202	PLS201	PLS200	PLS199	PLS198	PLS197	PLS196	PLS195	PLS194	PLS193	PLS192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[223:192]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS7

Address(es): VIC.PLS7 A001 011Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS255	PLS254	PLS253	PLS252	PLS251	PLS250	PLS249	PLS248	PLS247	PLS246	PLS245	PLS244	PLS243	PLS242	PLS241	PLS240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS239	PLS238	PLS237	PLS236	PLS235	PLS234	PLS233	PLS232	PLS231	PLS230	PLS229	PLS228	PLS227	PLS226	PLS225	PLS224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[255:224]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 192 to 255)

This bit selects the interrupt input detection type.

PLS[255:192] corresponds to vector numbers 255 to 192.

- PLS8

Address(es): VIC.PLS8 A001 1100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PLS287	PLS286	PLS285	PLS284	PLS283	PLS282	PLS281	PLS280	PLS279	PLS278	PLS277	PLS276	PLS275	PLS274	PLS273	PLS272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PLS271	PLS270	PLS269	PLS268	PLS267	PLS266	PLS265	PLS264	PLS263	PLS262	PLS261	PLS260	PLS259	PLS258	PLS257	PLS256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PLS[287:256]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W

- PLS9

Address(es): VIC.PLS9 A001 1104h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PLS300	PLS299	PLS298	PLS297	PLS296	PLS295	PLS294	PLS293	PLS292	PLS291	PLS290	PLS289	PLS288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b12 to b0	PLS[300:288]	Interrupt Input Detection Type Selection	0: Detects the level. 1: Detects the edge.	R/W
b31 to b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PLSi Bit (Interrupt Input Detection Type Selection) (i = 256 to 300)

This bit selects the interrupt input detection type.

PLS[300:256] corresponds to vector numbers 300 to 256.

12.4.2.6 Edge Detection Bit Clear Register n (PICn) (n = 0 to 9)

If you detect an edge, the interrupt detection status is retained for each interrupt input bit (See section 12.4, Cortex-R4 Vector Interrupt Controller (VIC) and, section 12.4.4.3, (3) IRQ Interrupt (Edge Interrupt)).
The PICn (n = 0 to 9) register clears the edge detection circuit for the interrupt input bit of which edge was detected to 0. This register can only be written in 32-bit units.

- PIC0

Address(es): VIC.PIC0 A001 0120h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC31	PIC30	PIC29	PIC28	PIC27	PIC26	PIC25	PIC24	PIC23	PIC22	PIC21	PIC20	PIC19	PIC18	PIC17	PIC16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC15	PIC14	PIC13	PIC12	PIC11	PIC10	PIC9	PIC8	PIC7	PIC6	PIC5	PIC4	PIC3	PIC2	PIC1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	The write value should be 0.	W
b31 to b1	PIC[31:1]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC1

Address(es): VIC.PIC1 A001 0124h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC63	PIC62	PIC61	PIC60	PIC59	PIC58	PIC57	PIC56	PIC55	PIC54	PIC53	PIC52	PIC51	PIC50	PIC49	PIC48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC47	PIC46	PIC45	PIC44	PIC43	PIC42	PIC41	PIC40	PIC39	PIC38	PIC37	PIC36	PIC35	PIC34	PIC33	PIC32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[63:32]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

PICi Bit (Edge Detection Clear) (i = 1 to 63)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

- PIC2

Address(es): VIC.PIC2 A001 0128h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC95	PIC94	PIC93	PIC92	PIC91	PIC90	PIC89	PIC88	PIC87	PIC86	PIC85	PIC84	PIC83	PIC82	PIC81	PIC80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC79	PIC78	PIC77	PIC76	PIC75	PIC74	PIC73	PIC72	PIC71	PIC70	PIC69	PIC68	PIC67	PIC66	PIC65	PIC64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[95:64]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC3

Address(es): VIC.PIC3 A001 012Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC127	PIC126	PIC125	PIC124	PIC123	PIC122	PIC121	PIC120	PIC119	PIC118	PIC117	PIC116	PIC115	PIC114	PIC113	PIC112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC111	PIC110	PIC109	PIC108	PIC107	PIC106	PIC105	PIC104	PIC103	PIC102	PIC101	PIC100	PIC99	PIC98	PIC97	PIC96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[127:96]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

PICi Bit (Edge Detection Clear) (i = 64 to 127)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

- PIC4

Address(es): VIC.PIC4 A001 0130h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC159	PIC158	PIC157	PIC156	PIC155	PIC154	PIC153	PIC152	PIC151	PIC150	PIC149	PIC148	PIC147	PIC146	PIC145	PIC144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC143	PIC142	PIC141	PIC140	PIC139	PIC138	PIC137	PIC136	PIC135	PIC134	PIC133	PIC132	PIC131	PIC130	PIC129	PIC128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[159:128]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC5

Address(es): VIC.PIC5 A001 0134h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC191	PIC190	PIC189	PIC188	PIC187	PIC186	PIC185	PIC184	PIC183	PIC182	PIC181	PIC180	PIC179	PIC178	PIC177	PIC176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC175	PIC174	PIC173	PIC172	PIC171	PIC170	PIC169	PIC168	PIC167	PIC166	PIC165	PIC164	PIC163	PIC162	PIC161	PIC160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[191:160]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

PICi Bit (Edge Detection Clear) (i = 128 to 191)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

- PIC6

Address(es): VIC.PIC6 A001 0138h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC223	PIC222	PIC221	PIC220	PIC219	PIC218	PIC217	PIC216	PIC215	PIC214	PIC213	PIC212	PIC211	PIC210	PIC209	PIC208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC207	PIC206	PIC205	PIC204	PIC203	PIC202	PIC201	PIC200	PIC199	PIC198	PIC197	PIC196	PIC195	PIC194	PIC193	PIC192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[223:192]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC7

Address(es): VIC.PIC7 A001 013Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC255	PIC254	PIC253	PIC252	PIC251	PIC250	PIC249	PIC248	PIC247	PIC246	PIC245	PIC244	PIC243	PIC242	PIC241	PIC240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC239	PIC238	PIC237	PIC236	PIC235	PIC234	PIC233	PIC232	PIC231	PIC230	PIC229	PIC228	PIC227	PIC226	PIC225	PIC224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[255:224]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

PICi Bit (Edge Detection Clear) (i = 192 to 255)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

- PIC8

Address(es): VIC.PIC8 A001 1120h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PIC287	PIC286	PIC285	PIC284	PIC283	PIC282	PIC281	PIC280	PIC279	PIC278	PIC277	PIC276	PIC275	PIC274	PIC273	PIC272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PIC271	PIC270	PIC269	PIC268	PIC267	PIC266	PIC265	PIC264	PIC263	PIC262	PIC261	PIC260	PIC259	PIC258	PIC257	PIC256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	PIC[287:256]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W

- PIC9

Address(es): VIC.PIC9 A001 1124h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PIC300	PIC299	PIC298	PIC297	PIC296	PIC295	PIC294	PIC293	PIC292	PIC291	PIC290	PIC289	PIC288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b12 to b0	PIC[300:288]	Edge Detection Clear	0: Nothing is changed. 1: Clears edge detection.	W
b31 to b13	—	Reserved	The write value should be 0.	W

PICi Bit (Edge Detection Clear) (i = 256 to 300)

For interrupt requests of which edges were detected, this bit clears the edge detection circuit for each interrupt request.

12.4.2.7 Interrupt Priority Level Mask Register 0 (PRLM0)

The PRLM0 register controls mask for interrupts to the interrupt priority level.
When a bit of this register is set to 1, it cannot be cleared to 0. To clear the bit to 0, use the interrupt priority level mask clear register 0 (PRLC0).
This register can only be read and written in 32-bit units.

Address(es): VIC.PRLM0 A001 01C0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRLM15	PRLM14	PRLM13	PRLM12	PRLM11	PRLM10	PRLM9	PRLM8	PRLM7	PRLM6	PRLM5	PRLM4	PRLM3	PRLM2	PRLM1	PRLM0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PRLM[15:0]	Interrupt Priority Level Setting	0: Nothing is changed. 1: Mask the same priority level as the corresponding bit number.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PRLMi Bit (Interrupt Priority Level Setting) (i = 0 to 15)

This is an interrupt priority level setting bit. For the sources with vector numbers 1 to 255, this bit sets mask of interrupts for the interrupt priority level.
The bit position of the register equals to the applicable priority level. When a bit is set to 1, the same priority level as the corresponding bit number is masked.

12.4.2.8 Interrupt Priority Level Mask Register 1 (PRLM1)

The PRLM1 register controls mask for interrupts to the interrupt priority level.
When a bit of this register is set to 1, it cannot be cleared to 0. To clear the bit to 0, use the interrupt priority level mask clear register 1 (PRLC1).
This register can only be read and written in 32-bit units.

Address(es): VIC.PRLM1 A001 11C0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRLM15	PRLM14	PRLM13	PRLM12	PRLM11	PRLM10	PRLM9	PRLM8	PRLM7	PRLM6	PRLM5	PRLM4	PRLM3	PRLM2	PRLM1	PRLM0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PRLM[15:0]	Interrupt Priority Level Setting	0: Nothing is changed. 1: Mask the same priority level as the corresponding bit number (PRLMi) + 16.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PRLMi Bit (Interrupt Priority Level Setting) (i = 0 to 15)

This is an interrupt priority level setting bit. For the sources with vector numbers 256 to 300, this bit sets mask of interrupts for the interrupt priority level.
When a bit is set to 1, the same priority level as the corresponding bit number (PRLMi) + 16 is masked.

12.4.2.9 Interrupt Priority Level Mask Clear Register 0 (PRLC0)

The PRLC0 register clears each bit of the PRLM0 register to 0.
This register can only be written in 32-bit units.

Address(es): VIC.PRLC0 A001 01C4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRLC 15	PRLC 14	PRLC 13	PRLC 12	PRLC 11	PRLC 10	PRLC 9	PRLC8	PRLC7	PRLC6	PRLC5	PRLC4	PRLC3	PRLC2	PRLC1	PRLC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PRLC[15:0]	Interrupt Priority Level Clear	0: Nothing is changed. 1: The bit corresponding to the bit number is cleared to 0.	W
b31 to b16	—	Reserved	The write value should be 0.	W

PRLCi Bit (Interrupt Priority Level Clear) (i = 0 to 15)

This is an interrupt priority level clear bit. For the sources with vector numbers 1 to 255, this bit clears interrupt priority level mask register 0 (PRLM0) to 0.
Once a bit is set to 1, the bit corresponding to the set bit number is cleared to 0.

12.4.2.10 Interrupt Priority Level Mask Clear Register 1 (PRLC1)

The PRLC1 register clears each bit of the PRLM1 register to 0.
The PRLC1 register can only be written in 32-bit units.

Address(es): VIC.PRLC1 A001 11C4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PRLC 15	PRLC 14	PRLC 13	PRLC 12	PRLC 11	PRLC 10	PRLC 9	PRLC8	PRLC7	PRLC6	PRLC5	PRLC4	PRLC3	PRLC2	PRLC1	PRLC0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	PRLC[15:0]	Interrupt Priority Level Clear	0: Nothing is changed. 1: The bit corresponding to the bit number is cleared to 0.	W
b31 to b16	—	Reserved	The write value should be 0.	W

PRLCi Bit (Interrupt Priority Level Clear) (i = 0 to 15)

This is an interrupt priority level clear bit. For the sources with vector numbers 256 to 300, this bit clears interrupt priority level mask register 1 (PRLM1) to 0.
Once the bit is set to 1, the bit corresponding to the set bit number is cleared to 0.

12.4.2.11 User Mode Enable Register 0 (UEN0)

This register is used to enable or disable access to the interrupt control register (VIC control registers except UEN0 and UEN1 registers) in privilege mode.
When a bus master cannot generate protection information correctly, set the UE bit to 1 to enable access to the interrupt control register in user mode. The initial value of the UE bit is 1 and access to the interrupt control register in user mode is enabled.
The register can be read in 32-bit units.
Unlike other registers, the register can be written in 32-bit units only in privilege mode.

Address VIC.UEN0 A001 01C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	UE*1	Interrupt control register access selection	0: Disables access in user mode. Enables access only in privilege mode. 1: Enables access in user mode. Access to interrupt control register in both user and privilege mode is possible (initial value).	W
b31 to b1	—	Reserved	The write value should be 0.	W

Note 1. Write access to the register is only possible in privilege mode.

UE Bit (Interrupt control register access selection)

This bit enables or disables access to the interrupt control register for sources corresponding to vector numbers 1 to 255.

12.4.2.12 User Mode Enable Register 1 (UEN1)

This register is used to enable or disable access to the interrupt control register (VIC control registers except UEN0 and UEN1 registers) in privilege mode.

When a bus master cannot generate protection information correctly, set the UE bit to 1 to enable access to the interrupt control register in user mode. The initial value of the UE bit is 1 and access to the interrupt control register in user mode is enabled.

The register can be read in 32-bit units.

Unlike other registers, the register can be written in 32-bit units only in privilege mode.

Address VIC.UEN1 A001 11C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	UE*1	Interrupt control register access selection	0: Disables access in user mode. Enables access only in privilege mode. 1: Enables access in user mode. Access to interrupt control register in both user and privilege mode is possible (initial value).	W
b31 to b1	—	Reserved	The write value should be 0.	W

Note 1. Write access to the register is only possible in privilege mode.

UE Bit (Interrupt control register access selection)

This bit enables or disables access to the interrupt control register for sources corresponding to vector numbers 256 to 300.

12.4.2.13 Interrupt Address Register (HVA0)

An arbitrary value must be written to the HVA0 register after being released from a reset in order to initialize the VIC. Also, an arbitrary value must be written to the HVA0 register at the end of an interrupt service routine (ISR). Writing to the HVA0 register causes the interrupt controller to recognize the completion of interrupt processing and clear the priority level of the stored interrupt. This leads to the processing of interrupts at the next priority level from that of the interrupt for which processing was just completed. The HVA0 register does not reflect values written to it.
Access to the register for any purpose other than initializing the VIC after it is released from a reset or ending interrupt processing is prohibited and attempting such access may result in incorrect interrupt operations.
This register can only be written in 32-bit units.

Address VIC.HVA0 A001 0200h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	HVA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	HVA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	HVA[31:0]	Interrupt processing ending notification	These bits notify an end of interrupt processing (by writing an arbitrary value)	W

12.4.2.14 Interrupt Service Status Register n (ISSn) (n = 0 to 9)

The ISSn (n = 0 to 9) register indicates the service status of an IRQ interrupt.
This register stores information for which Cortex-R4 is executing or suspending an interrupt service routine (ISR).
This register can only be read in 32-bit units.

- ISS0

Address(es): VIC.ISS0 A001 0210h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS31	ISS30	ISS29	ISS28	ISS27	ISS26	ISS25	ISS24	ISS23	ISS22	ISS21	ISS20	ISS19	ISS18	ISS17	ISS16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS15	ISS14	ISS13	ISS12	ISS11	ISS10	ISS9	ISS8	ISS7	ISS6	ISS5	ISS4	ISS3	ISS2	ISS1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R
b31 to b1	ISS[31:1]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS1

Address(es): VIC.ISS1 A001 0214h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS63	ISS62	ISS61	ISS60	ISS59	ISS58	ISS57	ISS56	ISS55	ISS54	ISS53	ISS52	ISS51	ISS50	ISS49	ISS48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS47	ISS46	ISS45	ISS44	ISS43	ISS42	ISS41	ISS40	ISS39	ISS38	ISS37	ISS36	ISS35	ISS34	ISS33	ISS32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[63:32]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

ISSi Bit (IRQ Interrupt Request Service) (i = 1 to 63)

This flag indicates the service status of an IRQ interrupt request from vector numbers 1 to 63.

- ISS2

Address(es): VIC.ISS2 A001 0218h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS95	ISS94	ISS93	ISS92	ISS91	ISS90	ISS89	ISS88	ISS87	ISS86	ISS85	ISS84	ISS83	ISS82	ISS81	ISS80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS79	ISS78	ISS77	ISS76	ISS75	ISS74	ISS73	ISS72	ISS71	ISS70	ISS69	ISS68	ISS67	ISS66	ISS65	ISS64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[95:64]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS3

Address(es): VIC.ISS3 A001 021Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS127	ISS126	ISS125	ISS124	ISS123	ISS122	ISS121	ISS120	ISS119	ISS118	ISS117	ISS116	ISS115	ISS114	ISS113	ISS112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS111	ISS110	ISS109	ISS108	ISS107	ISS106	ISS105	ISS104	ISS103	ISS102	ISS101	ISS100	ISS99	ISS98	ISS97	ISS96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[127:96]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

ISSi Bit (IRQ Interrupt Request Service Flag) (i = 64 to 127)

This flag indicates the service status of an IRQ interrupt request from vector numbers 127 to 64.

- ISS4

Address(es): VIC.ISS4 A001 0220h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS159	ISS158	ISS157	ISS156	ISS155	ISS154	ISS153	ISS152	ISS151	ISS150	ISS149	ISS148	ISS147	ISS146	ISS145	ISS144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS143	ISS142	ISS141	ISS140	ISS139	ISS138	ISS137	ISS136	ISS135	ISS134	ISS133	ISS132	ISS131	ISS130	ISS129	ISS128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[159:128]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS5

Address(es): VIC.ISS5 A001 0224h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS191	ISS190	ISS189	ISS188	ISS187	ISS186	ISS185	ISS184	ISS183	ISS182	ISS181	ISS180	ISS179	ISS178	ISS177	ISS176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS175	ISS174	ISS173	ISS172	ISS171	ISS170	ISS169	ISS168	ISS167	ISS166	ISS165	ISS164	ISS163	ISS162	ISS161	ISS160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[191:160]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

ISSi Bit (IRQ Interrupt Request Service Flag) (i = 128 to 191)

This flag indicates the service status of an IRQ interrupt request from vector numbers 191 to 128.

- ISS6

Address(es): VIC.ISS6 A001 0228h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS223	ISS222	ISS221	ISS220	ISS219	ISS218	ISS217	ISS216	ISS215	ISS214	ISS213	ISS212	ISS211	ISS210	ISS209	ISS208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS207	ISS206	ISS205	ISS204	ISS203	ISS202	ISS201	ISS200	ISS199	ISS198	ISS197	ISS196	ISS195	ISS194	ISS193	ISS192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[223:192]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS7

Address(es): VIC.ISS7 A001 022Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS255	ISS254	ISS253	ISS252	ISS251	ISS250	ISS249	ISS248	ISS247	ISS246	ISS245	ISS244	ISS243	ISS242	ISS241	ISS240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS239	ISS238	ISS237	ISS236	ISS235	ISS234	ISS233	ISS232	ISS231	ISS230	ISS229	ISS228	ISS227	ISS226	ISS225	ISS224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[255:224]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

ISSi Bit (IRQ Interrupt Request Service Flag) (i = 192 to 255)

This flag indicates the service status of IRQ interrupt request from vector numbers 255 to 192.

- ISS8

Address(es): VIC.ISS8 A001 1210h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISS287	ISS286	ISS285	ISS284	ISS283	ISS282	ISS281	ISS280	ISS279	ISS278	ISS277	ISS276	ISS275	ISS274	ISS273	ISS272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISS271	ISS270	ISS269	ISS268	ISS267	ISS266	ISS265	ISS264	ISS263	ISS262	ISS261	ISS260	ISS259	ISS258	ISS257	ISS256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISS[287:256]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R

- ISS9

Address(es): VIC.ISS9 A001 1214h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

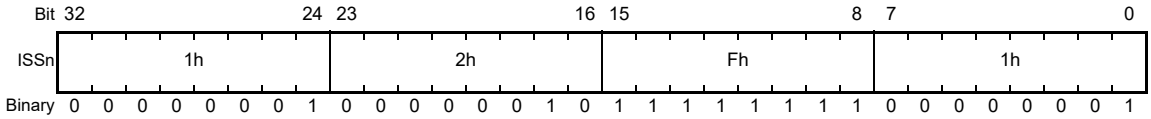
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	ISS300	ISS299	ISS298	ISS297	ISS296	ISS295	ISS294	ISS293	ISS292	ISS291	ISS290	ISS289	ISS288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b12 to b0	ISS[300:288]	IRQ Interrupt Request Service Flag	0: The service is not executed. 1: The interrupt service routine (ISR) is being executed or suspended.	R
b31 to b13	—	Reserved	These bits are read as 0.	R

ISSi Bit (IRQ Interrupt Request Service Flag) (i = 256 to 300)

This flag indicates the service status of IRQ interrupt request from vector numbers 300 to 256.

For example, if multiple interrupts are requested to the interrupt controller, the ISSn register shows the following status:



The figure indicates interrupts from vector numbers 24, 17, 15 to 8, and 0. Interrupt service routines (ISR) are serviced in descending order of priority which is set with the PRLm register. When the value of the PRLm is the same, the priority level of an interrupt with a smaller vector number is higher. When ISR finishes, the applicable bit of the ISSn register is cleared to 0, and then ISR that has the next highest priority level starts. In addition, if another interrupt is requested during ISR, the interrupt is also applied to this register.

12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9)

The ISCn (n = 0 to 9) register indicates the IRQ interrupt register with the highest priority level among IRQ interrupts which is set to 1 with interrupt service status register n (ISSn).
This register can only be read in 32-bit units.

- ISC0

Address(es): VIC.ISC0 A001 0230h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC31	ISC30	ISC29	ISC28	ISC27	ISC26	ISC25	ISC24	ISC23	ISC22	ISC21	ISC20	ISC19	ISC18	ISC17	ISC16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC15	ISC14	ISC13	ISC12	ISC11	ISC10	ISC9	ISC8	ISC7	ISC6	ISC5	ISC4	ISC3	ISC2	ISC1	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0.	R
b31 to b1	ISC[31:1]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC1

Address(es): VIC.ISC1 A001 0234h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC63	ISC62	ISC61	ISC60	ISC59	ISC58	ISC57	ISC56	ISC55	ISC54	ISC53	ISC52	ISC51	ISC50	ISC49	ISC48
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC47	ISC46	ISC45	ISC44	ISC43	ISC42	ISC41	ISC40	ISC39	ISC38	ISC37	ISC36	ISC35	ISC34	ISC33	ISC32
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[63:32]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 1 to 63)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

- ISC2

Address(es): VIC.ISC2 A001 0238h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC95	ISC94	ISC93	ISC92	ISC91	ISC90	ISC89	ISC88	ISC87	ISC86	ISC85	ISC84	ISC83	ISC82	ISC81	ISC80
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC79	ISC78	ISC77	ISC76	ISC75	ISC74	ISC73	ISC72	ISC71	ISC70	ISC69	ISC68	ISC67	ISC66	ISC65	ISC64
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[95:64]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC3

Address(es): VIC.ISC3 A001 023Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC127	ISC126	ISC125	ISC124	ISC123	ISC122	ISC121	ISC120	ISC119	ISC118	ISC117	ISC116	ISC115	ISC114	ISC113	ISC112
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC111	ISC110	ISC109	ISC108	ISC107	ISC106	ISC105	ISC104	ISC103	ISC102	ISC101	ISC100	ISC99	ISC98	ISC97	ISC96
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[127:96]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 64 to 127)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

- ISC4

Address(es): VIC.ISC4 A001 0240h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC159	ISC158	ISC157	ISC156	ISC155	ISC154	ISC153	ISC152	ISC151	ISC150	ISC149	ISC148	ISC147	ISC146	ISC145	ISC144
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC143	ISC142	ISC141	ISC140	ISC139	ISC138	ISC137	ISC136	ISC135	ISC134	ISC133	ISC132	ISC131	ISC130	ISC129	ISC128
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[159:128]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC5

Address(es): VIC.ISC5 A001 0244h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC191	ISC190	ISC189	ISC188	ISC187	ISC186	ISC185	ISC184	ISC183	ISC182	ISC181	ISC180	ISC179	ISC178	ISC177	ISC176
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC175	ISC174	ISC173	ISC172	ISC171	ISC170	ISC169	ISC168	ISC167	ISC166	ISC165	ISC164	ISC163	ISC162	ISC161	ISC160
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[191:160]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 128 to 191)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

- ISC6

Address(es): VIC.ISC6 A001 0248h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC223	ISC222	ISC221	ISC220	ISC219	ISC218	ISC217	ISC216	ISC215	ISC214	ISC213	ISC212	ISC211	ISC210	ISC209	ISC208
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC207	ISC206	ISC205	ISC204	ISC203	ISC202	ISC201	ISC200	ISC199	ISC198	ISC197	ISC196	ISC195	ISC194	ISC193	ISC192
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[223:192]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC7

Address(es): VIC.ISC7 A001 024Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC255	ISC254	ISC253	ISC252	ISC251	ISC250	ISC249	ISC248	ISC247	ISC246	ISC245	ISC244	ISC243	ISC242	ISC241	ISC240
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC239	ISC238	ISC237	ISC236	ISC235	ISC234	ISC233	ISC232	ISC231	ISC230	ISC229	ISC228	ISC227	ISC226	ISC225	ISC224
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[255:224]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

ISCi Bit (IRQ Interrupt Request Service Flag) (i = 192 to 255)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

- ISC8

Address(es): VIC.ISC8 A001 1230h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ISC287	ISC286	ISC285	ISC284	ISC283	ISC282	ISC281	ISC280	ISC279	ISC278	ISC277	ISC276	ISC275	ISC274	ISC273	ISC272
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ISC271	ISC270	ISC269	ISC268	ISC267	ISC266	ISC265	ISC264	ISC263	ISC262	ISC261	ISC260	ISC259	ISC258	ISC257	ISC256
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	ISC[287:256]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R

- ISC9

Address(es): VIC.ISC9 A001 1234h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	ISC300	ISC299	ISC298	ISC297	ISC296	ISC295	ISC294	ISC293	ISC292	ISC291	ISC290	ISC289	ISC288
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b12 to b0	ISC[300:288]	IRQ Interrupt Request Service Flag	0: The priority level is not the highest, or the interrupt service routine (ISR) is not executed. 1: An interrupt with the highest priority	R
b31 to b13	—	Reserved	These bits are read as 0.	R

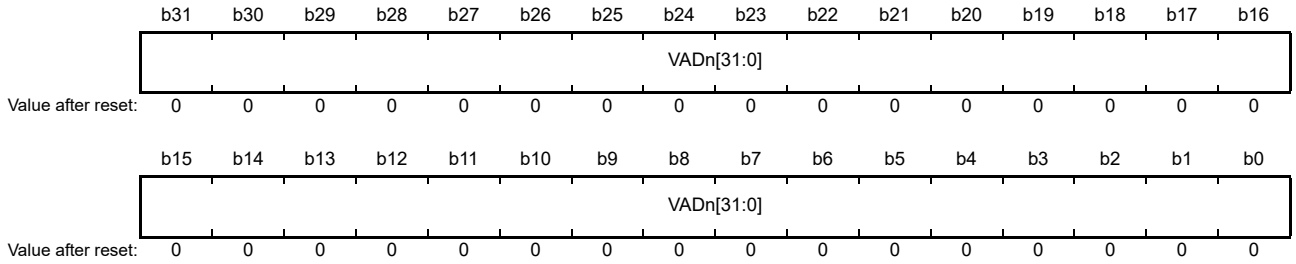
ISCi Bit (IRQ Interrupt Request Service Flag) (i = 256 to 300)

This flag indicates the service status of the IRQ interrupt request for interrupt service status register n (ISSn, n = 0 to 9).

12.4.2.16 Interrupt Address Store Register 0 (VADn) (n = 1 to 255)
Interrupt Address Store Register 1 (VADn) (n = 256 to 300)

The VADn (n = 1 to 300) register stores the vector address for each interrupt input.
This register can only be read and written in 32-bit units.

Address(es): VIC.VAD1 A001 0404h to VIC.VAD255 A001 07fch
VIC.VAD256 A001 1400h to VIC.VAD300 A001 14B0h



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	VADn[31:0]	Vector Address Store	VAD1 to VAD300 and vector numbers 1 to 300 are paired.	R/W

VADn Bit (Vector Address Store) (i = 1 to 300)

This is a vector address store bit. VAD1 to VAD300 and vector numbers 1 to 300 are paired.

- Connecting the CPU as the destination of interrupt requests
  Set the branch destination address of the interrupt handling routine to be run for interrupt requests n in this register.
- Connecting a DMAC as the destination of interrupt requests
  The occurrence of interrupt requests n that has been set up to do so starts the DMA transfer. Set the branch destination address of the interrupt handling routine to be run for the DMA transfer completion interrupt in this register. To connect a DMACAa, set vector number n in the associated register as described in section 14.2.8, DMACAa Unit 0 Source Select Register i (DMA0SELi) (i = 0 to 15), or section 14.2.9, DMACAa Unit 1 Source Select Register i (DMA1SELi) (i = 0 to 15).

12.4.2.17 Interrupt Priority Level Store Register 0 (PRLn) (n = 1 to 255)

The PRLn (n = 1 to 255) register stores the interrupt priority level for each interrupt input.
This register can only be read and written in 32-bit units.

Address(es): VIC.PRL1 A001 0804h to VIC.PRL255 A001 0BFCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	PRL3	PRL2	PRL1	PRL0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	PRL[3:0]	Interrupt Priority Level Store	The highest interrupt priority level is 0, and the lowest is 15.	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PRL[3:0] Bits (Interrupt Priority Level Store)

These bits store the interrupt priority level of vector numbers 1 to 255. The highest interrupt priority level is 0, and the lowest is 15.

12.4.2.18 Interrupt Priority Level Store Register 1 (PRLn) (n = 256 to 300)

The PRLn (n = 256 to 300) register stores the interrupt priority level for each interrupt input.
This register can only be read and written in 32-bit units.

Address(es): VIC.PRL256 A001 1800h to VIC.PRL300 A001 18B0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	PRL3	PRL2	PRL1	PRL0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	PRL[3:0]	Interrupt Priority Level Store	The highest interrupt priority level is the value of PRLn+16. The highest is 16, and the lowest is 31.	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PRL[3:0] Bits (Interrupt Priority Level Store)

These bits store the interrupt priority level of vector numbers 256 to 300. The highest interrupt priority level is the value of PRLn+16. The highest is 16, and the lowest is 31.

12.4.3 Vector Table

12.4.3.1 Interrupt Vector Table

Table 12.3 describes the vector table for interrupts to Cortex-R4 and DMACAa. Instead of an interrupt source from peripheral modules, transfer completion sources of DMACAa channels selected by the DMACAa source select register are connected to the vector number selected by the DMACAa source select register.

For the interrupt source for vector numbers 42 to 44, interrupt sources are different between Cortex-R4 and DMACAa. For Cortex-R4, the CPU interrupt source of the USB (func) is connected. Use vector numbers 43 and 44 as the DMA transfer sources for the USB (func). For details, see section 12.3.1, Selecting Interrupt Request Destinations. For details on interrupts from the USB, see section 23, USB2.0HS Host Module (USBh), and section 24, USB 2.0 HS Function Module (USBf).

The source of an interrupt for the Cortex-R4 and the same source as the trigger for DMACAa activation may be detected in different ways. The EtherCAT interrupt (vector number 75) is one example. For details, see section 14.3.4, DMA Transfer Request, and the notes on DMA transfer.

The following table explains the items of the Cortex-R4/DMACAa interrupt vector table.

Item	Description
Vector number	Indicates the vector number of the IRQ interrupt source in VIC for Cortex-R4.
Request source	Indicates the name of the interrupt request source.
Source	Indicates the interrupt name.
Detection type	Indicates the detection type for interrupts from peripheral modules. To connect a transfer completion interrupt from DMACAa, the edge must be selected.
CR4	“Y” indicates the interrupt source for Cortex-R4 (VIC).
DMACAa	“Y” indicates the DMACAa activation source.

Note: An error signal of each module is not input to the CPU directly, but to the error control module (ECM). The signal is merged into other errors and conveyed as an error detection source to the CPU. For details, see section 32, Error Control Module (ECM).

Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (1 / 7)

Vector Number	Request Source	Source		Detection Type	CR4	DMACAa *4
1	System (CR4)	—	Reserved	—	—	—
2		INTCTI	CTI (Cross Trigger Interface) interrupt	Edge	Y	N
3		FPUEX	FPU exception	Level	Y	N
4	External	IRQ0	IRQ pin interrupt 0	Level/Edge	Y	Y
5		IRQ1	IRQ pin interrupt 1	Level/Edge	Y	Y
6		IRQ2	IRQ pin interrupt 2	Level/Edge	Y	Y
7		IRQ3	IRQ pin interrupt 3	Level/Edge	Y	Y
8		IRQ4	IRQ pin interrupt 4	Level/Edge	Y	Y
9		—	Reserved	—	—	—
10		IRQ6	IRQ pin interrupt 6	Level/Edge	Y	Y
11		IRQ7	IRQ pin interrupt 7	Level/Edge	Y	Y
12		—	Reserved	—	—	—
13		IRQ9	IRQ pin interrupt 9	Level/Edge	Y	Y
14		—	Reserved	—	—	—
15		IRQ11	IRQ pin interrupt 11	Level/Edge	Y	Y
16		IRQ12	IRQ pin interrupt 12	Level/Edge	Y	Y
17		IRQ13	IRQ pin interrupt 13	Level/Edge	Y	Y
18		IRQ14	IRQ pin interrupt 14	Level/Edge	Y	Y
19		—	Reserved	—	—	—
20	ECM	ERRD	Error detection (maskable)	Edge	Y	N
21	CMT unit 0	CMI0	Compare match interrupt_ch0	Edge	Y	Y
22		CMI1	Compare match interrupt_ch1	Edge	Y	Y
23	CMT unit 1	CMI2	Compare match interrupt_ch0	Edge	Y	Y
24		CMI3	Compare match interrupt_ch1	Edge	Y	Y
25	CMTW unit 0	CMWI0	Compare match interrupt	Edge	Y	Y
26		IC0I0	Input capture 0 interrupt	Edge	Y	Y
27		IC1I0	Input capture 1 interrupt	Edge	Y	Y
28		OC0I0	Output compare 0 interrupt	Edge	Y	Y
29		OC1I0	Output compare 1 interrupt	Edge	Y	Y
30	CMTW unit 1	CMWI1	Compare match interrupt	Edge	Y	Y
31		IC0I1	Input capture 0 interrupt	Edge	Y	Y
32		IC1I1	Input capture 1 interrupt	Edge	Y	Y
33		OC0I1	Output compare 0 interrupt	Edge	Y	Y
34		OC1I1	Output compare 1 interrupt	Edge	Y	Y
35	—	—	Reserved	—	—	—
36		—	Reserved	—	—	—
37		—	Reserved	—	—	—
38		—	Reserved	—	—	—
39		—	Reserved	—	—	—
40		—	Reserved	—	—	—
41	USB	USBHI	USB (Host) CPU interrupt	Level	Y	N
42		USBFI	USB (Host) CPU interrupt	Level/Edge	Y	N
43		USBFDR1	USB (Func) DMA request 1	Level	N *3	Y
44		USBFDR2	USB (Func) DMA request 2	Level	N *3	Y

Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (2 / 7)

Vector Number	Request Source	Source		Detection Type	CR4	DMACAa *4
45	—	—	Reserved	—	—	—
46		—	Reserved	—	—	—
47		—	Reserved	—	—	—
48	Ethernet PHY	ETHPHYI0	Ethernet PHY interrupt 0	Level/Edge	Y	Y
49		ETHPHYI1	Ethernet PHY interrupt 1	Level/Edge	Y	Y
50	—	—	Reserved	—	—	—
51		—	Reserved	—	—	—
52		—	Reserved	—	—	—
53		—	Reserved	—	—	—
54		—	Reserved	—	—	—
55		—	Reserved	—	—	—
56		—	Reserved	—	—	—
57		—	Reserved	—	—	—
58		—	Reserved	—	—	—
59		—	Reserved	—	—	—
60		—	Reserved	—	—	—
61		—	Reserved	—	—	—
62		—	Reserved	—	—	—
63		—	Reserved	—	—	—
64		—	Reserved	—	—	—
65		—	Reserved	—	—	—
66		—	Reserved	—	—	—
67		—	Reserved	—	—	—
68		—	Reserved	—	—	—
69		—	Reserved	—	—	—
70		—	Reserved	—	—	—
71		—	Reserved	—	—	—
72		—	Reserved	—	—	—
73	ESC	ETHCSI0	EtherCAT Sync0 interrupt	Level/Edge*2	Y	Y
74		ETHCSI1	EtherCAT Sync1 interrupt	Level/Edge*2	Y	Y
75		ETHCI	EtherCAT interrupt	Level/Edge*1	Y	Y
76		ETHCSOFI	EtherCAT SOF interrupt	Edge	Y	Y
77		ETHCEOFI	EtherCAT EOF interrupt	Edge	Y	Y
78		ETHCWDTI	EtherCAT WDT interrupt	Edge	Y	N
79		ETHCRSTI	EtherCAT RESET interrupt	Edge	Y	N
80	RSPIa ch0	SPRI0	Reception buffer full	Edge	Y	Y
81		SPTI0	Transmission buffer empty	Edge	Y	Y
82		SPEI0	Mode fault error/overrun error/parity error	Level	Y	N
83		SPII0	RSPI idle	Level	Y	N
84	RSPIa ch1	SPRI1	Reception buffer full	Edge	Y	Y
85		SPTI1	Transmission buffer empty	Edge	Y	Y
86		SPEI1	Mode fault error/overrun error/parity error	Level	Y	N
87		SPII1	RSPI idle	Level	Y	N

Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (3 / 7)

Vector Number	Request Source	Source		Detection Type	CR4	DMACAa *4
88	—	—	Reserved	—	—	—
89		—	Reserved	—	—	—
90		—	Reserved	—	—	—
91		—	Reserved	—	—	—
92		—	Reserved	—	—	—
93		—	Reserved	—	—	—
94		—	Reserved	—	—	—
95		—	Reserved	—	—	—
96	SCIFA ch0	BRIF0	Break/overrun/framing error/parity error	Level	Y	N
97		RXIF0	Reception FIFO data full (RDF)	Level	Y	Y
98		TXIF0	Transmission FIFO data empty (TDFE)	Level	Y	Y
99		DRIF0	Transmit end/reception data ready	Level	Y	N
100	SCIFA ch1	BRIF1	Break/overrun/framing error/parity error	Level	Y	N
101		RXIF1	Reception FIFO data full (RDF)	Level	Y	Y
102		TXIF1	Transmission FIFO data empty (TDFE)	Level	Y	Y
103		DRIF1	Transmit end/reception data ready	Level	Y	N
104	RSCAN	CANRFI	CAN reception FIFO	Level	Y	N
105		—	Reserved	—	—	—
106		—	Reserved	—	—	—
107		CANFIR1	CAN1 transmission and reception FIFO transmission completed	Level	Y	N
108		CANTI1	CAN1 transmission	Level	Y	N
109	SCIFA ch2	BRIF2	Break/overrun/framing error/parity error	Level	Y	N
110		RXIF2	Reception FIFO data full (RDF)	Level	Y	Y
111		TXIF2	Transmission FIFO data empty (TDFE)	Level	Y	Y
112		DRIF2	Transmit end/reception data ready	Level	Y	N
113	SCIFA ch3	BRIF3	Break/overrun/framing error/parity error	Level	Y	N
114		RXIF3	Reception FIFO data full (RDF)	Level	Y	Y
115		TXIF3	Transmission FIFO data empty (TDFE)	Level	Y	Y
116		DRIF3	Transmit end/reception data ready	Level	Y	N
117	SCIFA ch4	BRIF4	Break/overrun/framing error/parity error	Level	Y	N
118		RXIF4	Reception FIFO data full (RDF)	Level	Y	Y
119		TXIF4	Transmission FIFO data empty (TDFE)	Level	Y	Y
120		DRIF4	Transmit end/reception data ready	Level	Y	N
121	—	—	Reserved	—	—	—
122		—	Reserved	—	—	—
123		—	Reserved	—	—	—
124	RIICa ch1	TEI1	Data transmission completed (TEND)	Level	Y	N
125		RXI1	Data reception completed (RDRF)	Edge	Y	Y
126		TXI1	Transmission data empty (TDRE)	Edge	Y	Y
127	—	—	Reserved	—	—	—
128		—	Reserved	—	—	—
129		—	Reserved	—	—	—
130		—	Reserved	—	—	—
131		—	Reserved	—	—	—

Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (4 / 7)

Vector Number	Request Source	Source		Detection Type	CR4	DMACAa *4
132	—	—	Reserved	—	—	—
133		—	Reserved	—	—	—
134		—	Reserved	—	—	—
135		—	Reserved	—	—	—
136		—	Reserved	—	—	—
137		—	Reserved	—	—	—
138		—	Reserved	—	—	—
139		—	Reserved	—	—	—
140		—	Reserved	—	—	—
141		—	Reserved	—	—	—
142		—	Reserved	—	—	—
143		—	Reserved	—	—	—
144		—	Reserved	—	—	—
145		—	Reserved	—	—	—
146		—	Reserved	—	—	—
147		—	Reserved	—	—	—
148		—	Reserved	—	—	—
149		—	Reserved	—	—	—
150		—	Reserved	—	—	—
151		—	Reserved	—	—	—
152		—	Reserved	—	—	—
153		—	Reserved	—	—	—
154		—	Reserved	—	—	—
155		—	Reserved	—	—	—
156		—	Reserved	—	—	—
157		—	Reserved	—	—	—
158		—	Reserved	—	—	—
159		—	Reserved	—	—	—
160		—	Reserved	—	—	—
161		—	Reserved	—	—	—
162		—	Reserved	—	—	—
163		—	Reserved	—	—	—
164		—	Reserved	—	—	—
165		—	Reserved	—	—	—
166		—	Reserved	—	—	—
167		—	Reserved	—	—	—
168		—	Reserved	—	—	—
169		—	Reserved	—	—	—
170		—	Reserved	—	—	—
171		—	Reserved	—	—	—
172		—	Reserved	—	—	—
173		—	Reserved	—	—	—

Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (5 / 7)

Vector Number	Request Source	Source		Detection Type	CR4	DMACAa *4
174	—	—	Reserved	—	—	—
175		—	Reserved	—	—	—
176		—	Reserved	—	—	—
177		—	Reserved	—	—	—
178		—	Reserved	—	—	—
179		—	Reserved	—	—	—
180		—	Reserved	—	—	—
181		—	Reserved	—	—	—
182		—	Reserved	—	—	—
183		—	Reserved	—	—	—
184		—	Reserved	—	—	—
185		—	Reserved	—	—	—
186		—	Reserved	—	—	—
187		—	Reserved	—	—	—
188		—	Reserved	—	—	—
189		—	Reserved	—	—	—
190		—	Reserved	—	—	—
191		—	Reserved	—	—	—
192		—	Reserved	—	—	—
193		—	Reserved	—	—	—
194		—	Reserved	—	—	—
195		—	Reserved	—	—	—
196		—	Reserved	—	—	—
197		—	Reserved	—	—	—
198		—	Reserved	—	—	—
199		—	Reserved	—	—	—
200		—	Reserved	—	—	—
201		—	Reserved	—	—	—
202		—	Reserved	—	—	—
203		—	Reserved	—	—	—
204		—	Reserved	—	—	—
205		—	Reserved	—	—	—
206		—	Reserved	—	—	—
207		—	Reserved	—	—	—
208		—	Reserved	—	—	—
209		—	Reserved	—	—	—
210		—	Reserved	—	—	—
211		—	Reserved	—	—	—
212		—	Reserved	—	—	—
213		—	Reserved	—	—	—
214		—	Reserved	—	—	—
215		—	Reserved	—	—	—

Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (6 / 7)

Vector Number	Request Source	Source		Detection Type	CR4	DMACAa *4
216	—	—	Reserved	—	—	—
217		—	Reserved	—	—	—
218		—	Reserved	—	—	—
219		—	Reserved	—	—	—
220		—	Reserved	—	—	—
221		—	Reserved	—	—	—
222		—	Reserved	—	—	—
223		—	Reserved	—	—	—
224		—	Reserved	—	—	—
225		—	Reserved	—	—	—
226		—	Reserved	—	—	—
227		—	Reserved	—	—	—
228		—	Reserved	—	—	—
229		—	Reserved	—	—	—
230		—	Reserved	—	—	—
231		—	Reserved	—	—	—
232		—	Reserved	—	—	—
233		—	Reserved	—	—	—
234		—	Reserved	—	—	—
235		—	Reserved	—	—	—
236		—	Reserved	—	—	—
237		—	Reserved	—	—	—
238		—	Reserved	—	—	—
239		—	Reserved	—	—	—
240		—	Reserved	—	—	—
241		—	Reserved	—	—	—
242	ELC	ELCIRQ1	Interrupt 1 (ELSR18)	Edge	Y	Y
243		ELCIRQ2	Interrupt 2 (ELSR19)	Edge	Y	Y
244	—	—	Reserved	—	—	—
245		—	Reserved	—	—	—
246		—	Reserved	—	—	—
247		—	Reserved	—	—	—
248		—	Reserved	—	—	—
249		—	Reserved	—	—	—
250		—	Reserved	—	—	—
251	DMACAa	DMASRQ0	DMA software activation (unit 0)	Edge	N*3	Y
252		DMASRQ1	DMA software activation (unit 1)	Edge	N*3	Y
253	—	—	Reserved	—	—	—
254		—	Reserved	—	—	—
255		—	Reserved	—	—	—
256		—	Reserved	—	—	—
257		—	Reserved	—	—	—
258		—	Reserved	—	—	—
259		—	Reserved	—	—	—
260		—	Reserved	—	—	—

Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (7 / 7)

Vector Number	Request Source	Source		Detection Type	CR4	DMACAa *4
261	RIICa ch1	EEI1	Stop condition detection/Start condition detection/NACK detection/arbitration lost/time-out occurrence	Level	Y	N
262	RSCAN	CANGE	CAN global error	Level	Y	N
263		—	Reserved	—	—	—
264		CANIE1	CAN1 error	Level	Y	N
265	—	—	Reserved	—	—	—
266		—	Reserved	—	—	—
267		—	Reserved	—	—	—
268		—	Reserved	—	—	—
269		—	Reserved	—	—	—
270		—	Reserved	—	—	—
271		—	Reserved	—	—	—
272		—	Reserved	—	—	—
273		—	Reserved	—	—	—
274		—	Reserved	—	—	—
275		—	Reserved	—	—	—
276		—	Reserved	—	—	—
277		—	Reserved	—	—	—
278		—	Reserved	—	—	—
279		—	Reserved	—	—	—
280		—	Reserved	—	—	—
281		—	Reserved	—	—	—
282		—	Reserved	—	—	—
283		—	Reserved	—	—	—
284		—	Reserved	—	—	—
285		—	Reserved	—	—	—
286		—	Reserved	—	—	—
287		—	Reserved	—	—	—
288		—	Reserved	—	—	—
289		—	Reserved	—	—	—
290		—	Reserved	—	—	—
291		—	Reserved	—	—	—
292		—	Reserved	—	—	—
293	DMACAa	DMAERR0	DMA transfer transfer error (unit 0)	Edge	Y	N
294		DMAERR1	DMA transfer transfer error (unit 1)	Edge	Y	N
295	—	—	Reserved	—	—	—
296		—	Reserved	—	—	—
297		—	Reserved	—	—	—
298		—	Reserved	—	—	—
299	CMT unit 2	CMI4	Compare match interrupt_ch0	Edge	Y	N
300		CMI5	Compare match interrupt_ch1	Edge	Y	N

Note: Do not select the interrupt request destinations that do not have Y for the request destination.
Note 1. Select level detection when using this source to generate interrupts for the CPU and edge detection when using it as a trigger for DMA activation.
Note 2. Select the level or edge detection when using this source to generate interrupts for the CPU, and edge detection when using it as a trigger for DMA activation.

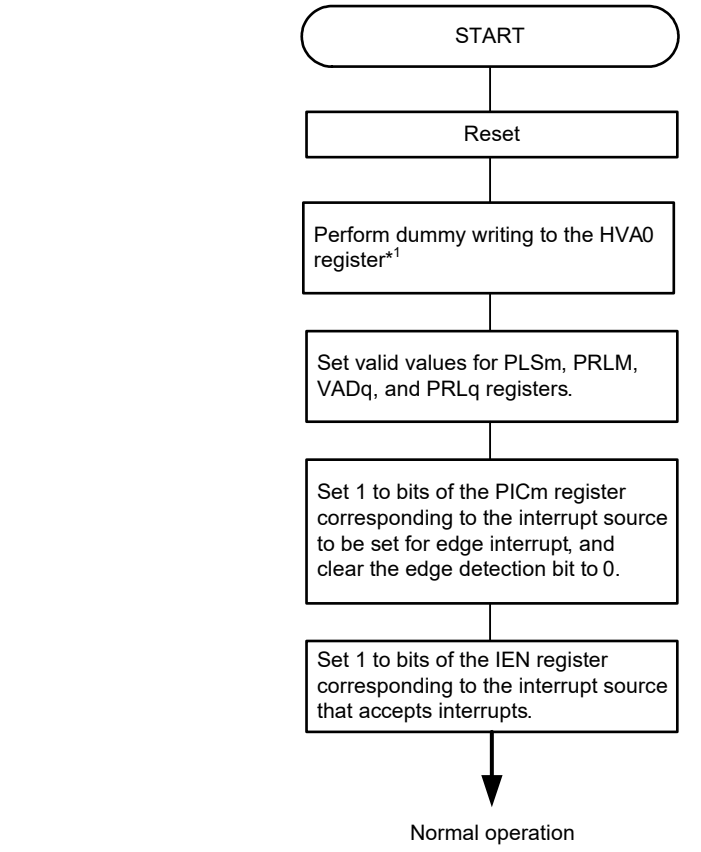
Note 3. Though an interrupt servings as a DMA activating source is not generated as an interrupt, the interrupt handling routine with the vector number of the DMA transfer completion interrupt is executed on completion of DMA transfer. For details, see section 12.3.1, Selecting Interrupt Request Destinations, and section 14.4.2, DMA Transfer Completion Interrupts.

Note 4. When connecting the DMACa transfer completion interrupt, always select the same edge as that selected by the setting of interrupt detection type select register n (PLSn). For details, see section 12.4.4.3, (1) Specifying Interrupt Detection Types.

12.4.4 Operation

12.4.4.1 Initializing Registers of VIC

Figure 12.5 shows the procedure for initializing registers of VIC.
When you canceled reset, registers are not ready to operate due to interrupt priority level settings of VIC, or other reasons. For this reason, you must initialize the registers after a reset release.



m = 0 to 9, q = 1 to 300
Note 1. For initializing the VIC, write this register once after being released from a reset. Also, write this register at the end of each interrupt handling routine. For details, see section 12.4.2.13, Interrupt Address Register (HVA0).

Figure 12.5 Initializing Registers of VIC

12.4.4.2 Procedure for Rewriting the PLS, PRLM, VAD, and PRL Registers

To rewrite the following registers while VIC is in operation, finish all interrupt processing, and then disable interrupts. To disable interrupts, set 1 to the I bit of the CPSR register for Cortex-R4.

- PLS (Interrupt detection type select register)
- PRLM (Interrupt priority level mask register)
- VAD (Interrupt address storage register)
- PRL (Interrupt priority level storage register)

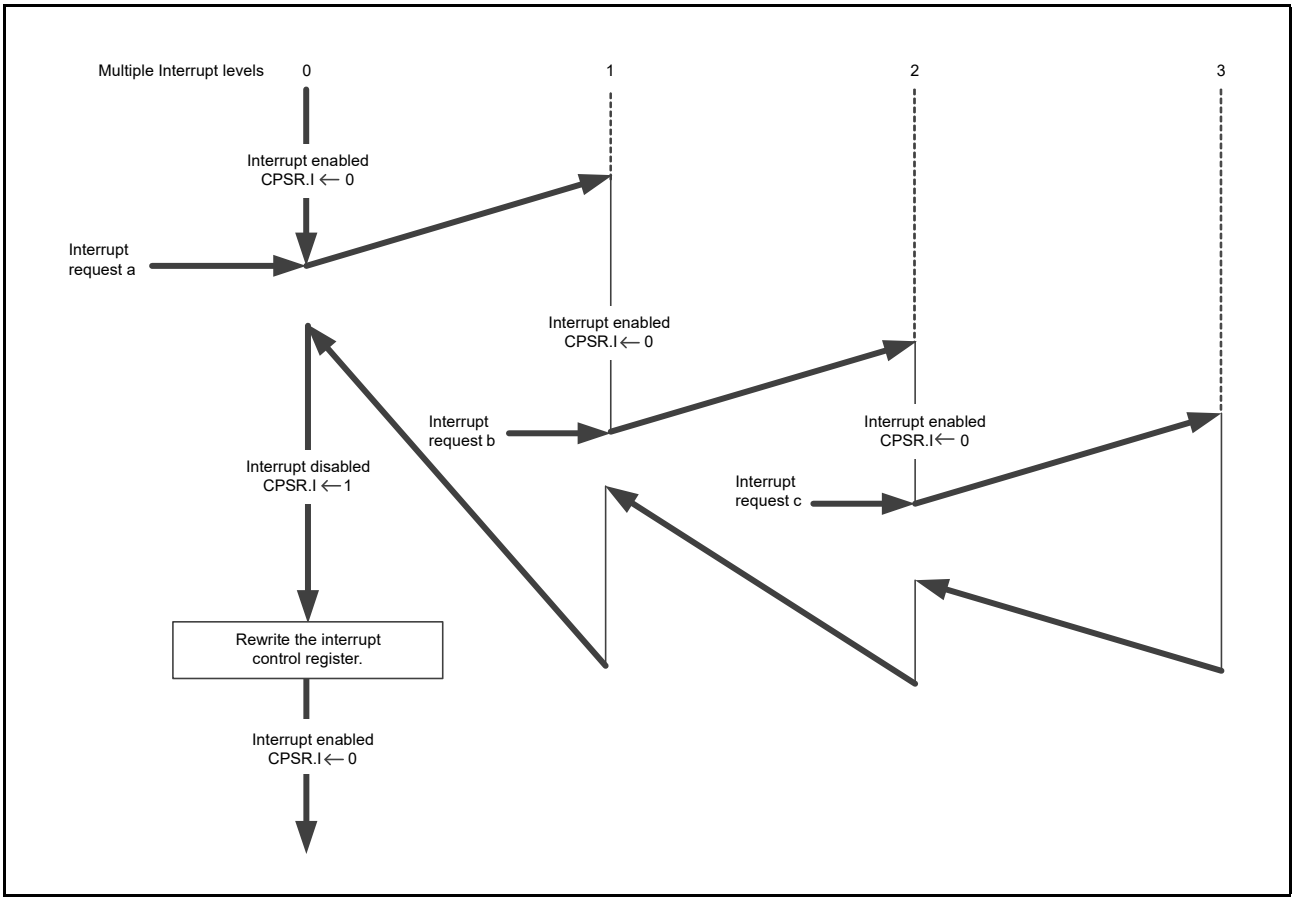


Figure 12.6 Period for Changing Register Settings

Follow the flow below to change register settings by using software.

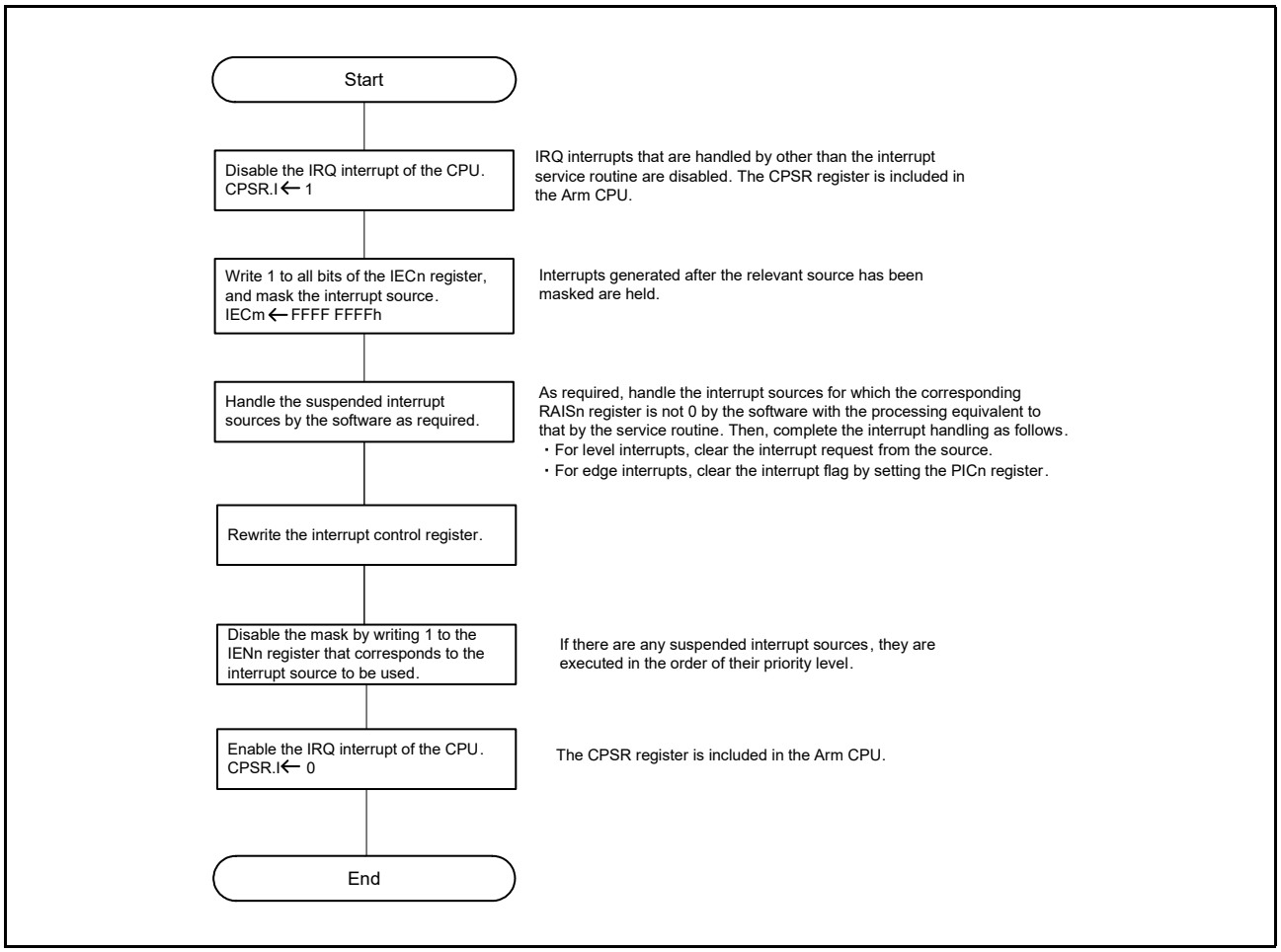


Figure 12.7 Register Rewrite Flow

12.4.4.3 Detecting Interrupts

(1) Specifying Interrupt Detection Types

When connecting external interrupts except interrupts from the NMI pin, and interrupts from on-chip peripheral modules to Cortex-R4, you must use VIC to select the edge or level detection with interrupt detection type select register n (PLSn). Table 12.4 lists settings of the interrupt detection type for VIC. When connecting transfer completion interrupt from DMACAa, always select the edge.

Table 12.4 VIC Settings by Interrupt Detection Type

Interrupt Requests Type	PLSm
Edge interrupt	1
Level interrupt	0

m: Interrupt vector number (0 to 300)

(2) IRQ Interrupt (Level interrupt)

Figure 12.8 shows level interrupt operation.

When you complete a level interrupt, stop interrupt output of the request source for the level interrupt. At that time, the applicable bit of IRQ status register n (IRQSn, n = 0 to 9) is cleared to 0. After that, make sure the interrupt is no longer requested. This operation is needed to prevent the same interrupt from being accepted after restoration because there is a delay before the interrupt output stop processing of the interrupt request source by the software is applied to the hardware. In addition, stop the interrupt output of the interrupt request source at the appropriate position of the service routine (ISR) according to the operation of the request source.

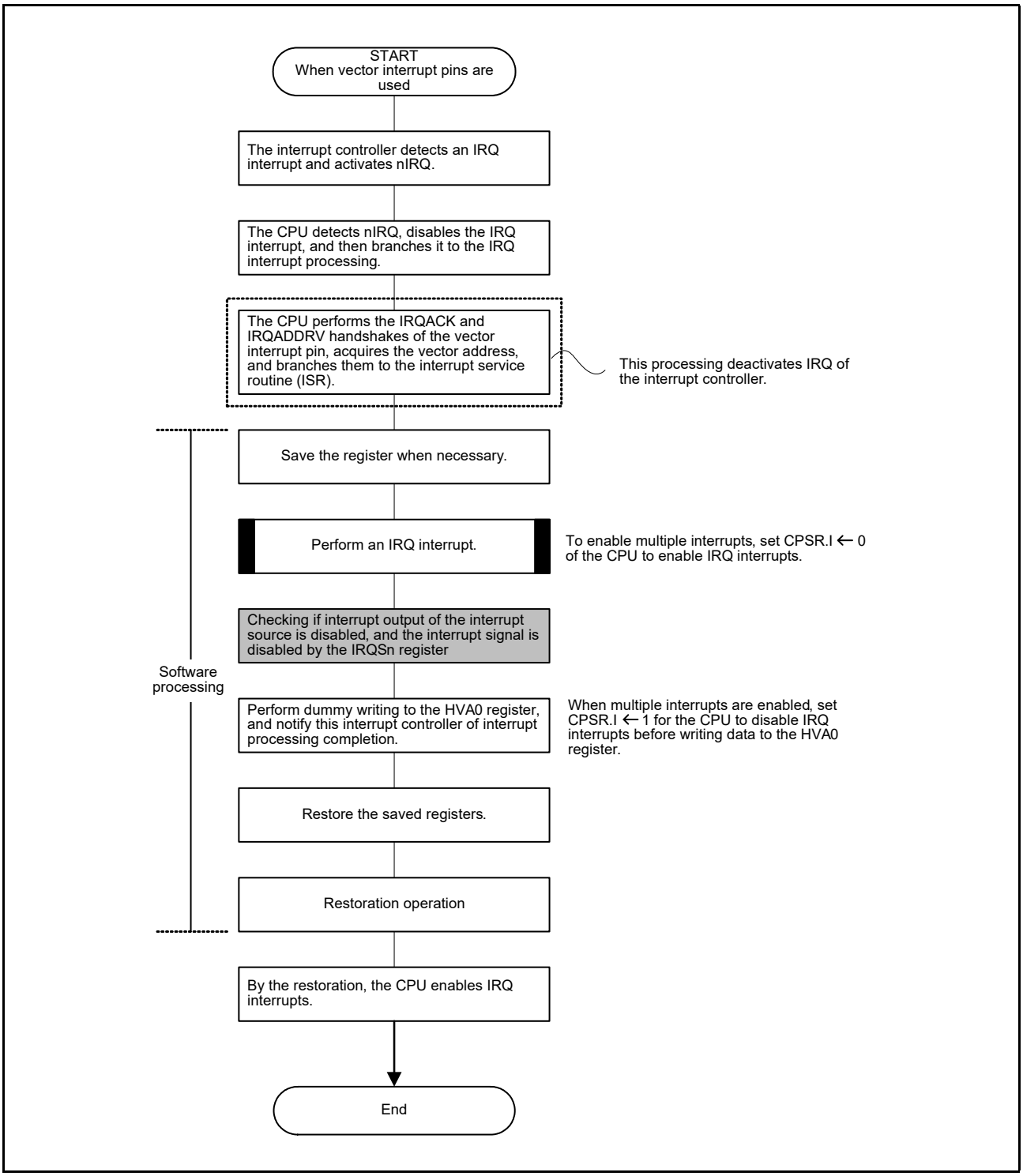


Figure 12.8 IRQ Interrupt Operation (Level Operations)

(3) IRQ Interrupt (Edge Interrupt)

Figure 12.9 shows edge interrupt operation.

Clear the edge interrupt request with edge detection bit clear register n (PICn, n = 0 to 9).

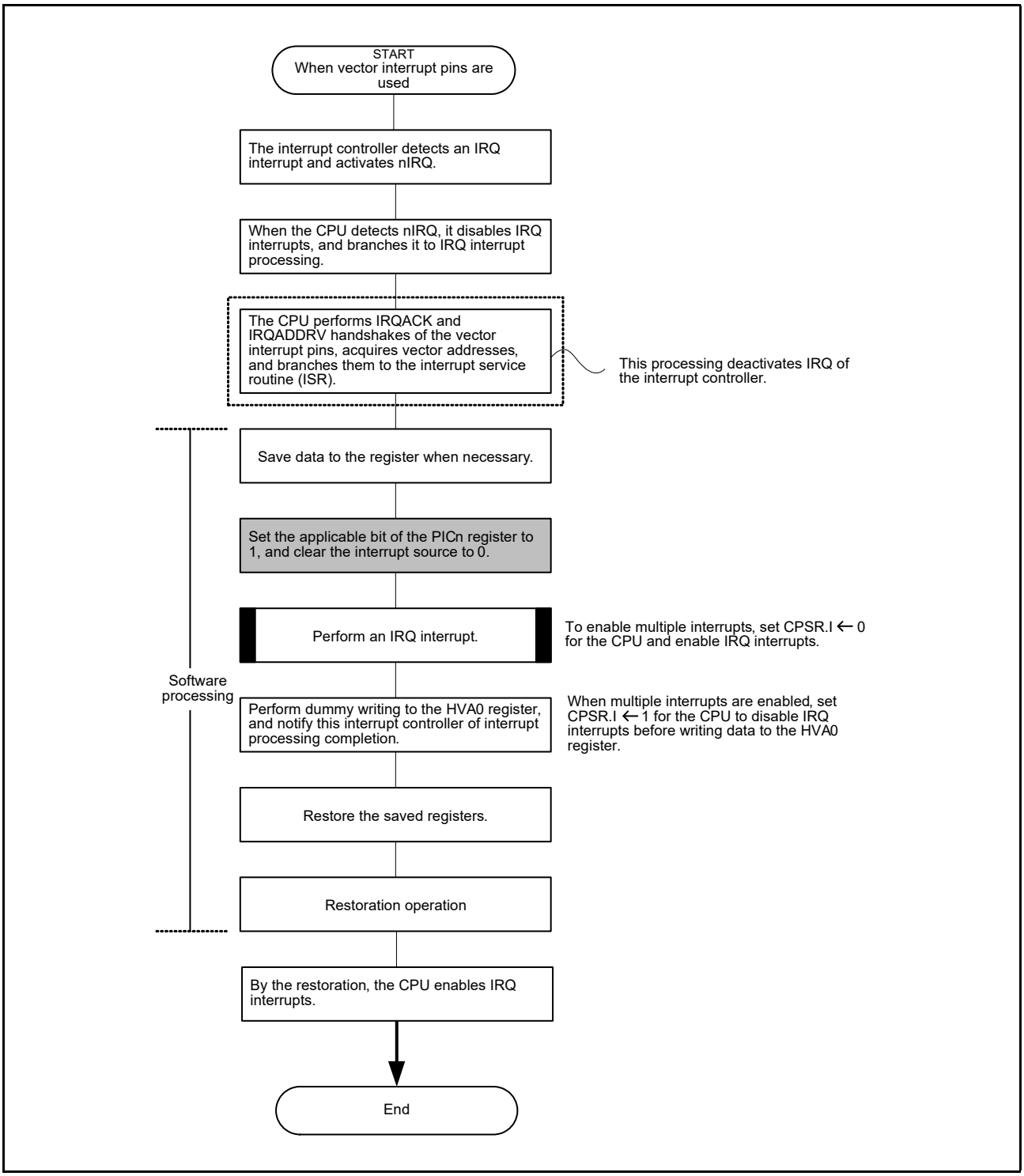


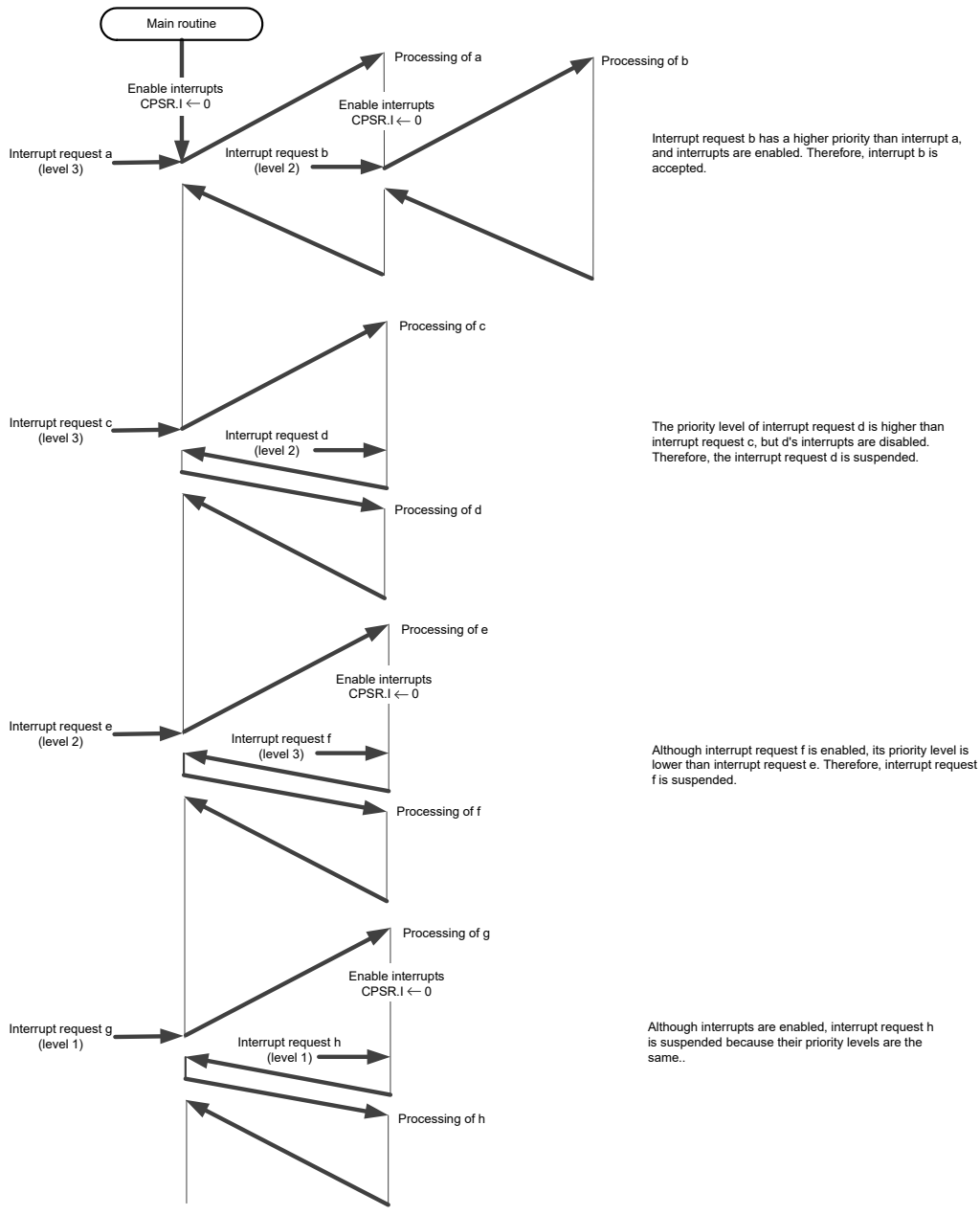
Figure 12.9 IRQ Interrupt Operation (Edge Interrupt)

12.4.4.4 Priority Level for Interrupt Multiple Control

If an interrupt is being handled (only when interrupt multiple control is being performed), only an interrupt that has higher priority than the interrupt which is being serviced is accepted. At that time, interrupts with lower priority than the currently serviced interrupt are suspended.

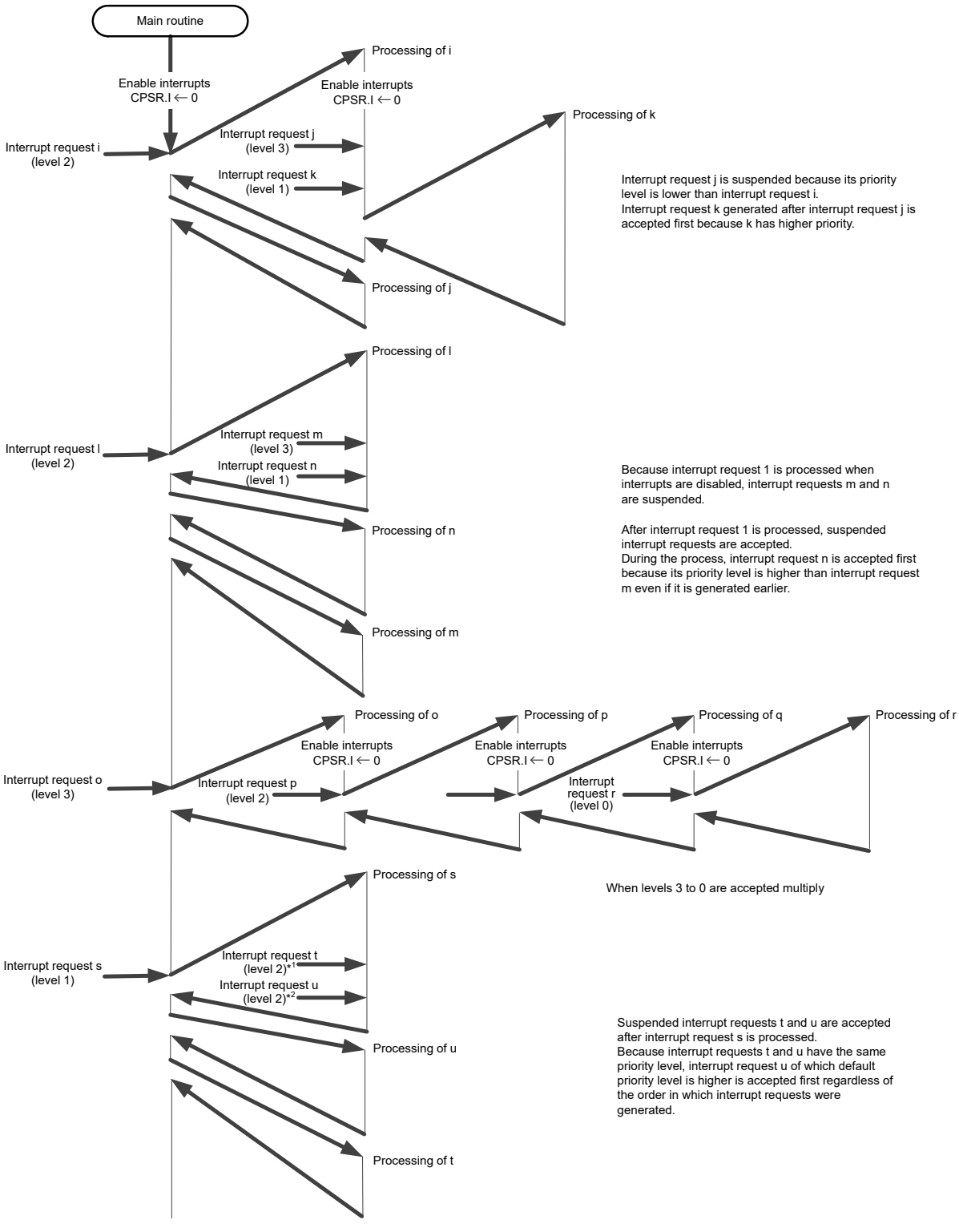
12.4.4.5 Handling Multiple Interrupts

Figure 12.10 provides an example of multiple interrupts that accept another interrupt when an interrupt is being handled.



a through u in the figure (including 2/2) are tentative names used to identify each interrupt request signal.
High/Low of the default priority level in the figure indicate the relative priority level between two interrupt request signals.
The priorities are higher for smaller vector numbers, which means, if the same value is set in two interrupts by the PRLm register, the one with smaller vector number takes priority.

Figure 12.10 Concept of Multiple Interrupts (1 / 2)



Note 1. The default priority level is low.
Note 2. The default priority level is high.
The priorities are higher for smaller vector numbers, which means, if the same value is set in two interrupts by the PRLm register, the one with smaller vector number takes priority.

Figure 12.10 Concept of Multiple Interrupts (2 / 2)

12.4.4.6 Handling IRQ Interrupt Source Conditions by Polling

Figure 12.11 shows the procedure for handling IRQ interrupt source conditions by polling the interrupt status registers (RAISn).

That is, source conditions for IRQ interrupts can be detected by checking the bits of the interrupt input status registers (RAISn). This is useful when interrupts are masked by settings in the interrupt enable registers (IENn). This enables interrupt processing without hardware forcing branches to the interrupt service routines (ISRs).

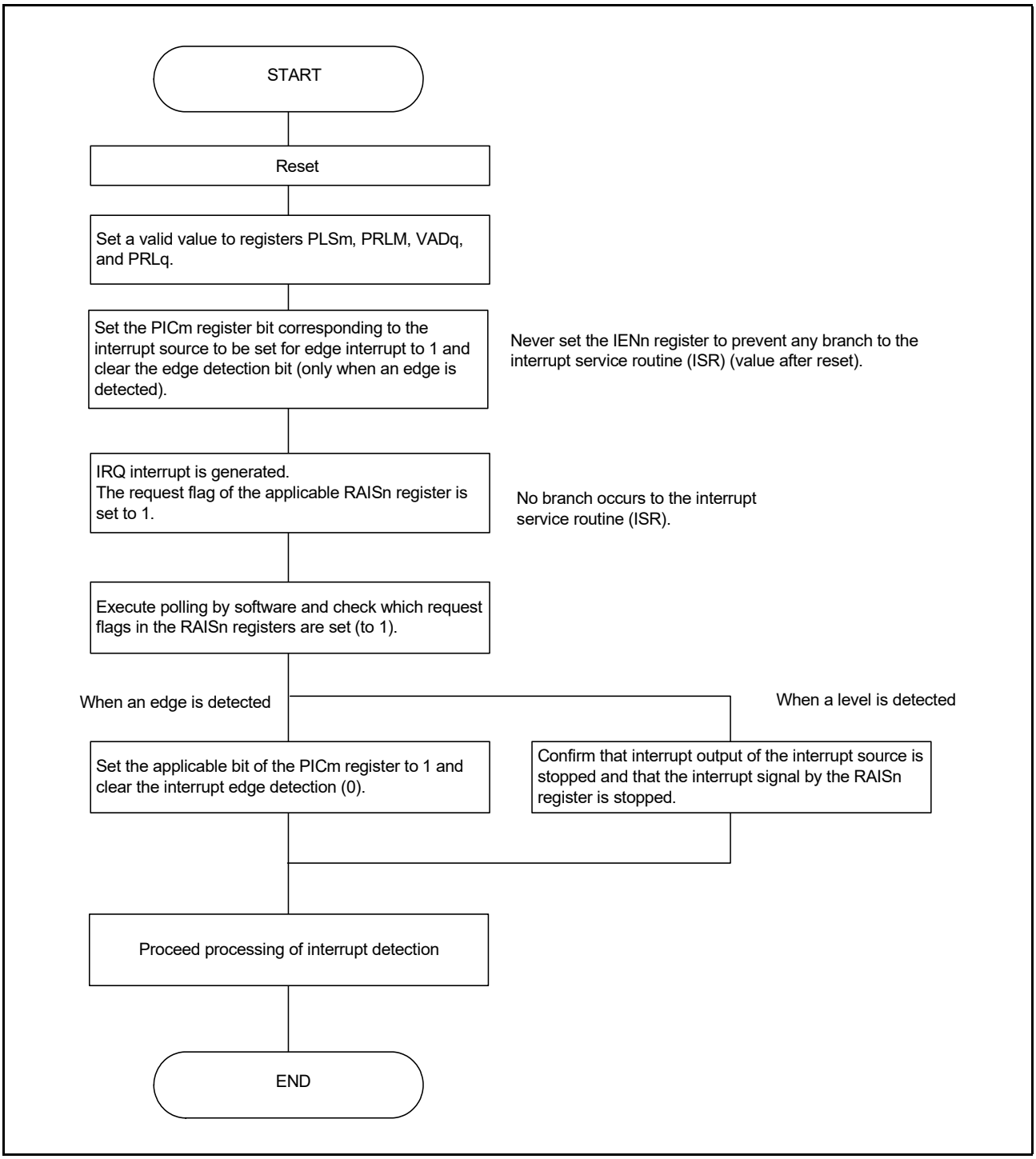


Figure 12.11 IRQ Interrupt Operation by Polling (Edge/Level Detection)

12.4.5 Usage Notes

12.4.5.1 Restrictions on VIC Priority Levels

VIC specifies the priority in 16 levels for each source with the interrupt priority level store register n (PRLn, n = 1 to 300). But lower priority levels are assigned to sources with vector numbers 256 and later compared to sources with vector numbers 1 to 255. Table 12.5 lists the relationship between vector numbers and priority levels.

Table 12.5 Relationship between Vector Numbers and Priority Levels

Vector Number	Priority Level*1
1 to 255	PRLn
256 to 300	PRLn + 16

n = 1 to 300
Note 1. The highest priority level i 0.

12.4.5.2 Notes on Accessing HVA0 Register

In cases of contention due to a vector interrupt being generated at the same time as writing to the HVA0 register, the AHB allows dummy-writing to the HVA0 register to proceed and blocks conveying of the IRQ interrupt to the CPU until a response to the write operation is returned so that processing to handle a new vector interrupt remains inactive over that time.
When writing to the HVA0 register, never fail to wait for the response indicating completion of the write operation before allowing IRQ interrupts to be sent to the CPU. Successful writing will be ensured by following the procedure illustrated in Figure 12.8 and executing a DMB instruction immediately after the write to the HVA0 register (as in the program example below).

- Program example

```
VIC.HVA0.LONG = 0x00000000;
asm("dmb");                    //DMB instruction
```

Note: The program format may vary according to the compiler. Confirm the applicable format for each compiler in the individual manuals.

12.4.5.3 Notes on Selecting Level Detection

When an interrupt request is set to level detection, do not cancel an interrupt request that was already generated except interrupt cancellation by the CPU. The correct vector address might not be output. For example, if level interrupt A is generated, the source for the output of interrupt A must be cleared by the service routine for interrupt A in a normal situation as shown in Figure 12.8 of section 12.4.4.3, Detecting Interrupts, but the source for the output of interrupt A may be transiently cleared if the service routine for another interrupt B has cleared or masked the source for the output of interrupt A.
When this interrupt controller accepts an interrupt, and the interrupt request is canceled before the CPU acquires the vector address, if another interrupt request is generated at the same time, 0000 0014h is output as the vector address. Therefore, to handle transitional cancellation of interrupt requests as mentioned above, it is recommended to use the return instruction only for 0000 0014h of the CPU (refer to the Example Program below).
In addition, if an interrupt request signal is transitionally withheld during the processing of an interrupt for which multiple interrupts are enabled, branching returns to the interrupt processing for which service is currently in progress. This interrupt handler writing to the HVA0 register causes the controller for this interrupt to recognize the completion of interrupt processing. Note that there is a gap between the actual interrupt source generating an interrupt request and the CPU recognizing the interrupt source.

- Example Program
  reserved_handler:
  subs pc, lr, #4 ; locate at 0000 0014h.

Note: The program format may differ for each complier. Confirm the applicable format for each in their manuals.

12.4.5.4 Notes when Rewriting the IECn Register

If the value of the IECn register is to be changed, do so while interrupts are disabled. To disable interrupts, set the I bit in the CPSR register of the Arm CPU to 1.

12.4.5.5 Notes on Vector Settings

In the specification of this product, use of the fixed vector by setting the SCTLR.[24]VE bit to 0 is prohibited. Only providing addresses from those of the VIC by setting SCTLR.[24]VE bit = 1 is possible. The addresses can be set by using the VADn registers (n: vector number).

12.5 Usage Note

12.5.1 Using “Falling-Edge” or “Rising and Falling Edges” Detection with the External Pin Interrupts

Since the internal level on external pin interrupts after a reset is high, when external pin interrupts are used with low as the initial input level and the detection of “alling edge” or “ising and falling edges” follow the procedure below. Otherwise, follow the procedures shown in section 12.3.3, External Pin Interrupts.
In addition, make sure that a falling edge is not input to the external pin interrupts before these settings are completed.

[For IRQ pins]

1. Clear the applicable IENn bit to 0 (set the IECn bit).
2. Set the Pmn I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).
3. Set and check (read) the I/O port (PmnPFS.ISEL bit).
4. Clear the IRQFLTE.FLTENi bit to 0.*1
5. Set the digital noise filter sampling clock with the IRQFLTC.FCLKSEL[1:0] bits.*1
6. Set the IRQFLTE.FLTENi bit to 1.*1
7. Select the edge for detection as “falling” or “rising and falling” by setting the IRQCRi.IRQMD[1:0] bits.
8. Set the applicable PICn register to 1 (when an edge is detected).
9. Set the applicable IENn bit to 1.

Note 1. Setting is required only when the digital filter is used.

[For ETH0_INT/ETH1_INT pins]

1. Set the applicable IENn bit to 0 (set the IECn bit).
2. Set the Pmn I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).
3. Set the I/O port (the PmnPFS.PSEL[5:0] bits and PMR register), and check (read) the PmnPFS register.
4. Clear the EPHYFLTE.EFLTENi bit to 0.*1
5. Set the digital noise filter sampling clock with the EPHYFLTC.EFCLKSEL[1:0] bits.*1
6. Set the EPHYFLTE.EFLTENi bit to 1.*1
7. Select the edge for detection as “falling” or “rising and falling” by setting the EPHYCRi.EPHYMD[1:0] bits.
8. Set the applicable PICn register to 1 (when an edge is detected).
9. Set the applicable IENn bit to 1.

Note 1. Setting is required only when the digital filter is used.

12.5.2 Using Falling-Edge Detection with the NMI Pin

Since the internal level on the NMI pin after a reset is high, when the NMI pin is used with low as the initial input level and the detection of falling edges, follow the procedure below.
Otherwise, follow the procedure shown in section 12.3.4, NMI Pin Interrupts.
In addition, make sure that a falling edge is not input to the NMI pin before these settings are completed.

1. Select the edge for detection as rising by setting the NMICR.NMIMD bit to 1.
2. Set and check the I/O port pin (P35PFS.ISEL bit).
3. Clear the NMIFLTE.NFLTEN bit to 0.*1
4. Set the sampling clock cycle for the digital noise filter in the NMIFLTC.NFCLKSEL[1:0] bits.*1
5. Clear the NMICR.NMIMD bit to 0 (falling edge detection).
6. Set the NMICLR.NMICLR bit to 1, and clear the NMISR.NMIST flag to 0.
7. Set the NMIFLTE.NFLTEN bit to 1.*1
8. Set the P35 I/O select bit in the port direction register (PDR) of the I/O port to 10b (input).

Note 1. This setting is only required when the digital noise filter is to be used.

13. Internal Buses

13.1 Overview

This product contains two internal main buses, a memory buses, and multiple internal peripheral buses.
Table 13.1 lists the specifications of internal buses and Figure 13.1 shows the internal bus configuration.

Table 13.1 Specifications of Internal Buses

Internal Bus Type		Description
Internal main bus	Internal main bus 1	• Operates in synchronization with ICLK • Bus protocol: AMBA AXI • Priority order decision: Round-robin
	Internal main bus 2	• Operates in synchronization with ICLK • Bus protocol: AMBA AHB • Priority order decision: Round-robin with fixed priority order (DMA0 with top priority)
Buses for Ethernet	Communication bus	Operates in synchronization with ICLK
Peripheral bus 1	ESC	Operates in synchronization with PCLKA
Peripheral bus 2	SCIFA, RSPIa	Operates in synchronization with SERICLK
Peripheral bus 3	RSCAN, CRC, ECM	Operates in synchronization with PCLKD
Peripheral bus 4	ELC, CMT, CMTW, WDTA, IWDTa, RIICa	Operates in synchronization with PCLKD
Peripheral bus 5	Clock generation circuit, CLMA	Operates in synchronization with PCLKB
External serial flash bus		Operates in synchronization with ICLK

Note: Peripheral buses 4 and 5 are on the same slave layer, and are connected with internal main bus 2.

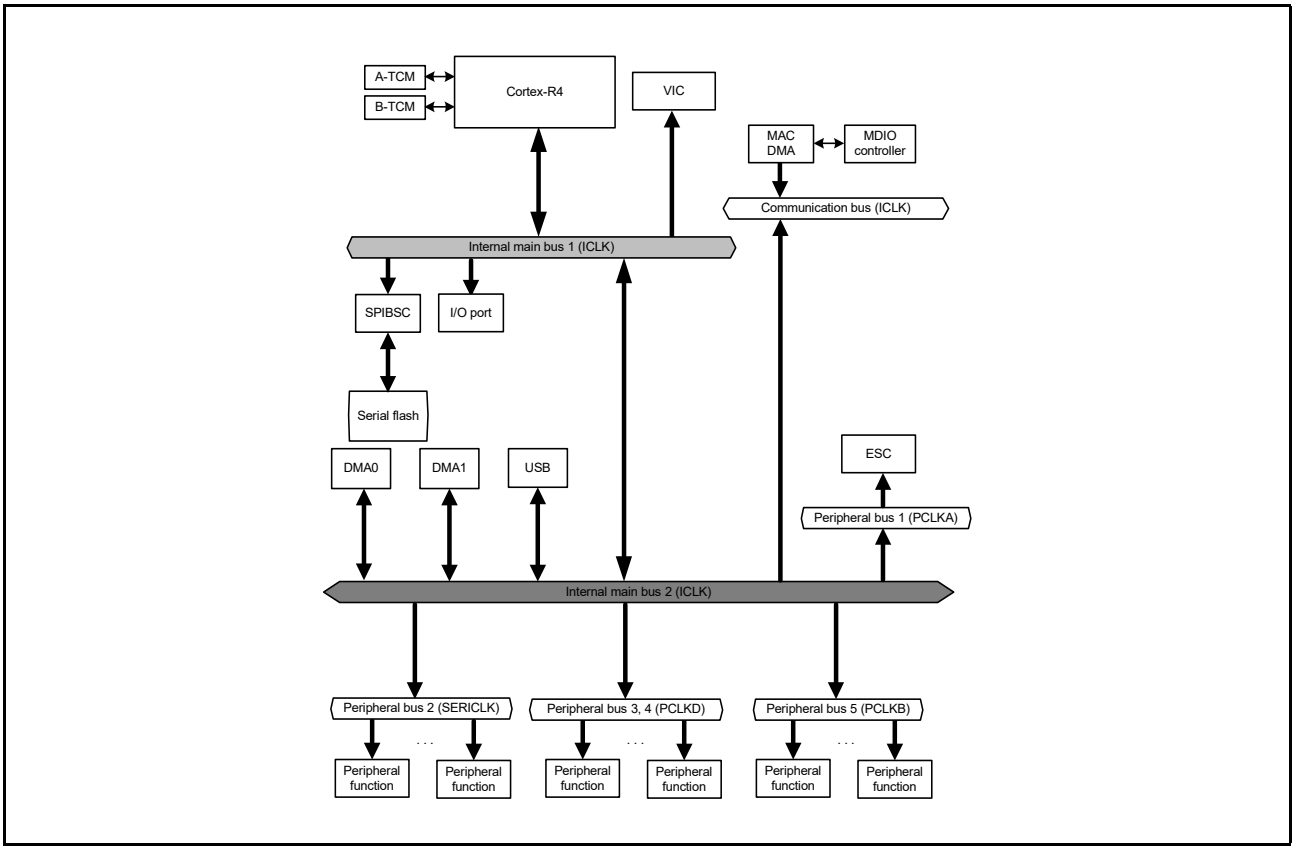


Figure 13.1 Bus Configuration

13.2 Internal Main Bus

Internal main buses 1 and 2 of this LSI both have a multiple-layer structure. If bus masters request to access different bus slaves respectively, multiple accesses are processed in parallel. If bus masters request to access the same bus slave, priority order decision is performed and accesses are processed sequentially according to the priority order.
Table 13.2 lists the connection between the bus master and the bus slave for internal main bus 1, and Table 13.3 lists the bus master-slave connection for internal main bus 2.

Table 13.2 Internal Main Bus 1: Connection between Bus Master and Bus Slave

Bus Slave \ Bus Master	CPU (Cortex-R4)	Internal Main Bus 2
CPU (Cortex-R4)	A	A
I/O port	A	A
SPIBSC	A	A
VIC	A	A
Internal Main Bus 2	A	—

A: Accessible
—: Inaccessible

Table 13.3 Internal Main Bus 2: Connection between Bus Master and Bus Slave

Bus Slave \ Bus Master	Internal Main Bus 1	DMA0	DMA1	USB
Internal main bus 1	—	A	A	A
DMA0	A	—	—	—
DMA1	A	—	—	—
USB	A	A	A	—
Peripheral bus 1	A	A	A	—
Peripheral bus 2	A	A	A	—
Peripheral bus 3	A	A	A	—
Peripheral bus 4	A	A	A	—
Peripheral bus 5	A	A	A	—

A: Accessible
—: Inaccessible

14. DMA Controller (DMACAa)

This LSI contains DMACAa (Direct Memory Access Controller) consisting of two units, DMACAa0 and DMACAa1. The DMACAa transfers data without using the CPU. When a transfer is requested, the DMACAa transfers data stored at the transfer source address to the transfer destination address.

14.1 Overview

Table 14.1 lists specifications of DMACAa.

Table 14.1 Specifications of DMACAa

Item		Description	
		DMACAa0	**DMACAa1**
Number of channels		16 channels	16 channels
Address space		4 Gbytes	
DMACAa activation source		External interrupts (IRQ) On-chip peripheral module requests/software requests*1	
Channel priority		• Can be selected from the fixed priority for channels 0 to 7, and 8 to 15, or round robin mode. • Channels 0 to 7 and 8 to 15 operate in round robin mode.	
Transfer data unit		8, 16, 32, 128, 256, and 512 bits	8, 16, 32, and 128 bits
Maximum transfer size		$2^{32} - 1$ bytes	
Transfer mode	Single transfer	Performs DMA transfer for each DMA activation request.	
	Block transfer	Performs DMA transfer of the specified size for a DMACAa transfer request.	
DMA mode	Register mode	• DMA transfer setting value: Control register value within the DMA controller • DMA transfer to the source/destination specified by a register	
	Link mode	• DMA transfer setting value: Descriptors in the internal RAM. • Various DMA transfer specified by descriptors can be performed (responsiveness: register mode > link mode).	
Interval function		The DMA transfer interval can be specified (bus occupation ratio can be adjusted).	
Skip function		• For the area to be accessed by DMA transfer, the continuous access size and the discrete access size (skip) can be set separately. • After the size of data specified for the continuous access is transferred, the specified number of addresses to be accessed next can be skipped.	
Suspending function		Current DMA transfer can be paused.	
Buffer flush function		When DMACAa is stopped forcibly, data in the buffer can be flushed.	
Interrupt request		Each channel has the following interrupt requests: • Transfer completion: Indicates completion of transfer of the specified size; each channel has this source. • Transfer error: Indicates a bus error; total of two sources, one for unit 0 and the other for unit 1.	

Note 1. A software request is output as a source of an on-chip peripheral module request from the interrupt controller. For how to set a software request, see section 12, Interrupt Controller (ICUA).

14.2 Register Descriptions

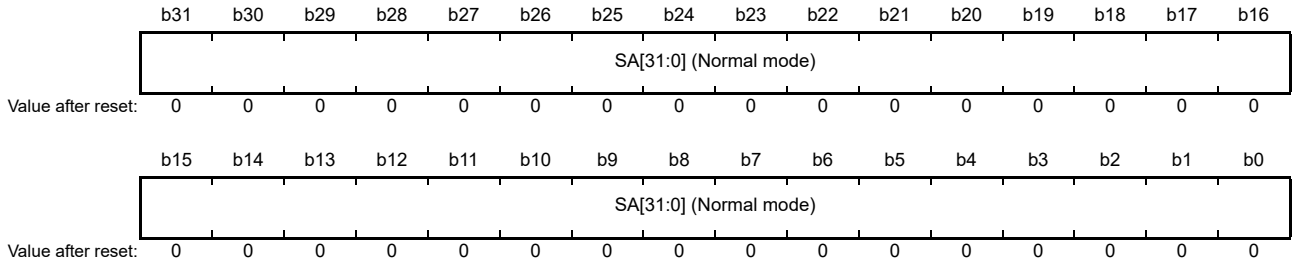
14.2.1 Next Source Address Register n (N0SA_n_N, N0SA_n_W, N1SA_n_N, N1SA_n_W)

The N0SA_n and N1SA_n registers set the DMA transfer source address of DMA channel n (n = 15 to 0).
The N0SA_n register is for Next0 Register Set, and the N1SA_n register is for Next1 Register Set.
In write-only mode (CHCFG_n register WONLY = 1), this register is used to set writing data.

- For N0SA_n_N and N1SA_n_N (normal mode)

Address(es): DMACAa0
N0SA_0_N: A006 2000h, N0SA_1_N: A006 2040h, N0SA_2_N: A006 2080h, N0SA_3_N: A006 20C0h,
N0SA_4_N: A006 2100h, N0SA_5_N: A006 2140h, N0SA_6_N: A006 2180h, N0SA_7_N: A006 21C0h,
N0SA_8_N: A006 2400h, N0SA_9_N: A006 2440h, N0SA_10_N: A006 2480h, N0SA_11_N: A006 24C0h,
N0SA_12_N: A006 2500h, N0SA_13_N: A006 2540h, N0SA_14_N: A006 2580h, N0SA_15_N: A006 25C0h
DMACAa1
N0SA_0_N: A006 3000h, N0SA_1_N: A006 3040h, N0SA_2_N: A006 3080h, N0SA_3_N: A006 30C0h,
N0SA_4_N: A006 3100h, N0SA_5_N: A006 3140h, N0SA_6_N: A006 3180h, N0SA_7_N: A006 31C0h,
N0SA_8_N: A006 3400h, N0SA_9_N: A006 3440h, N0SA_10_N: A006 3480h, N0SA_11_N: A006 34C0h,
N0SA_12_N: A006 3500h, N0SA_13_N: A006 3540h, N0SA_14_N: A006 3580h, N0SA_15_N: A006 35C0h

DMACAa0
N1SA_0_N: A006 200Ch, N1SA_1_N: A006 204Ch, N1SA_2_N: A006 208Ch, N1SA_3_N: A006 20CCh,
N1SA_4_N: A006 210Ch, N1SA_5_N: A006 214Ch, N1SA_6_N: A006 218Ch, N1SA_7_N: A006 21CCh,
N1SA_8_N: A006 240Ch, N1SA_9_N: A006 244Ch, N1SA_10_N: A006 248Ch, N1SA_11_N: A006 24CCh,
N1SA_12_N: A006 250Ch, N1SA_13_N: A006 254Ch, N1SA_14_N: A006 258Ch, N1SA_15_N: A006 25CCh
DMACAa1
N1SA_0_N: A006 300Ch, N1SA_1_N: A006 304Ch, N1SA_2_N: A006 308Ch, N1SA_3_N: A006 30CCh,
N1SA_4_N: A006 310Ch, N1SA_5_N: A006 314Ch, N1SA_6_N: A006 318Ch, N1SA_7_N: A006 31CCh,
N1SA_8_N: A006 340Ch, N1SA_9_N: A006 344Ch, N1SA_10_N: A006 348Ch, N1SA_11_N: A006 34CCh,
N1SA_12_N: A006 350Ch, N1SA_13_N: A006 354Ch, N1SA_14_N: A006 358Ch, N1SA_15_N: A006 35CCh



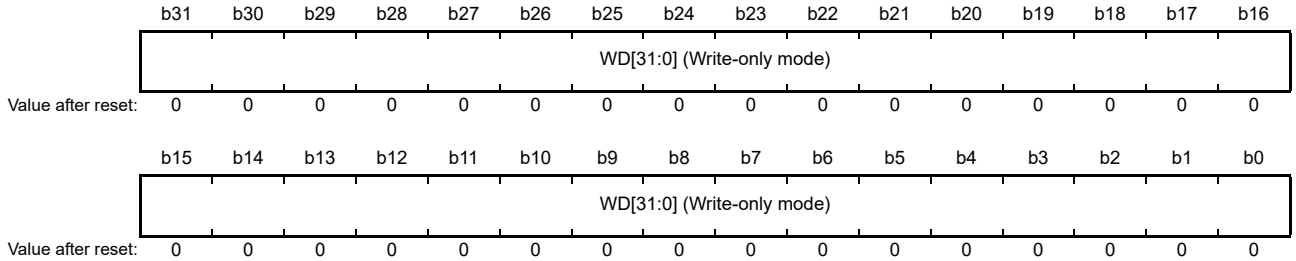
Bit	Symbol	Bit Name	Description	R/W
b31 to b0	SA[31:0] (Normal mode)	Source Address	Sets the start address of the DMA transfer source.	R/W

Note: During link mode transfer, descriptor read data is set to the N0SA_n_N register automatically.

- For N0SA_n_W and N1SA_n_W (write-only mode)

Address(es): DMACAa0
N0SA_0_W: A006 2000h, N0SA_1_W: A006 2040h, N0SA_2_W: A006 2080h, N0SA_3_W: A006 20C0h,
N0SA_4_W: A006 2100h, N0SA_5_W: A006 2140h, N0SA_6_W: A006 2180h, N0SA_7_W: A006 21C0h,
N0SA_8_W: A006 2400h, N0SA_9_W: A006 2440h, N0SA_10_W: A006 2480h, N0SA_11_W: A006 24C0h,
N0SA_12_W: A006 2500h, N0SA_13_W: A006 2540h, N0SA_14_W: A006 2580h, N0SA_15_W: A006 25C0h
DMACAa1
N0SA_0_W: A006 3000h, N0SA_1_W: A006 3040h, N0SA_2_W: A006 3080h, N0SA_3_W: A006 30C0h,
N0SA_4_W: A006 3100h, N0SA_5_W: A006 3140h, N0SA_6_W: A006 3180h, N0SA_7_W: A006 31C0h,
N0SA_8_W: A006 3400h, N0SA_9_W: A006 3440h, N0SA_10_W: A006 3480h, N0SA_11_W: A006 34C0h,
N0SA_12_W: A006 3500h, N0SA_13_W: A006 3540h, N0SA_14_W: A006 3580h, N0SA_15_W: A006 35C0h

DMACAa0
N1SA_0_W: A006 200Ch, N1SA_1_W: A006 204Ch, N1SA_2_W: A006 208Ch, N1SA_3_W: A006 20CCh,
N1SA_4_W: A006 210Ch, N1SA_5_W: A006 214Ch, N1SA_6_W: A006 218Ch, N1SA_7_W: A006 21CCh,
N1SA_8_W: A006 240Ch, N1SA_9_W: A006 244Ch, N1SA_10_W: A006 248Ch, N1SA_11_W: A006 24CCh,
N1SA_12_W: A006 250Ch, N1SA_13_W: A006 254Ch, N1SA_14_W: A006 258Ch, N1SA_15_W: A006 25CCh
DMACAa1
N1SA_0_W: A006 300Ch, N1SA_1_W: A006 304Ch, N1SA_2_W: A006 308Ch, N1SA_3_W: A006 30CCh,
N1SA_4_W: A006 310Ch, N1SA_5_W: A006 314Ch, N1SA_6_W: A006 318Ch, N1SA_7_W: A006 31CCh,
N1SA_8_W: A006 340Ch, N1SA_9_W: A006 344Ch, N1SA_10_W: A006 348Ch, N1SA_11_W: A006 34CCh,
N1SA_12_W: A006 350Ch, N1SA_13_W: A006 354Ch, N1SA_14_W: A006 358Ch, N1SA_15_W: A006 35CCh



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	WD[31:0] (Write-only mode)	Write Data	Sets write data for write-only mode.	R/W

Note: During link mode transfer, descriptor read data is set to the N0SA_n_W register automatically.

14.2.2 Next Destination Address Register n (N0DA_n and N1DA_n)

The N0DA_n and N1DA_n registers set the DMA transfer destination address of DMA channel n (n = 15 to 0).
The N0DA_n register is for Next0 Register Set, and the N1DA_n register is for the Next1 Register Set.

Address(es): DMACAa0
N0DA_0: A006 2004h, N0DA_1: A006 2044h, N0DA_2: A006 2084h, N0DA_3: A006 20C4h,
N0DA_4: A006 2104h, N0DA_5: A006 2144h, N0DA_6: A006 2184h, N0DA_7: A006 21C4h,
N0DA_8: A006 2404h, N0DA_9: A006 2444h, N0DA_10: A006 2484h, N0DA_11: A006 24C4h,
N0DA_12: A006 2504h, N0DA_13: A006 2544h, N0DA_14: A006 2584h, N0DA_15: A006 25C4h
DMACAa1
N0DA_0: A006 3004h, N0DA_1: A006 3044h, N0DA_2: A006 3084h, N0DA_3: A006 30C4h,
N0DA_4: A006 3104h, N0DA_5: A006 3144h, N0DA_6: A006 3184h, N0DA_7: A006 31C4h,
N0DA_8: A006 3404h, N0DA_9: A006 3444h, N0DA_10: A006 3484h, N0DA_11: A006 34C4h,
N0DA_12: A006 3504h, N0DA_13: A006 3544h, N0DA_14: A006 3584h, N0DA_15: A006 35C4h

DMACAa0
N1DA_0: A006 2010h, N1DA_1: A006 2050h, N1DA_2: A006 2090h, N1DA_3: A006 20D0h,
N1DA_4: A006 2110h, N1DA_5: A006 2150h, N1DA_6: A006 2190h, N1DA_7: A006 21D0h,
N1DA_8: A006 2410h, N1DA_9: A006 2450h, N1DA_10: A006 2490h, N1DA_11: A006 24D0h,
N1DA_12: A006 2510h, N1DA_13: A006 2550h, N1DA_14: A006 2590h, N1DA_15: A006 25D0h
DMACAa1
N1DA_0: A006 3010h, N1DA_1: A006 3050h, N1DA_2: A006 3090h, N1DA_3: A006 30D0h,
N1DA_4: A006 3110h, N1DA_5: A006 3150h, N1DA_6: A006 3190h, N1DA_7: A006 31D0h,
N1DA_8: A006 3410h, N1DA_9: A006 3450h, N1DA_10: A006 3490h, N1DA_11: A006 34D0h,
N1DA_12: A006 3510h, N1DA_13: A006 3550h, N1DA_14: A006 3590h, N1DA_15: A006 35D0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	DA[31:0]	Destination Address	Sets the start address of the DMA transfer destination.	R/W

Note: During link mode transfer, descriptor read data is set to the N0DA_n register automatically.

14.2.3 Next Transaction Byte Register n (N0TB_n and N1TB_n)

The N0TB_n and N1TB_n registers set the total number of transfer bytes of DMA channel n (n = 15 to 0).
The N0TB_n register is for Next0 Register Set, and the N1TB_n register is for Next1 Register Set.

Address(es): DMACAa0
N0TB_0: A006 2008h, N0TB_1: A006 2048h, N0TB_2: A006 2088h, N0TB_3: A006 20C8h,
N0TB_4: A006 2108h, N0TB_5: A006 2148h, N0TB_6: A006 2188h, N0TB_7: A006 21C8h,
N0TB_8: A006 2408h, N0TB_9: A006 2448h, N0TB_10: A006 2488h, N0TB_11: A006 24C8h,
N0TB_12: A006 2508h, N0TB_13: A006 2548h, N0TB_14: A006 2588h, N0TB_15: A006 25C8h
DMACAa1
N0TB_0: A006 3008h, N0TB_1: A006 3048h, N0TB_2: A006 3088h, N0TB_3: A006 30C8h,
N0TB_4: A006 3108h, N0TB_5: A006 3148h, N0TB_6: A006 3188h, N0TB_7: A006 31C8h,
N0TB_8: A006 3408h, N0TB_9: A006 3448h, N0TB_10: A006 3488h, N0TB_11: A006 34C8h,
N0TB_12: A006 3508h, N0TB_13: A006 3548h, N0TB_14: A006 3588h, N0TB_15: A006 35C8h

DMACAa0
N1TB_0: A006 2014h, N1TB_1: A006 2054h, N1TB_2: A006 2094h, N1TB_3: A006 20D4h,
N1TB_4: A006 2114h, N1TB_5: A006 2154h, N1TB_6: A006 2194h, N1TB_7: A006 21D4h,
N1TB_8: A006 2414h, N1TB_9: A006 2454h, N1TB_10: A006 2494h, N1TB_11: A006 24D4h,
N1TB_12: A006 2514h, N1TB_13: A006 2554h, N1TB_14: A006 2594h, N1TB_15: A006 25D4h
DMACAa1
N1TB_0: A006 3014h, N1TB_1: A006 3054h, N1TB_2: A006 3094h, N1TB_3: A006 30D4h,
N1TB_4: A006 3114h, N1TB_5: A006 3154h, N1TB_6: A006 3194h, N1TB_7: A006 31D4h,
N1TB_8: A006 3414h, N1TB_9: A006 3454h, N1TB_10: A006 3494h, N1TB_11: A006 34D4h,
N1TB_12: A006 3514h, N1TB_13: A006 3554h, N1TB_14: A006 3594h, N1TB_15: A006 35D4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TB[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TB[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	TB[31:0]	Transaction Byte	Sets the total number of transfer bytes. Note 1. Do not start DMA transfer when 0 is set.	R/W

Note: During link mode transfer, descriptor read data is set to the N0TB_n register automatically.

14.2.4 Current Source Address Register (CRSA_n)

The CRSA_n register is a register that indicates the DMA transfer source address of DMA channel n (n = 15 to 0).

During DMA transfer, the value is incremented automatically (fixed when the SAD bit in the CHCFG_n register = 1, and not fixed when the WONLY bit in the CHCFG_n register = 1).

Address(es): DMACAa0
CRSA_0: A006 2018h, CRSA_1: A006 2058h, CRSA_2: A006 2098h, CRSA_3: A006 20D8h,
CRSA_4: A006 2118h, CRSA_5: A006 2158h, CRSA_6: A006 2198h, CRSA_7: A006 21D8h,
CRSA_8: A006 2418h, CRSA_9: A006 2458h, CRSA_10: A006 2498h, CRSA_11: A006 24D8h,
CRSA_12: A006 2518h, CRSA_13: A006 2558h, CRSA_14: A006 2598h, CRSA_15: A006 25D8h
DMACAa1
CRSA_0: A006 3018h, CRSA_1: A006 3058h, CRSA_2: A006 3098h, CRSA_3: A006 30D8h,
CRSA_4: A006 3118h, CRSA_5: A006 3158h, CRSA_6: A006 3198h, CRSA_7: A006 31D8h,
CRSA_8: A006 3418h, CRSA_9: A006 3458h, CRSA_10: A006 3498h, CRSA_11: A006 34D8h,
CRSA_12: A006 3518h, CRSA_13: A006 3558h, CRSA_14: A006 3598h, CRSA_15: A006 35D8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRSA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRSA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRSA[31:0]	Current Source Address	Indicates the read address of the next DMA transfer.	R

The value after a reset is loaded from the following registers:

In register mode:
Loads the transfer source address from the Next0/1 register.

In link mode:
Loads the transfer source address from the descriptor read data (The hardware inputs descriptor read data to the N0SA_n register automatically, and loads it to the CRSA_n register when transfer starts).

The value is incremented when reading data from the transfer source is completed.

Read this register after DMA stops (when the TACT bit in the CHSTAT_n register = 0). (Handle the value during DMA operation as a reference value.)

14.2.5 Current Destination Address Register (CRDA_n)

The CRDA_n register indicates the DMA transfer destination address of DMA channel n (n = 15 to 0).
During DMA transfer, the value is incremented automatically (fixed when the DAD bit in the CHCFG_n register = 1).

Address(es): DMACAa0
CRDA_0: A006 201Ch, CRDA_1: A006 205Ch, CRDA_2: A006 209Ch, CRDA_3: A006 20DCh,
CRDA_4: A006 211Ch, CRDA_5: A006 215Ch, CRDA_6: A006 219Ch, CRDA_7: A006 21DCh,
CRDA_8: A006 241Ch, CRDA_9: A006 245Ch, CRDA_10: A006 249Ch, CRDA_11: A006 24DCh,
CRDA_12: A006 251Ch, CRDA_13: A006 255Ch, CRDA_14: A006 259Ch, CRDA_15: A006 25DCh
DMACAa1
CRDA_0: A006 301Ch, CRDA_1: A006 305Ch, CRDA_2: A006 309Ch, CRDA_3: A006 30DCh,
CRDA_4: A006 311Ch, CRDA_5: A006 315Ch, CRDA_6: A006 319Ch, CRDA_7: A006 31DCh,
CRDA_8: A006 341Ch, CRDA_9: A006 345Ch, CRDA_10: A006 349Ch, CRDA_11: A006 34DCh,
CRDA_12: A006 351Ch, CRDA_13: A006 355Ch, CRDA_14: A006 359Ch, CRDA_15: A006 35DCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRDA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRDA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRDA[31:0]	Current Destination Address	Indicates the write address of the next DMA transfer.	R

The value after a reset is loaded from the following registers:

In register mode:
Loads the transfer destination address from the Next0/1 register.

In link mode:
Loads the transfer destination address from the descriptor read data (The hardware inputs the descriptor read data to the N0DA_n register automatically, and loads it to the CRDA_n register when transfer starts).

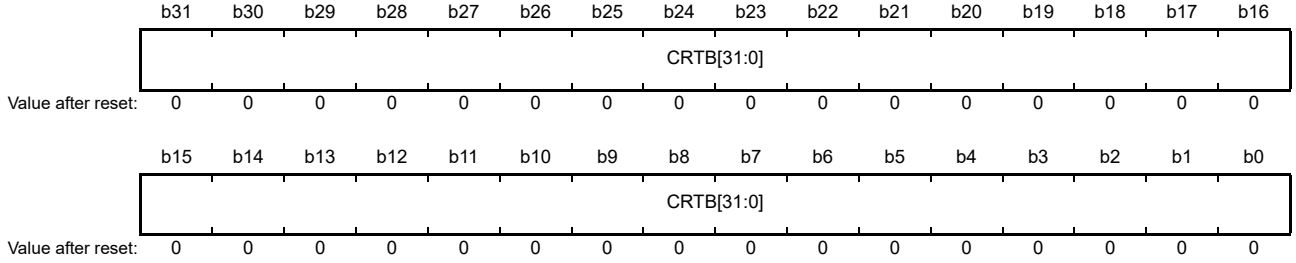
The value is incremented when writing data to the transfer destination completes.

Read this register after DMA stops (when the TACT bit in the CHSTAT_n register = 0). (Handle the value during DMA operation as a reference value.)

14.2.6 Current Transaction Byte Register (CRTB_n)

The CRTB_n register indicates the total number of transfer bytes of DMA channel n (n = 15 to 0). The value is cleared to 0 when transfer completes.
During DMA transfer, the value is decremented automatically.

Address(es): DMACAa0
CRTB_0: A006 2020h, CRTB_1: A006 2060h, CRTB_2: A006 20A0h, CRTB_3: A006 20E0h,
CRTB_4: A006 2120h, CRTB_5: A006 2160h, CRTB_6: A006 21A0h, CRTB_7: A006 21E0h,
CRTB_8: A006 2420h, CRTB_9: A006 2460h, CRTB_10: A006 24A0h, CRTB_11: A006 24E0h,
CRTB_12: A006 2520h, CRTB_13: A006 2560h, CRTB_14: A006 25A0h, CRTB_15: A006 25E0h
DMACAa1
CRTB_0: A006 3020h, CRTB_1: A006 3060h, CRTB_2: A006 30A0h, CRTB_3: A006 30E0h,
CRTB_4: A006 3120h, CRTB_5: A006 3160h, CRTB_6: A006 31A0h, CRTB_7: A006 31E0h,
CRTB_8: A006 3420h, CRTB_9: A006 3460h, CRTB_10: A006 34A0h, CRTB_11: A006 34E0h,
CRTB_12: A006 3520h, CRTB_13: A006 3560h, CRTB_14: A006 35A0h, CRTB_15: A006 35E0h



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRTB[31:0]	Current Transaction Byte	Indicates the remaining number of transfer bytes that is currently performed.	R

The value after a reset is loaded from the following registers:

In register mode:
Loads the number of transfer bytes from the Next0/1 register.

In link mode:
Loads the number of transfer bytes from descriptor read data (The hardware inputs the descriptor read data to the N0TB_n register automatically, and loads it to the CRTB_n register when transfer starts).

The value is decremented when writing data to the transfer destination completes.

Read this register after DMA stops (when the TACT bit in the CHSTAT_n register = 0). (Handle the value during DMA operation as a reference value.)

14.2.7 Channel Status Register n (CHSTAT_n)

The CHSTAT_n register indicates the status of DMA channel n (n = 15 to 0).

Address(es): DMACAa0
CHSTAT_0: A006 2024h, CHSTAT_1: A006 2064h, CHSTAT_2: A006 20A4h, CHSTAT_3: A006 20E4h,
CHSTAT_4: A006 2124h, CHSTAT_5: A006 2164h, CHSTAT_6: A006 21A4h, CHSTAT_7: A006 21E4h,
CHSTAT_8: A006 2424h, CHSTAT_9: A006 2464h, CHSTAT_10: A006 24A4h, CHSTAT_11: A006 24E4h,
CHSTAT_12: A006 2524h, CHSTAT_13: A006 2564h, CHSTAT_14: A006 25A4h, CHSTAT_15: A006 25E4h
DMACAa1
CHSTAT_0: A006 3024h, CHSTAT_1: A006 3064h, CHSTAT_2: A006 30A4h, CHSTAT_3: A006 30E4h,
CHSTAT_4: A006 3124h, CHSTAT_5: A006 3164h, CHSTAT_6: A006 31A4h, CHSTAT_7: A006 31E4h,
CHSTAT_8: A006 3424h, CHSTAT_9: A006 3464h, CHSTAT_10: A006 34A4h, CHSTAT_11: A006 34E4h,
CHSTAT_12: A006 3524h, CHSTAT_13: A006 3564h, CHSTAT_14: A006 35A4h, CHSTAT_15: A006 35E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DNUM[7:0]								—	—	—	—	—	SWPRQ	DMARQM	INTM
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	MODE	DER	DW	DL	SR	—	END	ER	SUS	TACT	RQST	EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	EN	DMA Activation Enable	Indicates the status of operation enable/disable of DMA channel n. 0: Operation is disabled. 1: Operation is enabled. [Setting condition] • Writes 1 to the SETEN bit in the CHCTRL_n register. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLREN bit in the CHCTRL_n register. • When a bus error was reported during transfer • When all DMA transfers complete in register mode (transfers complete when the REN bit in the CHCFG_n register = 0) • In link mode, when DMA transfer (write back when WBD = 0) of the last descriptor (LE = 1) completes • When reading the descriptor in link mode stopped (when LV = 0, and DRRP in the CHCFG_n register = 0)	R

Bit	Symbol	Bit Name	Description	R/W
b1	RQST	DMA Transfer Request	This bit indicates that the transfer request is accepted. 0: The DMA transfer request is not accepted. 1: The DMA transfer request is accepted. [Setting condition] • The transfer request is accepted. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the SWRST bit in the CHCTRL_n register. • Writes 1 to the CLRRQ bit in the CHCTRL_n register. • When all DMA transfers complete in register mode (transfer completes when the REN bit in the CHCFG_n register = 0) • In link mode, DMA transfer of the last descriptor (LE = 1) completes • In link mode, reading the descriptor is disabled (when LV = 0, and DRRP in the CHCFG_n register = 0) • In link mode, the DEM bit in the CHCFG_n register = 0, and DMA transfer completes. • When a buss error was reported to the master interface	R
b2	TACT	DMACAa Operating Status	This bit indicates that DMACAa is running. This bit is used to make sure the channel stops completely. For details, see section 14.3.8, DMA Transfer Status. 0: DMA in Channel_n stops. 1: DMA in Channel_n is running. [Setting condition] • Write 1 to the SETEN bit in the CHCTRL_n register (Reading the descriptor starts, or the DMA request is being waited). [Clearing condition] • When the internal state is idling (the EN bit in the CHSTAT_n register is cleared to 0, and all DMA transfer completes).	R
b3	SUS	Suspend	This bit indicates that the channel is suspended. For details, see section 14.3.9, Suspending a Transfer. 0: Channel_n is not suspended. 1: Channel_n is suspended. [Setting condition] • Writes 1 to the SETSUS bit in the CHCTRL_n register during DMA transfer of Channel_n, and the internal state is changed to the suspended state. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLRSUS bit in the CHCTRL_n register. • Writes 1 to the CLREN bit in the CHCTRL_n register. • Condition for clearing the EN bit in the CHSTAT_n register	R
b4	ER	DMA Error	This bit indicates that a DMA error interrupt is generated as a result of the bus error during DMA transfer. 0: No bus error occurred. 1: A bus error occurred. [Setting condition] • A buss error is reported to the bus cycle [Clearing condition] • Writes 1 to the SWRST bit in the CHCTRL_n register.	R

Bit	Symbol	Bit Name	Description	R/W
b5	END	DMA Transfer Completion Interrupt	This bit indicates that DMA transfer completes, and a DMA interrupt is generated. 0: DMA transfer is not completed. 1: DMA transfer is completed. [Setting conditions] When any of the following conditions is met: • The following conditions being met while the DEM bit in the CHCFG_n register = 0 (1) When a transfer for the total number of transfer bytes loaded to the CRTB register completes in register mode (2) When a transfer for the total number of transfer bytes loaded to the CRTB register completes in link mode while the WBD bit in the header of the descriptor is 1 (3) Completion of descriptor write-back in link mode while the WBD bit in the header of the descriptor is 0 • In link mode, the descriptor is read, LV of header = 0, and DRRP in the CHCFG_n register = 0, and DIM = 0. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLREND bit in the CHCTRL_n register. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R
b6	—	Reserved	This bit is always read as 0.	R
b7	SR	Next Register Select	In register mode, indicates the selected register set. 0: Next0 Register Set 1: Next1 Register Set [Setting condition] • Set 1 to the RSEL bit in the CHCFG_n register. [Clearing condition] • Clears the RSEL bit in the CHCFG_n register to 0.	R
b8	DL	Descriptor Load	Indicates that the descriptor is being read. In addition, if a bus error is reported while the descriptor is being read, 1 is retained. 0: The descriptor is not being read. 1: (When ER = 0) The descriptor is being read in link mode. (When ER = 1) A buss error occurred while the descriptor is being read in link mode. [Setting condition] • Reading the descriptor in link mode is started [Clearing conditions] When any of the following conditions is met: • Reading the descriptor in link mode completes with the OK response. • Writes 1 to the SWRST bit in the CHCTRL_n register. (If 1 is retained for the bus error, it can be cleared to 0 only by the SWRST bit.)	R

Bit	Symbol	Bit Name	Description	R/W
b9	DW	Descriptor Write Back	Indicates that the descriptor is being written back. In addition, if a bus error is reported while the descriptor is written back, 1 is retained. 0: Header is not written back in link mode. 1: (The ER bit in the CHSTAT_n register = 0) Header is written back in link mode. (The ER bit in the CHSTAT_n register = 1) A buss error occurred when header is written back in link mode. [Setting condition] • Writing back of header in link mode started. [Clearing conditions] • Writing back of header in link mode completes with the OK response. • Writes 1 to the SWRST bit in the CHCTRL_n register (If 1 is retained due to the buss error, it can be cleared to 0 only by the SWRST bit).	R
b10	DER	Descriptor Error	Indicates that the read descriptor is invalid (LV = 0). (Does not depend on the value of the DIM bit in the CHCFG_n register.) 0: No descriptor error occurred. 1: Descriptor error occurred. [Setting condition] • In link mode, the DRRP bit in the CHCFG_n register = 0, and the read descriptor's LV is 0. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLRDE bit in the CHCTRL_n register. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R
b11	MODE	DMA Mode	This bit indicates DMA mode. Indicates the setting value of the DMS bit in the CHCFG_n register. 0: Register mode 1: Link mode	R
b15 to b12	—	Reserved	These bits are always read as 0.	R
b16	INTM	Interrupt Request Mask	Indicates the temporary mask status of DMA interrupt output. 1: Temporary mask status 0: The temporary mask status is cleared. [Setting condition] • Writes 1 to the SETINTM bit in the CHCTRL_n register. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLRINTM bit in the CHCTRL_n register. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R

Bit	Symbol	Bit Name	Description	R/W
b17	DMARQM	DMA Activation Request Mask	Indicates the temporary mask status of the DMA request. 1: Temporary mask status 0: Temporary mask status is cleared. [Setting condition] • Sets 1 to the SETDMARQM bit in the CHCTRL_n register. [Clearing conditions] When any of the following conditions is met: • Writes 1 to the CLRDMARQM bit in the CHCTRL_n register. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R
b18	SWPRQ	Forced Ejection Request	Indicates the forced ejection request status. Indicates the software forced ejection request (a request activated by the SETSSWPRQ bit in the CHCTRL_n register). 1: Forced ejection is requested. 0: Forced ejection is not requested. [Setting condition] • Sets 1 to the SETSSWPRQ bit in the CHCTRL_n register. [Clearing conditions] When any of the following conditions is met: • The amount of data in the buffer becomes 0 because of forced ejection. • Writes 1 to the SWRST bit in the CHCTRL_n register.	R
b23 to b19	—	Reserved	These bits are always read as 0.	R
b31 to b24	DNUM	Amount of Data in the Buffer	These bits indicate the amount of valid data in the buffer. Read data from the DMA transfer source, and indicate the amount of data that is not written to the transfer destination (in bytes). [Increment condition] • When DMA read transfer completes [Decrement condition] • When DMA write transfer completes [Clearing conditions] When any of the following conditions is met: • Condition for clearing the EN bit • Writes 1 to the SWRST bit in the CHCTRL_n register.	R

Note 1. Handle the transfer when the ER bit in the CHSTAT_n register is set to 1 as invalid.
Note 2. To suspend DMA transfer, mask or clear the transfer request, or clear the EN bit in the CHSTAT_n register. See section 14.3.10, Aborting a Transfer for the procedure).
Note 3. To request a transfer by software, make sure that the previously requested DMA transfer operation completes (by checking Current Register), set the DMREQ bit in the DMACAa software activation register (DMASTG), and then activate DMA.

14.2.8 DMACAa Unit 0 Source Select Register i (DMA0SELi) (i = 0 to 15)

The DMA0SELi register selects the activation trigger source of channel i (i = 0 to 15) for DMACAa unit 0. For details on numbers that are selected by this source selection, see the vector numbers listed in Table 12.3, Cortex-R4/DMACAa Interrupt Vector Table. In addition, do not set the same source for multiple DMA0SELi and DMA1SELi registers. If the same source is set, the operation of this LSI cannot be guaranteed.

Address(es): DMA0.DMA0SEL0 A009 4000h, DMA0.DMA0SEL1 A009 4004h, DMA0.DMA0SEL2 A009 4008h, DMA0.DMA0SEL3 A009 400Ch, DMA0.DMA0SEL4 A009 4010h, DMA0.DMA0SEL5 A009 4014h, DMA0.DMA0SEL6 A009 4018h, DMA0.DMA0SEL7 A009 401Ch, DMA0.DMA0SEL8 A009 4020h, DMA0.DMA0SEL9 A009 4024h, DMA0.DMA0SEL10 A009 4028h, DMA0.DMA0SEL11 A009 402Ch, DMA0.DMA0SEL12 A009 4030h, DMA0.DMA0SEL13 A009 4034h, DMA0.DMA0SEL14 A009 4038h, DMA0.DMA0SEL15 A009 403Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	IFC0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	IFC0[7:0]	DMA channel source select	Select the trigger source of the DMA channel.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

14.2.9 DMACAa Unit 1 Source Select Register i (DMA1SELi) (i = 0 to 15)

The DMA1SELi register selects the activation trigger source of channel i (i = 0 to 15) for DMACAa unit 1. For details on numbers that are selected for this source selection, see the vector numbers listed in Table 12.3, Cortex-R4/DMACAa Interrupt Vector Table. In addition, do not set the same source for multiple DMA0SELi and DMA1SELi registers. If the same source is set, the operation cannot be guaranteed.

Address(es): DMA1.DMA1SEL0 A009 4040h, DMA1.DMA1SEL1 A009 4044h, DMA1.DMA1SEL2 A009 4048h, DMA1.DMA1SEL3 A009 404Ch, DMA1.DMA1SEL4 A009 4050h, DMA1.DMA1SEL5 A009 4054h, DMA1.DMA1SEL6 A009 4058h, DMA1.DMA1SEL7 A009 405Ch, DMA1.DMA1SEL8 A009 4060h, DMA1.DMA1SEL9 A009 4064h, DMA1.DMA1SEL10 A009 4068h, DMA1.DMA1SEL11 A009 406Ch, DMA1.DMA1SEL12 A009 4070h, DMA1.DMA1SEL13 A009 4074h, DMA1.DMA1SEL14 A009 4078h, DMA1.DMA1SEL15 A009 407Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	IFC1[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	IFC1[7:0]	DMA channel source select	Select the trigger source of the DMA channel.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

14.2.10 DMACAa Software Activation Register (DMASTG)

The DMASTG register controls activation of DMACAa by software.

Address(es): DMAC.DMASTG A009 4080h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMREQ1	DMREQ0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DMREQ0	DMA Unit 0 Software Activation	0: DMA transfer is not requested. 1: DMA transfer is requested.	W
b1	DMREQ1	DMA Unit 1 Software Activation	0: DMA transfer is not requested. 1: DMA transfer is requested.	W
b31 to b2	—	Reserved	The write value should be 0.	W

DMREQ0, DMREQ1 Bits (DMA Unit 0/1 Software Activation)

DMA transfer is requested if you select DMA activation by software for the DMA0SELi register and the DMA1SELi register (i = 0 to 15), and then write 1 to DMREQ0 and DMREQ1 bits.

These bits are write only. These bits are read as 0.

14.2.11 Channel Control Register n (CHCTRL_n)

The CHCTRL_n register controls DMA transfer of DMA channel n (n = 15 to 0).
This register is used to activate each function, and it does not retain the written value. These bits are always read as 0.
Only resources for channel n are masked temporary for forced ejection request and DMA transfer request inputs by the CLRDMARQM and SETDMARQM bits.

Address(es): DMACAa0
CHCTRL_0: A006 2028h, CHCTRL_1: A006 2068h, CHCTRL_2: A006 20A8h, CHCTRL_3: A006 20E8h,
CHCTRL_4: A006 2128h, CHCTRL_5: A006 2168h, CHCTRL_6: A006 21A8h, CHCTRL_7: A006 21E8h,
CHCTRL_8: A006 2428h, CHCTRL_9: A006 2468h, CHCTRL_10: A006 24A8h, CHCTRL_11: A006 24E8h,
CHCTRL_12: A006 2528h, CHCTRL_13: A006 2568h, CHCTRL_14: A006 25A8h, CHCTRL_15: A006 25E8h
DMACAa1
CHCTRL_0: A006 3028h, CHCTRL_1: A006 3068h, CHCTRL_2: A006 30A8h. CHCTRL_3: A006 30E8h,
CHCTRL_4: A006 3128h, CHCTRL_5: A006 3168h, CHCTRL_6: A006 31A8h, CHCTRL_7: A006 31E8h,
CHCTRL_8: A006 3428h, CHCTRL_9: A006 3468h, CHCTRL_10: A006 34A8h, CHCTRL_11: A006 34E8h,
CHCTRL_12: A006 3528h, CHCTRL_13: A006 3568h, CHCTRL_14: A006 35A8h, CHCTRL_15: A006 35E8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	CLRDMARQM	SETDMARQM	CLRINTM	SETINTM
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	SETSSWPRQ	—	SETREN	—	—	CLRSUS	SETSUS	CLRDE	—	CLRENDD	CLRRQ	SWRST	—	CLREN	SETEN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SETEN	DMA Activation Enable	Enables DMA transfer of DMA channel n. When this bit is set with the SWRST bit, clearing by the SWRST bit takes precedence, and a transfer does not start. This bit is always read as 0. 1: DMA transfer is enabled (The EN bit in the CHSTAT_n register is set). 0: Operation is not affected. Note: For resetting the DMA register, stop the ongoing DMA transfer by setting the CLREN bit and then set the SETEN bit.	R/W
b1	CLREN	DMA Activation Enable Clear	Clears the EN bit in the CHSTAT_n register (For details, see section 14.3.10, Aborting a Transfer). This bit is always read as 0. 1: DMA transfer is disabled (The EN bit in the CHSTAT_n register is cleared). 0: Operation is not affected.	R/W
b2	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b3	SWRST	Software Reset	Clears each bit in the CHSTAT_n register (For details on the bit to be cleared, see descriptions of the applicable bit). Set this bit to 0 when the EN bit and the TACT bit are cleared to 0. This bit is always read as 0. 1: Clears each bit in the CHSTAT_n register. 0: Operation is not affected.	R/W
b4	CLRRQ	DMA Transfer Request Clear	Clears the RQST bit in the CHSTAT_n register to 0. This bit is always read as 0. 1: Clears the RQST bit in the CHSTAT_n register. 0: Operation is not affected.	R/W
b5	CLREND	END Clear	Clears the END bit in the CHSTAT_n register to 0. This bit is always read as 0. 1: Clears the END bit. 0: Operation is not affected.	R/W
b6	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b7	CLRDE	DER Clear	Clears the descriptor error bit (DER) in the CHSTAT_n register to 0. This bit is always read as 0. 1: Clears the DER bit. 0: Operation is not affected.	R/W
b8	SETSUS	Suspend Request	If 1 is set to this bit when the EN bit in the CHSTAT_n register is 1, the current DMA transfer is suspended. This bit is always read as 0. 1: Suspends the current DMA transfer. 0: Operation is not affected.	R/W
b9	CLRSUS	Suspend Clear	If 1 is set to this bit when the SUS bit in the CHSTAT_n register is 1, the temporary suspend status is cleared. This bit is always read as 0. 1: Clears the temporary suspend status of the current DMA transfer. 0: Operation is not affected.	R/W
b11, b10	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W
b12	SETREN	REN Set Enable	Sets the register set enable bit (REN) in the CHCFG_n register. This bit is always read as 0. 1: Sets the REN bit in the CHCFG_n register. 0: Operation is not affected.	R/W
b13	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b14	SETSSWPRQ	Software Forced Ejection Request	Forcibly ejects data in the buffer to the transfer destination (See section 14.3.5, Forced Ejection Request). This bit is always read as 0. 1: Writes data in the buffer, which is not written, to the transfer destination. 0: Operation is not affected.	R/W
b15	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b16	SETINTM	Interrupt Request Mask	Masks DMA transfer completion interrupt output temporary. Besides, the INTM bit in the CHSTATn register is set to 1. This bit is always read as 0. 1: Masks a DMA transfer completion interrupt temporary. 0: Operation is not affected.	R/W
b17	CLRINTM	Interrupt Request Mask Clear	Clears the mask status of DMA transfer completion interrupt output. Besides, the INTM bit in the CHSTATn register is cleared to 0. If the mask state is cleared when the LVINT bit in the DCTRL register = 1, and the END bit in the CHSTAT_n register = 1, DMA transfer completion interrupt is deactivated. (It is not activated when LVINT = 0.) This bit is always read as 0. 1: Clears the mask state set by the SETINTM bit. 0: Operation is not affected.	R/W
b18	SETDMARQM	DMA Activation Request Mask	Masks DMA transfer request input temporary. Besides, the DMARQM bit in the CHSTATn register is set to 1. This bit is always read as 0. 1: Masks DMA transfer request input. 0: Operation is not affected.	R/W
b19	CLRDMARQM	DMA Activation Request Mask Clear	Clears the mask state of DMA transfer request input. Besides, the DMARQM bit in the CHSTATn register is cleared to 0. This bit is always read as 0. 1: Clears the mask state set by the SETDMARQM bit. 0: Operation is not affected.	R/W
b31 to b20	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W

14.2.12 Channel Configuration Register n (CHCFG_n)

The CHCFG_n register controls DMA transfer of DMA channel n (n = 15 to 0).

Specify the detection method depending on the DMA transfer source to be used. For details on the DMA request signals, see section 14.3.4.1, Specifying Detection Operation of DMA Transfer Requests for Each Source.

Address(es): DMACAa0
CHCFG_0: A006 202Ch, CHCFG_1: A006 206Ch, CHCFG_2: A006 20ACh, CHCFG_3: A006 20ECh,
CHCFG_4: A006 212Ch, CHCFG_5: A006 216Ch, CHCFG_6: A006 21ACh, CHCFG_7: A006 21ECh,
CHCFG_8: A006 242Ch, CHCFG_9: A006 246Ch, CHCFG_10: A006 24ACh, CHCFG_11: A006 24ECh,
CHCFG_12: A006 252Ch, CHCFG_13: A006 256Ch, CHCFG_14: A006 25ACh, CHCFG_15: A006 25ECh
DMACAa1
CHCFG_0: A006 302Ch, CHCFG_1: A006 306Ch, CHCFG_2: A006 30ACh, CHCFG_3: A006 30ECh,
CHCFG_4: A006 312Ch, CHCFG_5: A006 316Ch, CHCFG_6: A006 31ACh, CHCFG_7: A006 31ECh,
CHCFG_C: A006 342Ch, CHCFG_9: A006 346Ch, CHCFG_10: A006 34ACh, CHCFG_11: A006 34ECh,
CHCFG_12: A006 352Ch, CHCFG_13: A006 356Ch, CHCFG_14: A006 35ACh, CHCFG_15: A006 35ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DMS	REN	RSW	RSEL	SBE	DIM	—	DEM	WONLY	TM	DAD	SAD	DDS[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SDS[3:0]				DRRP	—	—	—	—	LVL	HIEN	LOEN	—	SEL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SEL[2:0]	Pin Select	Sets channels of DMACAa. Set the following values so that the channels of CHCFG_n (n = 0 to 15) and the channel set by SEL become the same: For example, set 001b to the SEL bit in CHCFG_1. Similarly, set 001b to the SEL bit in CHCFG_1. DMACAa0/1 b2 b0 000: Channel 0/8 001: Channel 1/9 010: Channel 2/10 011: Channel 3/11 100: Channel 4/12 101: Channel 5/13 110: Channel 6/14 111: Channel 7/15	R/W
b3	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b4	LOEN	L Detection Enable	Specifies the detection method of the DMA request signal. When LVL = 0 LOEN = 1: If DMA transfer request input detects a falling edge, it is regarded as requested. LOEN = 0: If the DMA transfer request input falls, the request is not recognized (a value after a reset). When LVL = 1 LOEN = 1: If DMA transfer request input detects the Low level, it is regarded as requested. LOEN = 0: If the DMA transfer request input is the Low level, the request is not recognized (a value after a reset).	R/W
b5	HIEN	H Detection Enable	Specifies the detection method of the DMA request signal. When LVL = 0 HIEN = 1: If DMA transfer request input detects a rising edge, it is regarded as requested. HIEN = 0: If the DMA transfer request input rises, the request is not recognized (a value after a reset). When LVL = 1 HIEN = 1: If DMA transfer request input detects the High level, it is regarded as requested. HIEN = 0: If the DMA transfer request input is the High level, the request is not recognized (a value after a reset).	R/W

Bit	Symbol	Bit Name	Description	R/W
b6	LVL	Level Detection Enable	Specifies the detection method of the DMA request signal. 0: Detects the edge (a value after a reset). 1: Detects the level.	R/W
b8, b7	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W
b9	—	Reserved	Always write 1 to this bit at the initial setting.	R/W
b10	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b11	DRRP	Descriptor Reload Enable	Specifies the operation when LV of header = 0 while during descriptor read (see section 14.3.1.2, Link Mode (1) Operation flows in link mode). 0: Sets the DER bit in the CHSTAT_n register, and stops the operation (a value after a reset). 1: Continues reading the same descriptor until LV becomes 1. When LV becomes 1, DMA transfer using the descriptor value starts. The DSCITVL register controls the descriptor read interval.	R/W
b15 to b12	SDS[3:0]	Source Data Size	Sets the size of data in the transfer source to be transferred at a time. For a single transfer, the specified amount of data is transferred by a single request. For a block transfer, data is transferred as many times as the setting value × N times until the CRTB register is cleared to 0. For a transfer of 32 bits or more, 32 bits × N times burst transfers are performed. The SDS[3] bit switches between normal mode and skip mode. 0: Normal mode (a value after a reset). 1: Skip mode The SDS[2:0] bits set the transfer size. b14 b12 000: 8 bits (a value after a reset) 001: 16 bits 010: 32 bits 011: Setting is prohibited. 100: 128 bits*2 101: 256 bits (Can be set only for DMACAa0) 110: 512 bits (Can be set only for DMACAa0)*1 111: Setting is prohibited.	R/W
b19 to b16	DDS[3:0]	Destination Data Size	Sets the size of data in the transfer destination to be transferred at a time. The DDS[3] bit switches between normal mode and skip mode. 0: Normal mode (a value after a reset). 1: Skip mode The DDS[2:0] bits set the transfer size. b18 b16 000: 8 bits (a value after a reset) 001: 16 bits 010: 32 bits 011: Setting is prohibited. 100: 128 bits*2 101: 256 bits (Can be set only for DMACAa0) 110: 512 bits (Can be set only for DMACAa0)*1 111: Setting is prohibited.	R/W
b20	SAD	Source Address Count Direction	Sets the count direction of the transfer source address of DMA channel n. 0: Increment (a value after a reset) 1: Fixed To use SKIP mode on the transfer source, do not specify SAD = 1 (fixed). If SAD = 1 (fixed) is specified, the address of the transfer source should be aligned with a boundary of the size specified by the SDS[3:0] bits.	R/W

Bit	Symbol	Bit Name	Description	R/W
b21	DAD	Destination Address Count Direction	Sets the count direction of the transfer destination address of DMA channel n. 0: Increment (a value after a reset) 1: Fixed To use SKIP mode on the transfer destination, do not specify DAD = 1 (fixed). If DAD = 1 (fixed) is specified, the address of the transfer destination should be aligned with a boundary of the size specified by the DDS[3:0] bits.	R/W
b22	TM	Transfer Mode	Sets DMA transfer mode. 0: Single transfer mode (a value after a reset) 1: Block transfer mode	R/W
b23	WONLY	Write-Only Mode	Sets the write-only mode (see section 14.3.1.3, Write-Only Mode). 0: Normal mode (a value after a reset). 1: Write-only mode	R/W
b24	DEM	Transfer Completion Interrupt Mask	Masks DMA transfer completion interrupt detection. If this bit is set to 1 when DMA transfer completion interrupt is output, DMA transfer completion interrupt is not activated. Besides, the END bit in the CHSTAT_n register is not set. In register mode, the DEM bit is automatically cleared to 0. In link mode, it is not cleared. 0: Does not mask (a value after a reset). 1: Masks. [Clearing condition] DEM = 1, and DMA transfer completes.	R/W
b25	—	Reserved	This bit is always read as 0. The write value should be 0.	R/W
b26	DIM	Descriptor Interrupt Mask	Sets the DMA transfer completion interrupt mask if LV = 0 when header of the descriptor is read. 0: Does not mask a DMA transfer completion interrupt (a value after reset). 1: Masks a DMA transfer completion interrupt.	R/W
b27	SBE	Buffer Flush Enable	If the EN bit in the CHSTAT_n register is cleared to 0 during DMA transfer, selects whether to stop flushing (writing) data that is already read and stored in the buffer. 0: Stops transfer without flushing data in the buffer (a value after a reset). 1: Stops transfer after flushing data in the buffer.	R/W
b28	RSEL	Next Register Select	Selects the Next register set to be executed next. This bit is valid in register mode only. When RSW = 1, the value is reversed automatically (0 is reversed to 1, 1 is reversed to 0) after DMA transfer completes. 0: Executes Next0 Register Set (a value after a reset). 1: Executes Next1 Register Set. [Transition condition] RSW = 1, and DMA transfer completes.	R/W
b29	RSW	RSEL Reverse	When DMA transfer completes, the RSEL bit is automatically reversed (0 is reversed to 1, 1 is reversed to 0). This bit is valid in register mode only. 0: Does not reverse the RSEL bit when DMA transfer completes (a value after a reset). 1: Reverses the RSEL bit when DMA transfer completes.	R/W

Bit	Symbol	Bit Name	Description	R/W
b30	REN	Register Set Enable	When DMA transfer completes, performs DMA transfer of the Next register set selected by the RSEL bit accordingly. This bit is valid in register mode only. 0: Does not perform DMA transfer accordingly. 1: Performs DMA transfer accordingly. [Setting conditions] When any of the following conditions is met: • Writes 1 to this bit. • Writes 1 to the SETREN bit in the CHCTRL_n register. [Clearing conditions] • Writes 0 to this bit. • REN = 1, and DMA transfer completes. To reset the REN bit during DMA transfer, use the SETREN bit in the CHCTRL_n register. Also, reset the CHCFG_n.DEM bit to mask detection of DMA transfer completion interrupt.	R/W
b31	DMS	DMA Mode Select	Sets DMA mode. 0: Register mode (a value after a reset). 1: Link mode	R/W

Note 1. If the size is set to 512 bits, the address of the transfer source/destination should be aligned with 512-bit boundaries.
Note 2. If the size is set to 128 bits when DMAC1 is used, the address of the transfer source/destination should be aligned with 128-bit boundaries.

14.2.13 Channel Interval Register n (CHITVL_n)

The CHITVL_n register sets the DMA transfer interval of DMA channel n (n = 15 to 0).
For details, see section 14.3.6, Interval Count Function.

Address(es): DMACAa0
CHITVL_0: A006 2030h, CHITVL_1: A006 2070h, CHITVL_2: A006 20B0h, CHITVL_3: A006 20F0h,
CHITVL_4: A006 2130h, CHITVL_5: A006 2170h, CHITVL_6: A006 21B0h, CHITVL_7: A006 21F0h,
CHITVL_8: A006 2430h, CHITVL_9: A006 2470h, CHITVL_10: A006 24B0h, CHITVL_11: A006 24F0h,
CHITVL_12: A006 2530h, CHITVL_13: A006 2570h, CHITVL_14: A006 25B0h, CHITVL_15: A006 25F0h
DMACAa1
CHITVL_0: A006 3030h, CHITVL_1: A006 3070h, CHITVL_2: A006 30B0h, CHITVL_3: A006 30F0h,
CHITVL_4: A006 3130h, CHITVL_5: A006 3170h, CHITVL_6: A006 31B0h, CHITVL_7: A006 31F0h,
CHITVL_8: A006 3430h, CHITVL_9: A006 3470h, CHITVL_10: A006 34B0h, CHITVL_11: A006 34F0h,
CHITVL_12: A006 3530h, CHITVL_13: A006 3570h, CHITVL_14: A006 35B0h, CHITVL_15: A006 35F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ITVL[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	ITVL	Interval	These bits set the DMA transfer internal.	R/W
b31 to b16	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W

14.2.14 Next Link Address Register n (NXLA_n)

The NXLA_n register sets the link address of DMA channel n (n = 15 to 0).

Address(es): DMACAa0
NXLA_0: A006 2038h, NXLA_1: A006 2078h, NXLA_2: A006 20B8h, NXLA_3: A006 20F8h,
NXLA_4: A006 2138h, NXLA_5: A006 2178h, NXLA_6: A006 21B8h, NXLA_7: A006 21F8h,
NXLA_8: A006 2438h, NXLA_9: A006 2478h, NXLA_10: A006 24B8h, NXLA_11: A006 24F8h,
NXLA_12: A006 2538h, NXLA_13: A006 2578h, NXLA_14: A006 25B8h, NXLA_15: A006 25F8h
DMACAa1
NXLA_0: A006 3038h, NXLA_1: A006 3078h, NXLA_2: A006 30B8h, NXLA_3: A006 30F8h,
NXLA_4: A006 3138h, NXLA_5: A006 3178h, NXLA_6: A006 31B8h, NXLA_7: A006 31F8h,
NXLA_8: A006 3438h, NXLA_9: A006 3478h, NXLA_10: A006 34B8h, NXLA_11: A006 34F8h,
NXLA_12: A006 3538h, NXLA_13: A006 3578h, NXLA_14: A006 35B8h, NXLA_15: A006 35F8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	NXLA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	NXLA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	NXLA[31:0]	Next Link Address	Sets the address of the link destination. Because the two lower-order bits are fixed to 0, only word-aligned addresses can be set.	R/W
b31 to b2				R/W

14.2.15 Current Link Address Register n (CRLA_n)

The CRLA_n register sets the link address of DMA channel n (n = 15 to 0).

Address(es): DMACAa0
CRLA_0: A006 203Ch, CRLA_1: A006 207Ch, CRLA_2: A006 20BCh, CRLA_3: A006 20FCh,
CRLA_4: A006 213Ch, CRLA_5: A006 217Ch, CRLA_6: A006 21BCh, CRLA_7: A006 21FCh,
CRLA_8: A006 243Ch, CRLA_9: A006 247Ch, CRLA_10: A006 24BCh, CRLA_11: A006 24FCh,
CRLA_12: A006 253Ch, CRLA_13: A006 257Ch, CRLA_14: A006 25BCh, CRLA_15: A006 25FCh
DMACAa1
CRLA_0: A006 303Ch, CRLA_1: A006 307Ch, CRLA_2: A006 30BCh, CRLA_3: A006 30FCh,
CRLA_4: A006 313Ch, CRLA_5: A006 317Ch, CRLA_6: A006 31BCh, CRLA_7: A006 31FCh,
CRLA_8: A006 343Ch, CRLA_9: A006 347Ch, CRLA_10: A006 34BCh, CRLA_11: A006 34FCh,
CRLA_12: A006 353Ch, CRLA_13: A006 357Ch, CRLA_14: A006 35BCh, CRLA_15: A006 35FCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRLA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRLA[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRLA[31:0]	Current Link Address	Indicates the address of the descriptor which is being executed.	R

14.2.16 Source Continuous Register n (SCNT_n)

The SCNT_n register sets the space size for continuous access during read access to the DMA transfer source (n = 15 to 0).

Use this register together with the SSKP_n register (see Figure 14.1).

When setting this register, set the SDS[3] bit in the CHCFG_n register to 1.

Address(es): DMACAa0
SCNT_0: A006 2200h, SCNT_1: A006 2220h, SCNT_2: A006 2240h, SCNT_3: A006 2260h,
SCNT_4: A006 2280h, SCNT_5: A006 22A0h, SCNT_6: A006 22C0h, SCNT_7: A006 22E0h,
SCNT_8: A006 2600h, SCNT_9: A006 2620h, SCNT_10: A006 2640h, SCNT_11: A006 2660h,
SCNT_12: A006 2680h, SCNT_13: A006 26A0h, SCNT_14: A006 26C0h, SCNT_15: A006 26E0h
DMACAa1
SCNT_0: A006 3200h, SCNT_1: A006 3220h, SCNT_2: A006 3240h, SCNT_3: A006 3260h,
SCNT_4: A006 3280h, SCNT_5: A006 32A0h, SCNT_6: A006 32C0h, SCNT_7: A006 32E0h,
SCNT_8: A006 3600h, SCNT_9: A006 3620h, SCNT_10: A006 3640h, SCNT_11: A006 3660h,
SCNT_12: A006 3680h, SCNT_13: A006 36A0h, SCNT_14: A006 36C0h, SCNT_15: A006 36E0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SCNT[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SCNT[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	SCNT[31:0]	Source Continuous Access Size	Sets the size of the continuous access space during read access to the DMA transfer source in bytes.	R/W

When performing a skip transfer on the transfer source, do not set the SAD bit in the CHCFG_n register to 1 (fixed). In addition, do not perform a skip transfer when this register is set to 0000 0000h.

14.2.17 Source Skip Register n (SSKP_n)

The SSKP_n register sets the skip amount during read access to the DMA transfer source.
During read access to the DMA transfer source, after data of the size set by using the SCNT_n register is accessed, the next DMA transfer source address of the size set by using this register is skipped (n = 15 to 0).
Use this register together with the SCNT_n register (see Figure 14.1).
When setting this register, set the SDS[3] bit in the CHCFG_n register to 1.

Address(es): DMACAa0
SSKP_0: A006 2204h, SSKP_1: A006 2224h, SSKP_2: A006 2244h, SSKP_3: A006 2264h,
SSKP_4: A006 2284h, SSKP_5: A006 22A4h, SSKP_6: A006 22C4h, SSKP_7: A00622E4h,
SSKP_8: A006 2604h, SSKP_9: A006 2624h, SSKP_10: A006 2644h, SSKP_11: A006 2664h,
SSKP_12: A006 2684h, SSKP_13: A006 26A4h, SSKP_14: A006 26C4h, SSKP_15: A006 26E4h
DMACAa1
SSKP_0: A006 3204h, SSKP_1: A006 3224h, SSKP_2: A006 3244h, SSKP_3: A006 3264h,
SSKP_4: A006 3284h, SSKP_5: A006 32A4h, SSKP_6: A006 32C4h, SSKP_7: A006 32E4h,
SSKP_8: A006 3604h, SSKP_9: A006 3624h, SSKP_10: A006 3644h, SSKP_11: A006 3664h,
SSKP_12: A006 3684h, SSKP_13: A006 36A4h, SSKP_14: A006 36C4h, SSKP_15: A006 36E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SSKP[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SSKP[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	SSKP[31:0]	Source Skip Size	Sets the skip amount during read access to the DMA transfer source in bytes.	R/W

When performing a skip transfer on the transfer source, do not set the SAD bit in the CHCFG_n register to 1 (fixed).

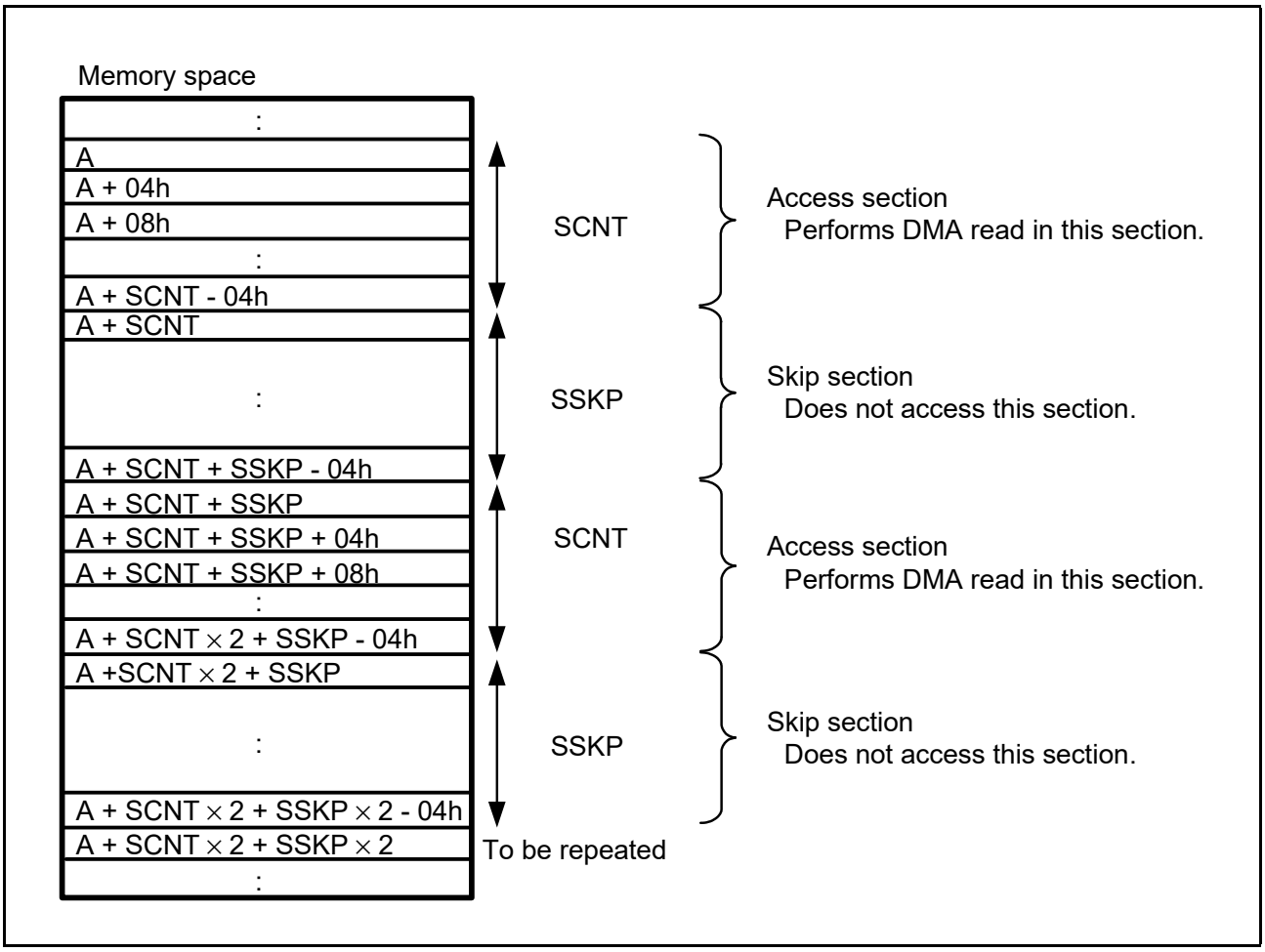


Figure 14.1 Relationship between SSKP and SCNT

Regardless of the source address and the setting value of the RDS field in the CHCFG_n register, the values for SCNT and SSKP can be set. DMACAa accesses in the size set in the SDS field in the CHCFG_n register, and acquires the buffer of the valid data only.

14.2.18 Destination Continuous Register n (DCNT_n)

The DCNT_n register sets the space size for continuous access during write access to the DMA transfer destination (n = 15 to 0).

Use this register together with the DSKP_n register (see Figure 14.2).

To set this register, set the DDS[3] bit in the CHCFG_n register to 1.

Address(es): DMACAa0
DCNT_0: A006 2208h, DCNT_1: A006 2228h, DCNT_2: A006 2248h, DCNT_3: A006 2268h,
DCNT_4: A006 2288h, DCNT_5: A006 22A8h, DCNT_6: A006 22C8h, DCNT_7: A006 22E8h,
DCNT_8: A006 2608h, DCNT_9: A006 2628h, DCNT_10: A006 2648h, DCNT_11: A006 2668h,
DCNT_12: A006 2688h, DCNT_13: A006 26A8h, DCNT_14: A006 26C8h, DCNT_15: A006 26E8h
DMACAa1
DCNT_0: A006 3208h, DCNT_1: A006 3228h, DCNT_2: A006 3248h, DCNT_3: A006 3268h,
DCNT_4: A006 3288h, DCNT_5: A006 32A8h, DCNT_6: A006 32C8h, DCNT_7: A006 32E8h,
DCNT_8: A006 3608h, DCNT_9: A006 3628h, DCNT_10: A006 3648h, DCNT_11: A006 3668h,
DCNT_12: A006 3688h, DCNT_13: A006 36A8h, DCNT_14: A006 36C8h, DCNT_15: A006 36E8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DCNT[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DCNT[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	DCNT[31:0]	Destination Continuous Access Size	Sets the size of the continuous access space during write access to the DMA transfer destination in bytes.	R/W

When performing a skip transfer on the destination, do not set the DAD bit in the CHCFG_n register to 1 (fixed).

In addition, do not perform a skip transfer when this register is set to 0000 0000h.

14.2.19 Destination Skip Register n (DSKP_n)

The DSKP_n register sets the skip amount during write access to the DMA transfer destination.
During write access to the DMA transfer destination, accesses data of the size set with the DCNT_n register, and then skips the number of the next DMA transfer destination addresses set with this register (n = 15 to 0).
Use this register together with the DCNT_n register (see Figure 14.2).
To set this register, set the DDS[3] bit in the CHCFG_n register to 1.

Address(es): DMACAa0
DSKP_0: A006 220Ch, DSKP_1: A006 222Ch, DSKP_2: A006 224Ch, DSKP_3: A006 226Ch,
DSKP_4: A006 228Ch, DSKP_5: A006 22ACh, DSKP_6: A006 22CCh, DSKP_7: A006 22ECh,
DSKP_8: A006 260Ch, DSKP_9: A006 262Ch, DSKP_10: A006 264Ch, DSKP_11: A006 266Ch,
DSKP_12: A006 268Ch, DSKP_13: A006 26ACh, DSKP_14: A006 26CCh, DSKP_15: A006 26ECh
DMACAa1
DSKP_0: A006 320Ch, DSKP_1: A006 322Ch, DSKP_2: A006 324Ch, DSKP_3: A006 326Ch,
DSKP_4: A006 328Ch, DSKP_5: A006 32ACh, DSKP_6: A006 32CCh, DSKP_7: A006 32ECh,
DSKP_8: A006 360Ch, DSKP_9: A006 362Ch, DSKP_10: A006 364Ch, DSKP_11: A006 366Ch,
DSKP_12: A006 368Ch, DSKP_13: A006 36ACh, DSKP_14: A006 36CCh, DSKP_15: A006 36ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DSKP[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DSKP[31:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	DSKP[31:0]	Destination Skip Size	Sets the skip amount during write access to the DMA transfer destination in bytes.	R/W

To perform skip transfer on the destination, do not set the DAD bit in the CHCFG_n register to 1 (fixed).

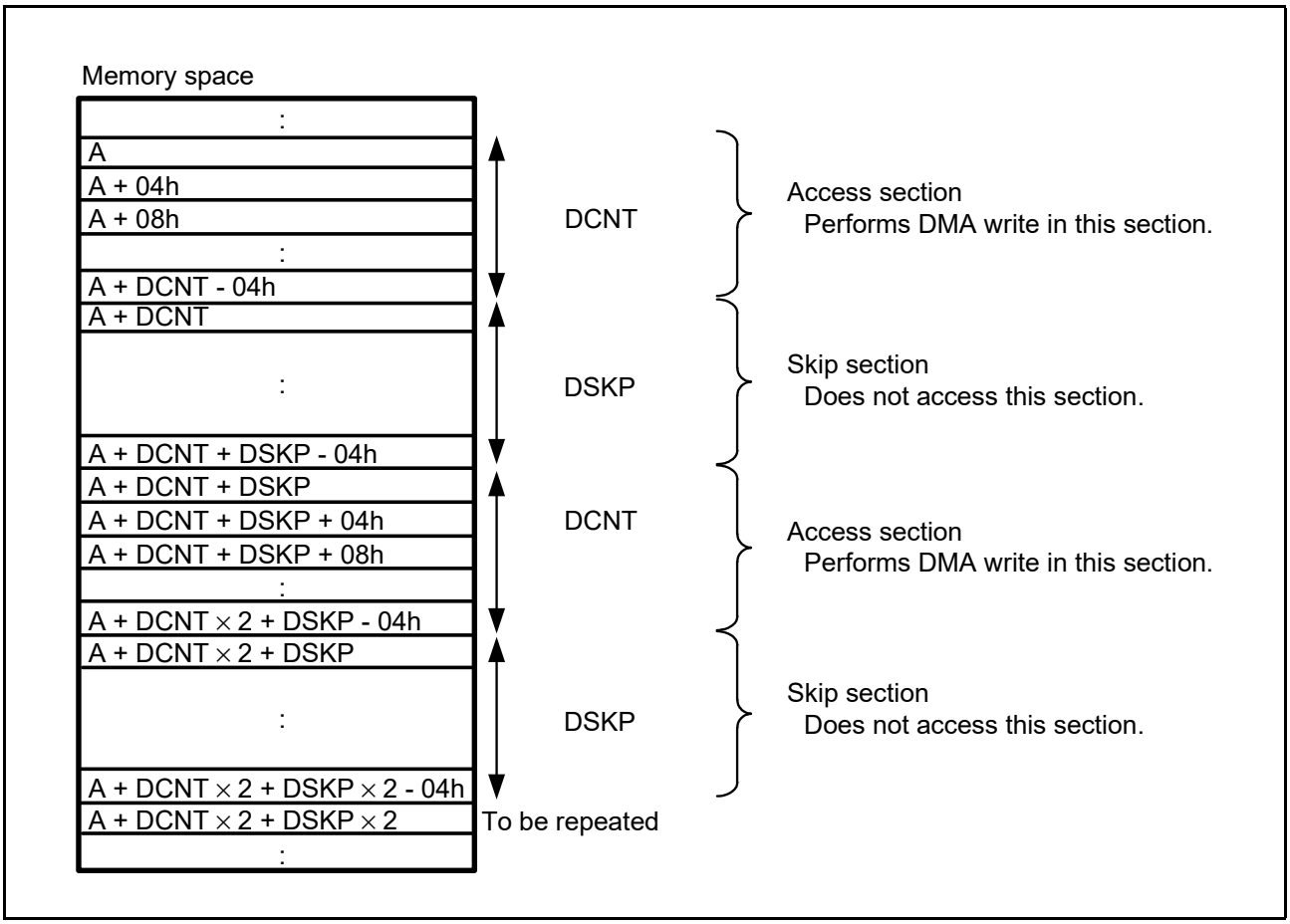


Figure 14.2 Relationship between DSKP and DCNT

Regardless of the destination address and the setting value of the DDS field in the CHCFG_n register, the values for DCNT and DSKP can be set. DMACAa performs write access only to the specified space by the size equal to or smaller than the value set in the DDS field in the CHCFG_n register.

14.2.20 DMA Control Register (DCTRL_X (X = A or B))

The DCTRL_X register sets the arbitration between channels in all channels (DCTRL_A = channels 0 to 7, DCTRL_B = channels 8 to 15).

Address(es): DMACAa0
DCTRL_A: A006 2300h, DCTRL_B: A006 2700h
DMACAa1
DCTRL_A: A006 3300h, DCTRL_B: A006 3700h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PR	Priority Control Select	Sets the transfer priority control mode (see section 14.3.3, DMA Channel Priority Control). 0: Fixed priority mode 1: Round-robin mode	R/W
b31 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

14.2.21 Descriptor Interval Register n (DSCITVL_X (X = A or B))

The DSCITVL_X register sets the descriptor read interval in all channels (DSCITVL_A = channels 0 to 7, DSCITVL_B = channels 8 to 15).

By setting the DRRP bit in the CHCFG_n register to 1, the descriptor continues to read the descriptor until it reaches LV = 1. This register sets the read interval.

Address(es): DMACAa0
DSCITVL_A: A006 2304h, DSCITVL_B: A006 2704h
DMACAa1
DSCITVL_A: A006 3304h, DSCITVL_B: A006 3704h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DITVL								—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W
b15 to b8	DITVL	Descriptor Interval	Sets the descriptor read interval. The descriptor is read again in the interval of (DITVL × 256) cycles. The descriptor read interval will be (the set value in the DITVL bit × 256 × ICLK) cycles.	R/W
b31 to b16	—	Reserved	These bits are always read as 0. The write value should be 0.	R/W

14.2.22 DMA Status EN Register (DST_EN_X (X = A or B))

The DST_EN_X register indicates the status for the EN bit of all channels (DST_EN_A = channels 0 to 7, DST_EN_B = channels 8 to 15).

Writing data to this register does not change the value of each bit.

Address(es): DMACAa0
DST_EN_A: A006 2310h, DST_EN_B: A006 2710h
DMACAa1
DST_EN_A: A006 3310h, DST_EN_B: A006 3710h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	EN 7/15	EN 6/14	EN 5/13	EN 4/12	EN 3/11	EN 2/10	EN 1/9	EN 0/8
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	EN0/8	Channel 0/8EN	Indicates the status for the EN bit of DMA channel 0/8.	R
b1	EN1/9	Channel 1/9EN	Indicates the status for the EN bit of DMA channel 1/9.	R
b2	EN2/10	Channel 2/10EN	Indicates the status for the EN bit of DMA channel 2/10.	R
b3	EN3/11	Channel 3/11EN	Indicates the status for the EN bit of DMA channel 3/11.	R
b4	EN4/12	Channel 4/12EN	Indicates the status for the EN bit of DMA channel 4/12.	R
b5	EN5/13	Channel 5/13EN	Indicates the status for the EN bit of DMA channel 5/13.	R
b6	EN6/14	Channel 6/14EN	Indicates the status for the EN bit of DMA channel 6/14.	R
b7	EN7/15	Channel 7/15EN	Indicates the status for the EN bit of DMA channel 7/15.	R
b31 to b8	—	Reserved	These bits are always read as 0.	R

14.2.23 DMA Status ER Register (DST_ER_X (X = A or B))

The DST_ER_X register indicates the status for the ER bit of all channels (DST_ER_A = channels 0 to 7, DST_ER_B = channels 8 to 15).

Writing data to this register does not change the value of each bit.

Address(es): DMACAa0
DST_ER_A: A006 2314h, DST_ER_B: A006 2714h
DMACAa1
DST_ER_A: A006 3314h, DST_ER_B: A006 3714h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ER 7/15	ER 6/14	ER 5/13	ER 4/12	ER 3/11	ER 2/10	ER 1/9	ER 0/8
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ER0/8	Channel 0/8ER	Indicates the status for the ER bit of DMA channel 0/8.	R
b1	ER1/9	Channel 1/9ER	Indicates the status for the ER bit of DMA channel 1/9.	R
b2	ER2/10	Channel 2/10ER	Indicates the status for the ER bit of DMA channel 2/10.	R
b3	ER3/11	Channel 3/11ER	Indicates the status for the ER bit of DMA channel 3/11.	R
b4	ER4/12	Channel 4/12ER	Indicates the status for the ER bit of DMA channel 4/12.	R
b5	ER5/13	Channel 5/13ER	Indicates the status for the ER bit of DMA channel 5/13.	R
b6	ER6/14	Channel 6/14ER	Indicates the status for the ER bit of DMA channel 6/14.	R
b7	ER7/15	Channel 7/15ER	Indicates the status for the ER bit of DMA channel 7/15.	R
b31 to b8	—	Reserved	These bits are always read as 0.	R

14.2.24 DMA Status END Register (DST_END_X (X = A or B))

The DST_END_X register indicates the status for the END bit of all channels (DST_END_A = channels 0 to 7, DST_END_B = channels 8 to 15). Writing data to this register does not change the value of each bit.

Address(es): DMACAa0
DST_END_A: A006 2318h, DST_END_B: A006 2718h
DMACAa1
DST_END_A: A006 3318h, DST_END_B: A006 3718h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	END 7/15	END 6/14	END 5/13	END 4/12	END 3/11	END 2/10	END 1/9	END 0/8
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	END0/8	Channel 0/8END	Indicates the status for the END bit of DMA channel 0/8.	R
b1	END1/9	Channel 1/9END	Indicates the status for the END bit of DMA channel 1/9.	R
b2	END2/10	Channel 2/10END	Indicates the status for the END bit of DMA channel 2/10.	R
b3	END3/11	Channel 3/11END	Indicates the status for the END bit of DMA channel 3/11.	R
b4	END4/12	Channel 4/12END	Indicates the status for the END bit of DMA channel 4/12.	R
b5	END5/13	Channel 5/13END	Indicates the status for the END bit of DMA channel 5/13.	R
b6	END6/14	Channel 6/14END	Indicates the status for the END bit of DMA channel 6/14.	R
b7	END7/15	Channel 7/15END	Indicates the status for the END bit of DMA channel 7/15.	R
b31 to b8	—	Reserved	These bits are always read as 0.	R

14.2.25 DMA Status SUS Register (DST_SUS_X (X = A or B))

The DST_SUS_X register indicates the status for the SUS bit of all channels (DST_SUS_A = channels 0 to 7, DST_SUS_B = channels 8 to 15). Writing data to this register does not change the value of each bit.

Address(es): DMACAa0
DST_SUS_A: A006 2320h, DST_SUS_B: A006 2720h
DMACAa1
DST_SUS_A: A006 3320h, DST_SUS_B: A006 3720h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	SUS 7/15	SUS 6/14	SUS 5/13	SUS 4/12	SUS 3/11	SUS 2/10	SUS 1/9	SUS 0/8
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SUS0/8	Channel 0/8SUS	Indicates the status for the SUS bit of DMA channel 0/8.	R
b1	SUS1/9	Channel 1/9SUS	Indicates the status for the SUS bit of DMA channel 1/9.	R
b2	SUS2/10	Channel 2/10SUS	Indicates the status for the SUS bit of DMA channel 2/10.	R
b3	SUS3/11	Channel 3/11SUS	Indicates the status for the SUS bit of DMA channel 3/11.	R
b4	SUS4/12	Channel 4/12SUS	Indicates the status for the SUS bit of DMA channel 4/12.	R
b5	SUS5/13	Channel 5/13SUS	Indicates the status for the SUS bit of DMA channel 5/13.	R
b6	SUS6/14	Channel 6/14SUS	Indicates the status for the SUS bit of DMA channel 6/14.	R
b7	SUS7/15	Channel 7/15SUS	Indicates the status for the SUS bit of DMA channel 7/15.	R
b31 to b8	—	Reserved	These bits are always read as 0.	R

14.3 Operation

14.3.1 DMA Mode

With the DMS bit in the CHCFG_n register, DMA mode can be switched between register mode and link mode.

Table 14.2 DMA Mode Setting

DMS (CHCFG_n)	Description	Applications
0	Register mode	With the values set for the next register set, performs a DMA transfer.
1	Link mode	Accesses the descriptor area, and performs a DMA transfer with the value set for the descriptor. Repeats descriptor read and DMA transfer unless you set the descriptor or use the control register to stop them.

14.3.1.1 Register Mode

In register mode, you can perform a DMA transfer with the value set in the internal register.
You can set two sets (Next0 Register Set and Next1 Register Set) of transfer source addresses, the transfer destination addresses, and the numbers of transfer bytes. You can select the next register set to perform a transfer, or transfer two next register sets successively.

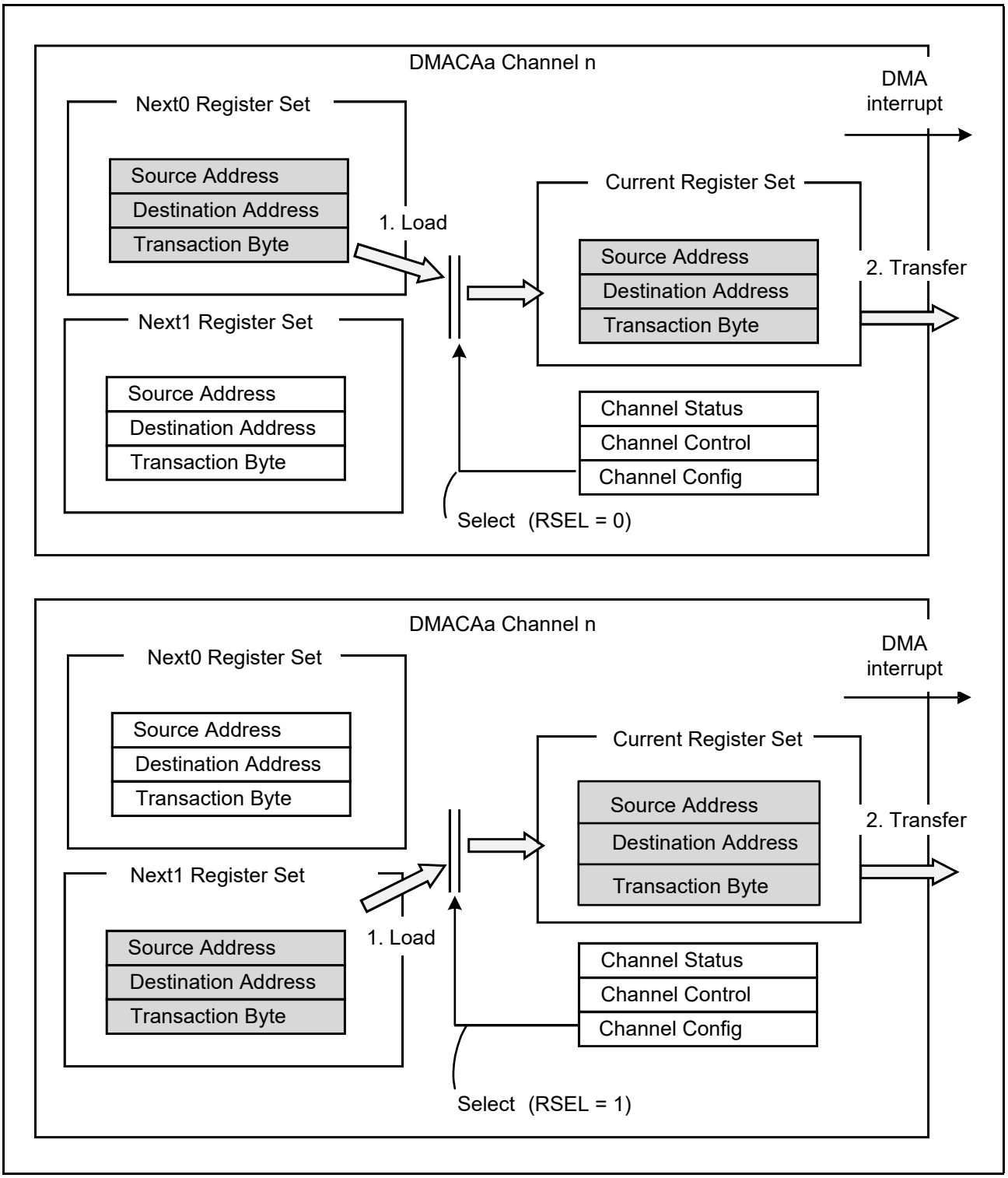


Figure 14.3 Overview of Register Normal Mode

Figure 14.3 illustrates operation when Next0 Register Set is executed (above), and when Next1 Register Set is executed (below).

(1) Operation Flow in Register Mode

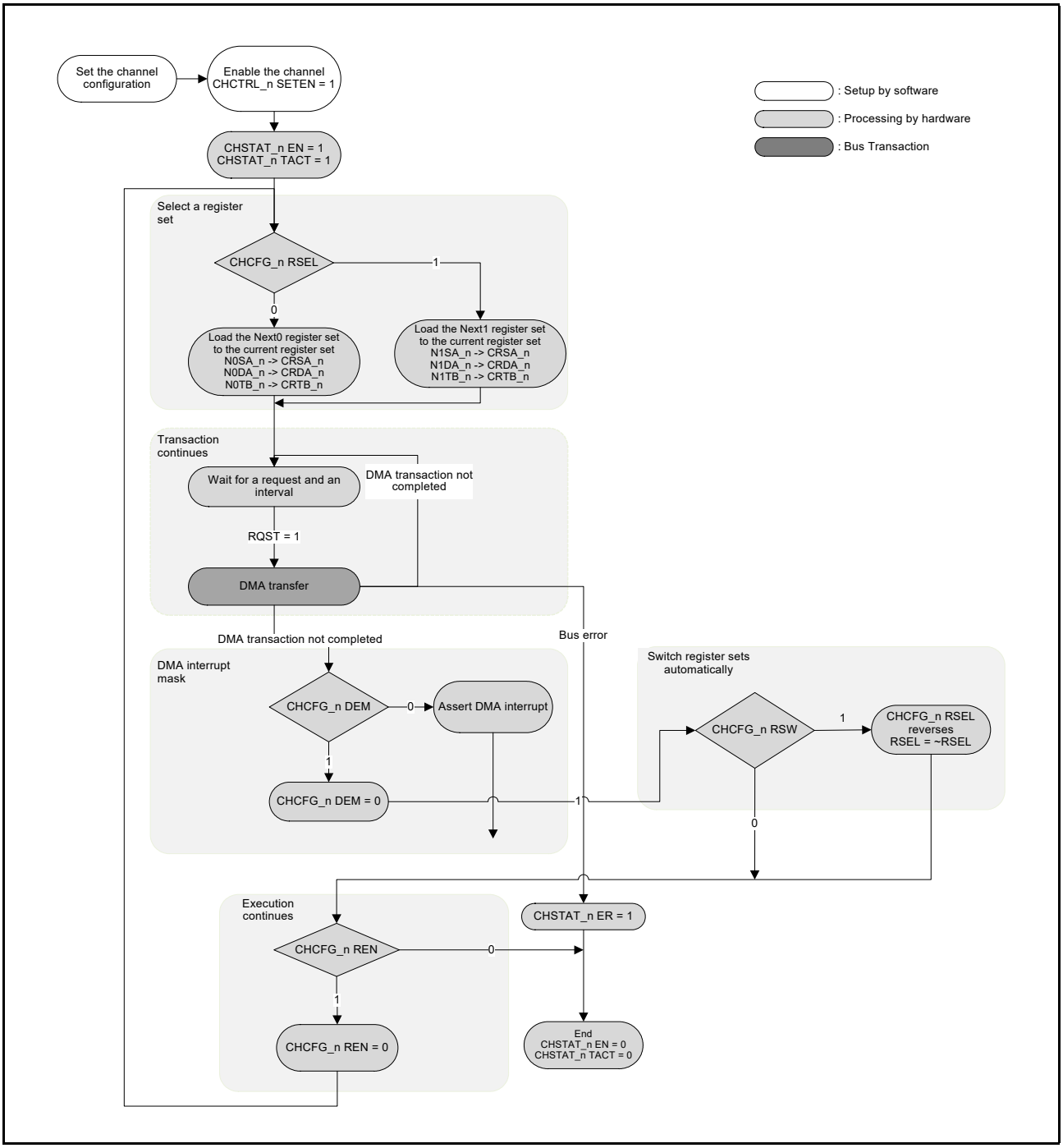


Figure 14.4 Register Mode Flow

<Register Mode Flow Description>

1. Channel Setting
   Set Next0 or Next1 Register Set (the transfer destination address, the transfer source address, and the total number of transfer bytes).
   In addition, use setting registers CHCTRL_n or CHCFG_n for each channel to set detection method of the DMA transfer request and the amount of data for a transfer.
2. Selecting the register set
   When 1 is written to the SETEN bit in the CHCTRL_n register, the EN bit and the TACT bit in the CHSTAT_n register are set to 1, and the setting value of the next register set selected by the RSEL bit in the CHCFG_n register is loaded to the current register set.
3. DMA Transfer
   According to the set value, a DMA transfer is performed. For details on the transfer, see from section 14.3.2, Transfer Mode to section 14.3.10, Aborting a Transfer.
4. DMA Transfer Completion Mask
   According to the value set for the DEM bit in the CHCFG_n register, the DMA transfer completion interrupt is masked. When DEM = 1, the DMA transfer completion interrupt is masked. Besides, immediately after the DMA transfer completion interrupt conditions are satisfied, the DEM bit is cleared to 0 automatically.
5. Switching Register Sets Automatically
   According to the value set for the RSW bit in the CHCFG_n register, the register set is switched to the other next register set.
6. Successive Execution
   According to the value set for the REN bit in the CHCFG_n register, DMA transfers are performed successively. When REN = 0, the EN bit and the TACT bit in the CHSTAT_n register are cleared to 0, and DMACAa stops operation. When REN = 1, the DMA transfer is continued. Besides, immediately after the conditions for performing DMA transfer again are satisfied with the REN bit, the REN bit is cleared to 0 automatically.

(2) Setting the Register Mode

- Register mode settings
  Select the register set to execute.

Table 14.3 Register Mode Settings

DMS (CHCFG_n)	RSEL (CHCFG_n)	Description
0	0	Executes Next0 Register Set.
	1	Executes Next1 Register Set.

- DMA transfer completion interrupt mask settings
  DMA transfer completion interrupts can be masked.

Table 14.4 DMA Transfer Completion Interrupt Mask Settings

DEM (CHCFG_n)	Description
0	When a DMA transfer completes, the DMA transfer completion interrupt is generated.
1	Even if a DMA transfer completes, no DMA transfer completion interrupt is generated. When a DMA transfer completes, the DEM bit is cleared to 0 automatically.

- Settings for executing register sets automatically
  After a DMA transfer, another DMA transfer can be performed successively.

Table 14.5 Settings for Automatic Execution of Register Sets

REN (CHCFG_n)	Operation	Remarks
0	When a DMA transfer of the register set that is set for the RSEL bit completes, the EN bit is cleared to 0, and the DMA operation ends.	Set this value if you want to perform a DMA transfer once.
1	After a DMA transfer completes, a DMA transfer is performed for the register set that was selected to be performed successively. When the successive transfer is performed, the REN bit is cleared to 0.	Set this register if you want to perform register sets successively.

- Settings for switching register sets automatically
  When a DMA transfer completes, the register set can be switched to the next register set.

Table 14.6 Settings for Switching Register Sets Automatically

RSW (CHCFG_n)	Operation	Remarks
0	When a DMA transfer completes, register sets are not switched.	Set this value when you want to use one register set only.
1	When a DMA transfer completes, the RSEL bit is reversed automatically, and the other register set is selected.	Set this value if you want to switch register sets.

(3) Register Mode Setting Examples

- Using the Next0 register set only

Table 14.7 Register Mode Setting Example 1

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (register mode)	0 (Next0)	0 (Do not mask)	0 (Do not switch)	0 (Continuous execution not in progress)

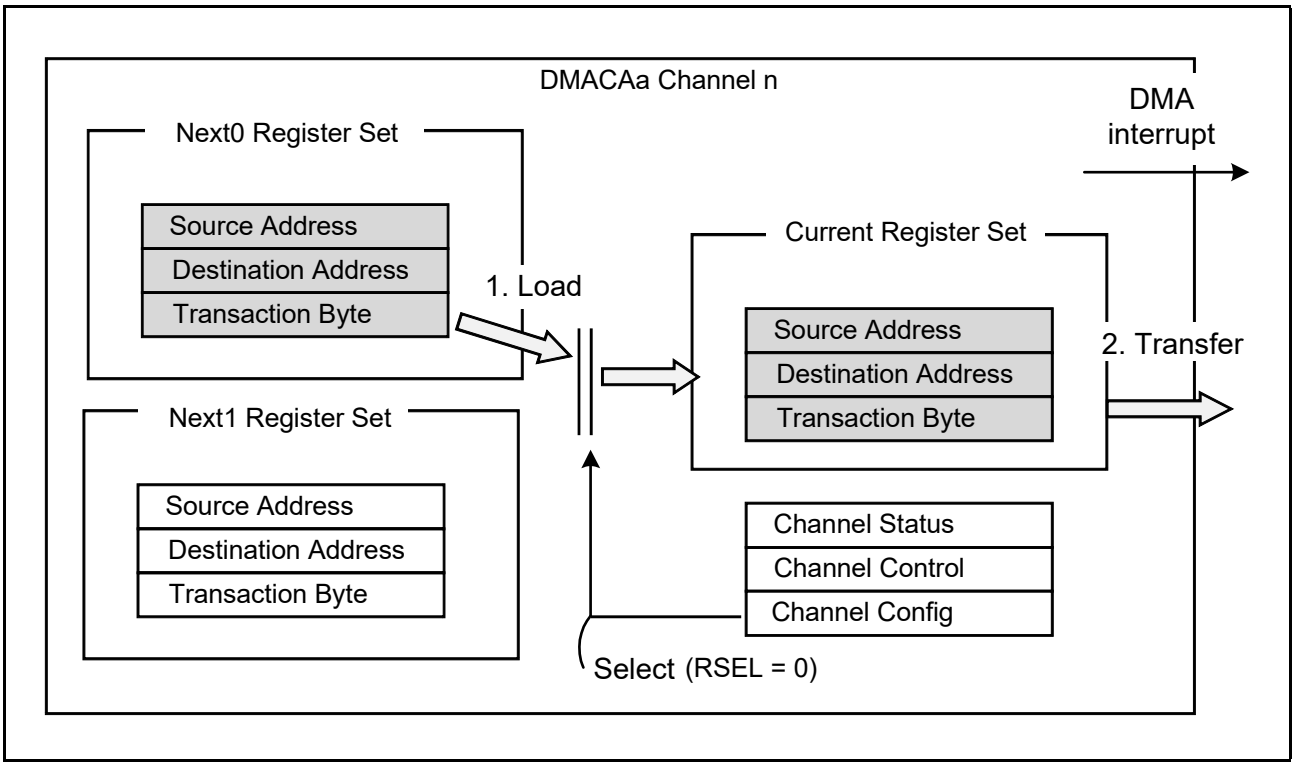


Figure 14.5 Register Mode Setting Example 1

1. Writing 1 to the SETEN bit in the CHCTRL_n register sets 1 to the EN bit in the CHSTAT_n register, and loads Next0 Register Set to the current register set.
2. According to the values set for the current register set and the channel register set, a DMA transfer is performed.
3. Because the DEM bit in the CHCFG_n register is 0, a DMA transfer completion interrupt is generated when a DMA transfer completes.
4. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0, and the operation ends.

- Using two register sets successively

Table 14.8 Register Mode Setting Example 2

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (Register mode)	0 (Next0)	1 (Mask)	1 (Switch)	1 (Continuous execution not in progress)

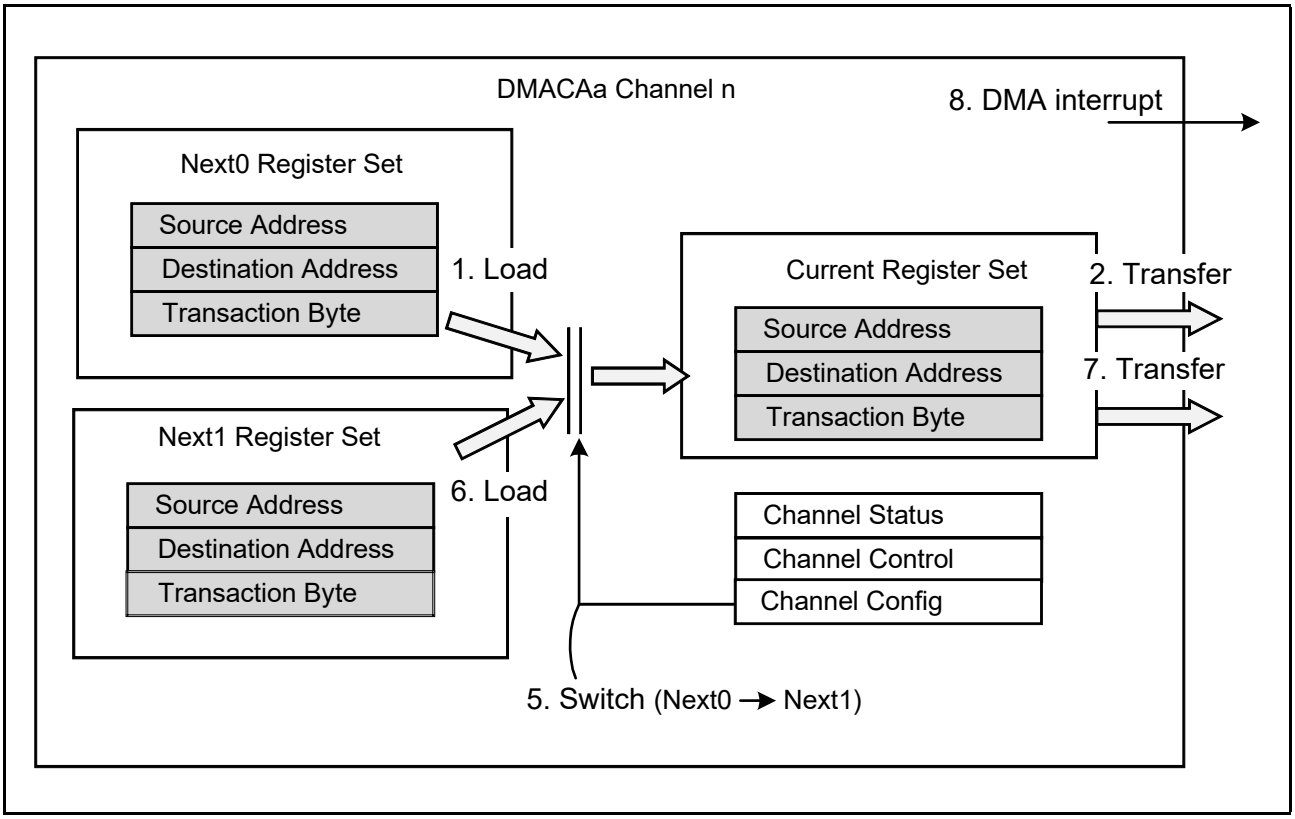


Figure 14.6 Register Mode Setting Example 2

1. Writing 1 to the SETEN bit in the CHCTRL_n register sets 1 to the EN bit in the CHSTAT_n register, and loads Next0 Register Set to the current register set.
2. According to the values set for the current register set and the channel register set, a DMA transfer is performed.
3. Because the DEM bit in the CHCFG_n register is 1, when the DMA transfer completes, no DMA transfer completion interrupt is generated. In addition, the DEM bit is cleared to 0 automatically.
4. Because the REN bit in the CHCFG_n register is 1, DMA transfers are performed successively. In addition, the REN bit is cleared to 0.
5. Because the RSW bit in the CHCFG_n register is 1, the register set to be executed next is switched (RSEL = 0 -> 1).
6. Loads Next1 Register Set to Current Register Set.
7. According to the values for Current Register Set and Channel Register Set, DMA transfers are performed.
8. Because the DEM bit in the CHCFG_n register is 0, a DMA transfer completion interrupt is generated when a DMA transfer completes.
9. Because the REN bit in the CHCFG_n register is 0, the EN bit in the CHSTAT_n register is cleared to 0 automatically.

14.3.1.2 Link Mode

In link mode, a DMA transfer is performed by reading a descriptor in the memory area outside the DMAC as the setting value. Within DMACAa, each channel has the next link address (NXLA_n) register and the current link address (CRLA_n) register. Each of them is used to set the address of the descriptor to be executed next, and to indicate the descriptor address of the current DMA transfer respectively.

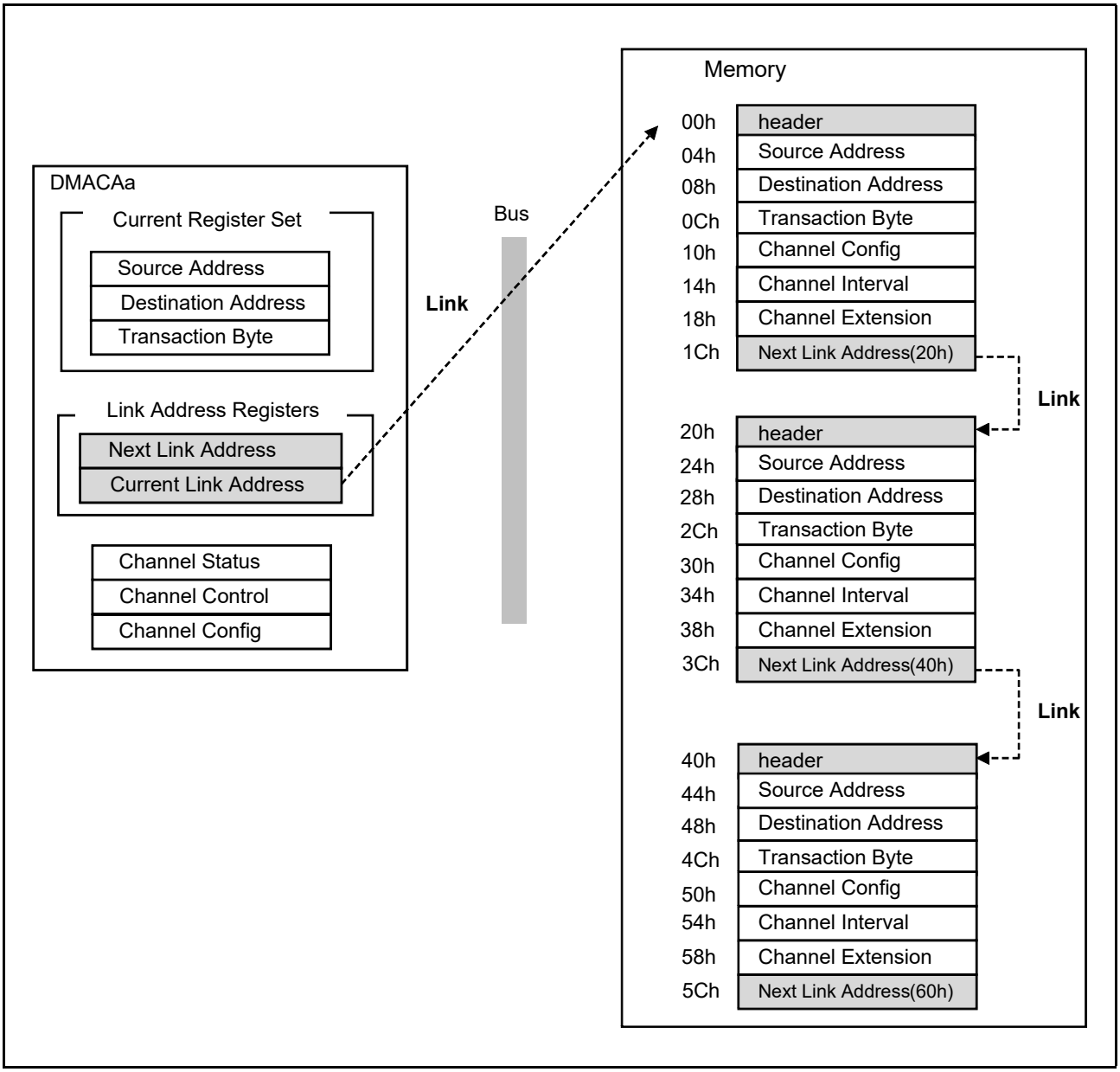


Figure 14.7 Link Mode Overview

(1) Operation flows in link mode

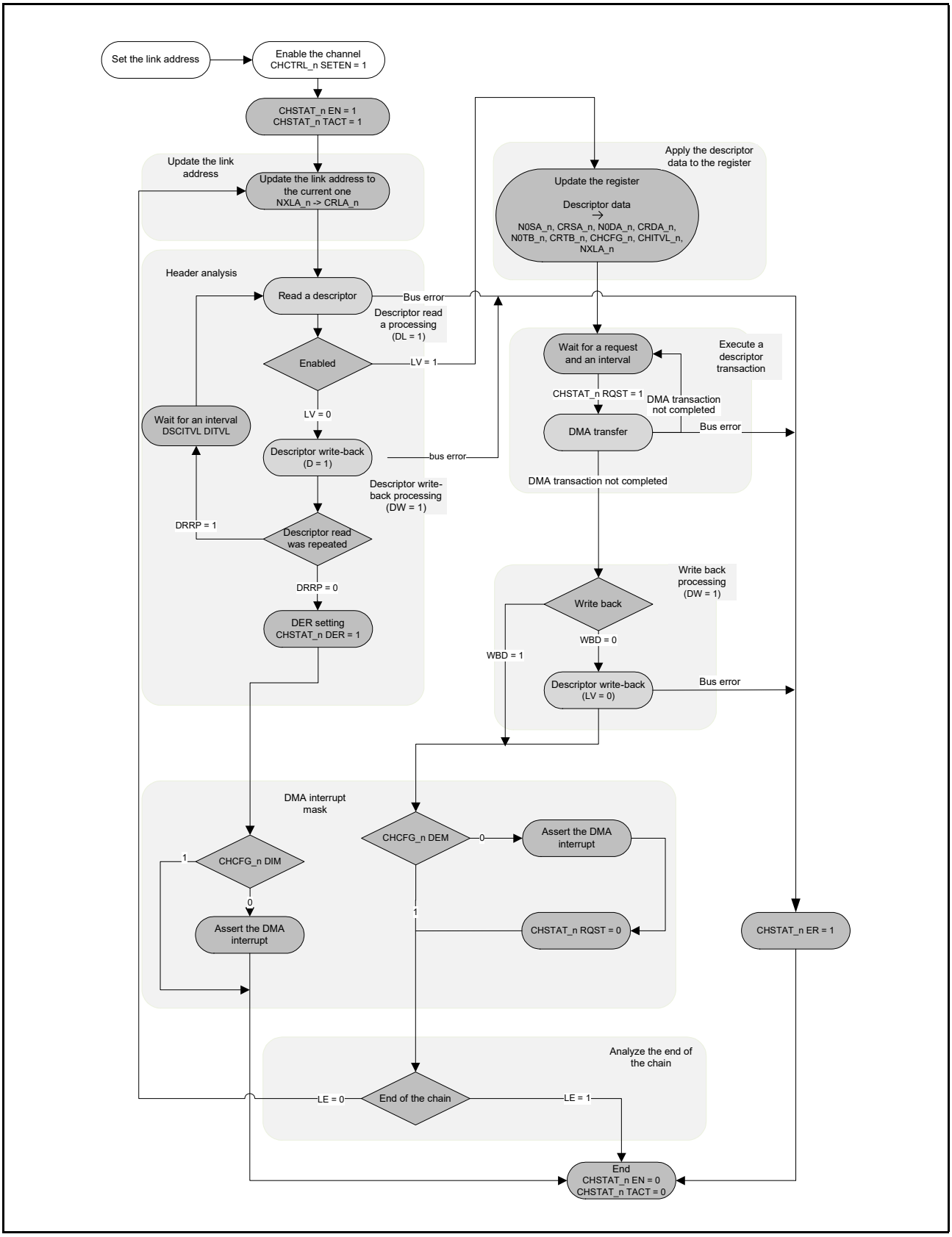


Figure 14.8 Link Mode Operation Flow

<Description of Link Mode Operation Flow>

1. Channel Setting
   Set the start address of the link destination in the NXLA_n register.
2. Updating the Link Address
   Writing 1 to the SETEN bit in the CHCTRL_n register sets 1 to the EN bit and the TACT bit in the CHSTAT_n register, and loads the link address set for the NXLA_n register to the CRLA_n register.
3. Descriptor Read and Header Judgment
   DMACAa starts reading a descriptor, and checks the header contents. When LV = 0, 1 is written back to the D bit of header. After that, when the DRRP bit in the CHCFG_n register = 1, the same descriptor is read again after the number of cycles set in the DSCITVL register elapsed. When DRRP =0, DER in the CHSTAT_n register = 1. This indicates the end state (the EN bit in the CHSTAT_n register = 0, and TACT = 0). At that time, if the DIM bit in the CHCFG_n register is 0, a DMA transfer completion interrupt is generated.
4. Descriptor Error
   When LV = 1, the read descriptor data is loaded to the current register set, and the channel register set. In addition, the next link destination is loaded to the NXLA_n register.
5. DMA Transfer
   According to the set value, a DMA transfer is performed. For details on the transfer, see from section 14.3.2, Transfer Mode to section 14.3.10, Aborting a Transfer.
6. Writing back of Header
   When WBD of header = 0, DMACAa writes LV = 0 back to the header area.
7. DMA Interrupt Mask
   When the DEM bit in the CHCFG_n register is 0, a DMA transfer completion interrupt is generated.
8. Link End Judgment
   When LE of header = 1, the EN bit and the TACT bit in the CHSTAT_n register are cleared to 0, and DMACAa stops operation. When LE = 0, the current register set is updated, and the next descriptor read is started.

(2) Register Settings

- Link mode settings
  When using link mode, set the DMS bit in the CHCFG_n register to 1.

Table 14.9 Link Mode Settings

DMS (CHCFG_n)	Description
1	Operates in link mode. This bit cannot be changed by using a descriptor.

- LINK address settings
  As registers that indicate the link destination, the next link address (NXLA_n) register, and the current link address (CRLA_n) register are available.
  To start link mode, set the link destination in the NXLA_n register.
  The NXLA_n register is updated to the next link after a descriptor is read. In addition, the CRLA_n register indicates the link address of the currently executed descriptor.

Table 14.10 Link Address Register Set

Register	Description
NXLA_n	Sets and indicates the next link destination. Before starting link mode, set the address of the link destination for this register.
CRLA_n	Indicates the currently executed link destination. This register is read-only.

(3) Descriptor settings

DMACAa supports multiple descriptor formats.
To switch formats, use the DSCFM field of bit[31:28] in the first word (header) of a descriptor.
The following table describes the relationship between the value of the DSCFM bit and the descriptor format.

Table 14.11 Descriptor Format

DSCFM	Descriptor Size	Next Link Address	Channel Interval	Channel Config	Transaction Size	Destination Address	Source Address	Header
3	Four words	Y	— (Reload)	— (Reload)	— (header)	Y	Y	Y (with STS)
1	Eight words	Y	Y	Y	Y	Y	Y	Y (without STS)
Other than above	Setting prohibited							

Table 14.12 Description of activation in Table 14.11 Descriptor Format

Field	Availability	Description	Remarks
Header	Y (with STS)	Indicates the STS field of [15:0] for header is enabled. The value set in the STS field is used as the total number of transfer bytes (transaction size).	—
	Y (without STS)	The STS field of [15:0] for header is disabled. Use the transaction size of the descriptor as the total number of transfer bytes.	
Source Address	Y	Specifies the source address.	—
Destination Address	Y	Specifies the destination address.	—
Transaction Size	Y	Specifies the transaction size.	—
	— (header)	Omits the transaction size. Use the value set in the STS field of header as the total number of transfer bytes (transaction size).	Because the STS field is 16 bits, the maximum size you can set is 65,535 bytes.
Channel Config Channel Interval	Y	Specifies Channel Config and Channel Interval.	—
	— (reload)	Omits Channel Config and Channel Interval. Inherits the previous setting values (values of the CHCFG_n and CHITVL_n registers at that time).	—
Next Link Address	Y	Specifies the next descriptor address (next link address) that is read after a DMA transfer of this descriptor.	—

DMACAa interprets data obtained through descriptor read in order. If a value less than eight words is specified in the DSCFM field, place descriptor data marked with Y in Table 14.11, Descriptor Format on memory.

Table 14.13 Descriptor Placement Example

	Address (Link Address + N)							
DSCFM	+1Ch	+18h	+14h	+10h	+0Ch	+08h	+04h	+00h
3h	—	—	—	—	Next Link Address	Destination Address	Source Address	header
1h	Next Link Address	—	Interval	Config	Transaction Byte	Destination Address	Source Address	header

- Header
  As shown below, header indicates descriptor statuses.
  This area is read by DMACAa before starting a DMA transfer in link mode. In addition, after the DMA transfer, the transfer status is written back by DMACAa.

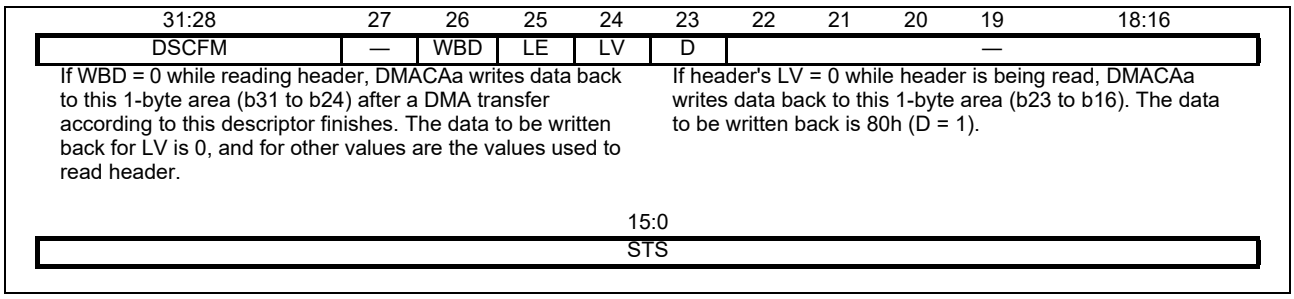


Figure 14.9 Header Area

Table 14.14 Header Area

Bit Position	Bit Name	Description
b15 to b0	STS	When DSCFM = 3, sets the transaction size in bytes. The maximum transfer bytes that can be set is 65,535 bytes. When DSCFM = 3, do not set 0 to the STS bit. If 0 is set, the operation is not guaranteed.
b22 to b16	—	A reserved area. Set 0.
b23	D	Indicates an access error of a descriptor. When LV = 0 while a descriptor is being read, DMACAa writes 1 back to this bit. 0: No descriptor error 1: LV = 0 during descriptor read.
b24	LV	Indicates that this descriptor is enabled. When WBD = 0, DMACAa writes 0 after the DMA transfer written in the descriptor. Set 1 when setting header. 0: The descriptor is disabled. 1: The descriptor is enabled.
b25	LE	Indicates that the link ends during DMA transfer of this descriptor. To indicate the end of the link, set this bit to 1. 0: The link continues. 1: The link ends.
b26	WBD	Masks write back execution of the LV bit. When this bit is 1, DMACAa does not perform write-back operation. 0: The LV bit is written back to 0. 1: The LV bit is not written back.
b27	—	A reserved area. Set 0.
b31 to b28	DSCFM	Specifies the descriptor format (descriptor length, and combination). For details, see Table 14.11.

If you add a descriptor during DMA transfer completion processing (writing back to the descriptor), access of the CPU to set the LV bit to 1, and the access of DMACAa to write 1 back to the D bit might conflict. Because of this, the data that was written first is overwritten with the data that was written later.
To avoid this problem, the byte lane of the D bit and the byte lane of the LV bit are placed differently. DMACAa uses the byte write method for writing back the D bit. Therefore, to set LV = 1, also use the byte write method.

- Setting descriptors other than header
  Data in descriptors except header has the same specifications as an on-chip register. For details on the on-chip register specifications, see section 14.2, Register Descriptions.
  For descriptor setting examples, see section 14.5.3, Setting Example 3 (Link Mode).

- Descriptor areas and DMA transfer areas
  The following figure provides an overview of the descriptor area and the DMA transfer area to which DMACAa accesses.

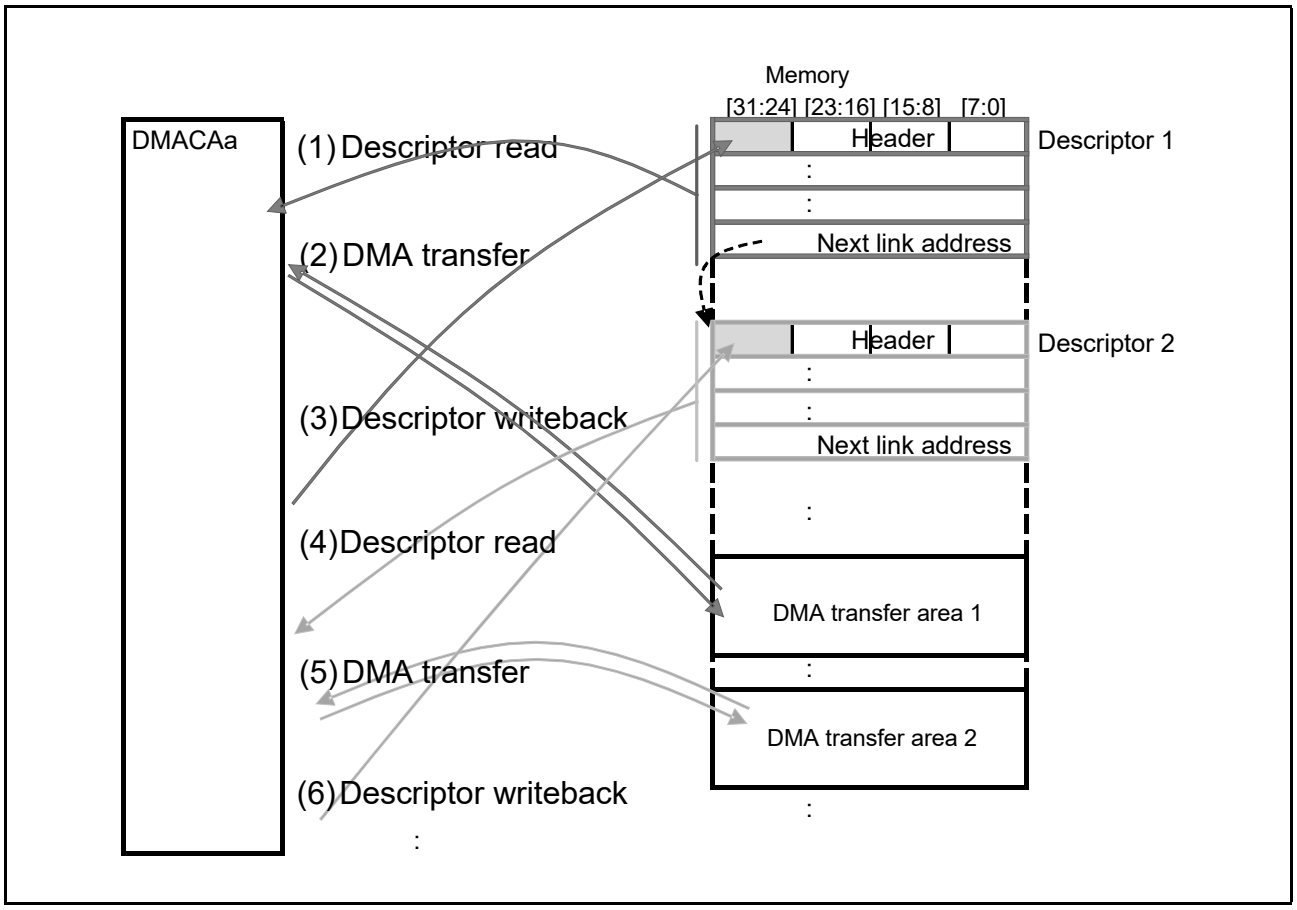


Figure 14.10 Header Area

(1) Descriptor read
Loads the value set for the on-chip NXLA_n register to the CRLA_n register, and then reads the descriptor from the memory space (descriptor1) indicated by the CRLA_n register.

(2) DMA transfer
When the LV bit in header is 1, performs a DMA transfer according to the descriptor information.

(3) Descriptor write-back
After the DMA transfer for the set number of bytes, if the WBD bit in header is 0, the LV bit writes 0 and other fields writes the value read in step 1 as data back to header[31:24] of descriptor1 in bytes.

(4) Descriptor read
If the LE bit in header of the descriptor which was read previously (step 1) is 0, reads the next descriptor from the address (descriptor2) indicated by the next link address in the descriptor.

(5) DMA transfer
When the LV bit in header is 1, performs a DMA transfer according to the descriptor information.

(6) Descriptor write-back
After the DMA transfer for the set number of bytes, if the WBD bit in header is 0, the LV bit writes 0 and other fields writes the value read in step 4 as data back to header[31:24] of descriptor2 in bytes.

Hereafter, repeats steps (4) through (6).

When LE of the header is 1 and WBD = 0, DMA transfer proceeds in accord with the descriptor settings, and 0 is written back to the LV bit of the header, after which operation ends.

When header's LE = 1, and WBD = 1, performs a DMA transfer with the setting, and ends the operation (no write-back is performed).

When header's LV = 0, writes 1 back to the D bit of header. And then, if the DRRP bit in the CHCFG_n register = 1, the descriptor is read again after the interval specified in the DITVL field in the DSCITVL_n register. When DRRP = 0, the operation stops.

- Notes on descriptors
  - In link mode, settings can be changed by reading a descriptor, but the timing for changing settings and for a hardware request cannot be synchronized. Because of this, if you want to issue a hardware request (an external interrupt), set the LVL, HIEN, LOEN, and SEL bits in the CHCFG_n register to 1 before setting the SETEN bit in the CHCTRL_n register. Besides, do not change these bits you set in a descriptor.

  - In a descriptor, the settings of the DMS field in the CHCFG_n register cannot be changed (always link mode). In addition, settings of the REN, RSW, and RSEL fields in the CHCFG_n register can be changed in the descriptor, but that does not affect the operation.

  - DMACAa references the DSCFM field and the LV bit of header to determine if the descriptor is enabled or disabled. Therefore, initialize (DSCFM = 1 or 3, and LV = 1) the memory area equivalent to the LV bit of the DSCFM field before DMACAa accesses it.

  - If you want to set the next descriptor on the memory while reading the transfer settings of DMA (during descriptor read), writes 1 to the LV bit after setting the descriptors after header (source address, destination address, ...next link address). This is to avoid DMA transfer using descriptor values (source address, destination address, and so on) before the setting if descriptor settings by the CPU and descriptor read of DMACAa conflict, and DMACAa's descriptor read interrupts descriptor settings by the CPU.

  - If you want to keep the information written back to the D bit of header, perform byte access to write 1 to the LV bit of header.

(4) LINK configuration examples

In link mode, descriptors can be configured as in the following figure.

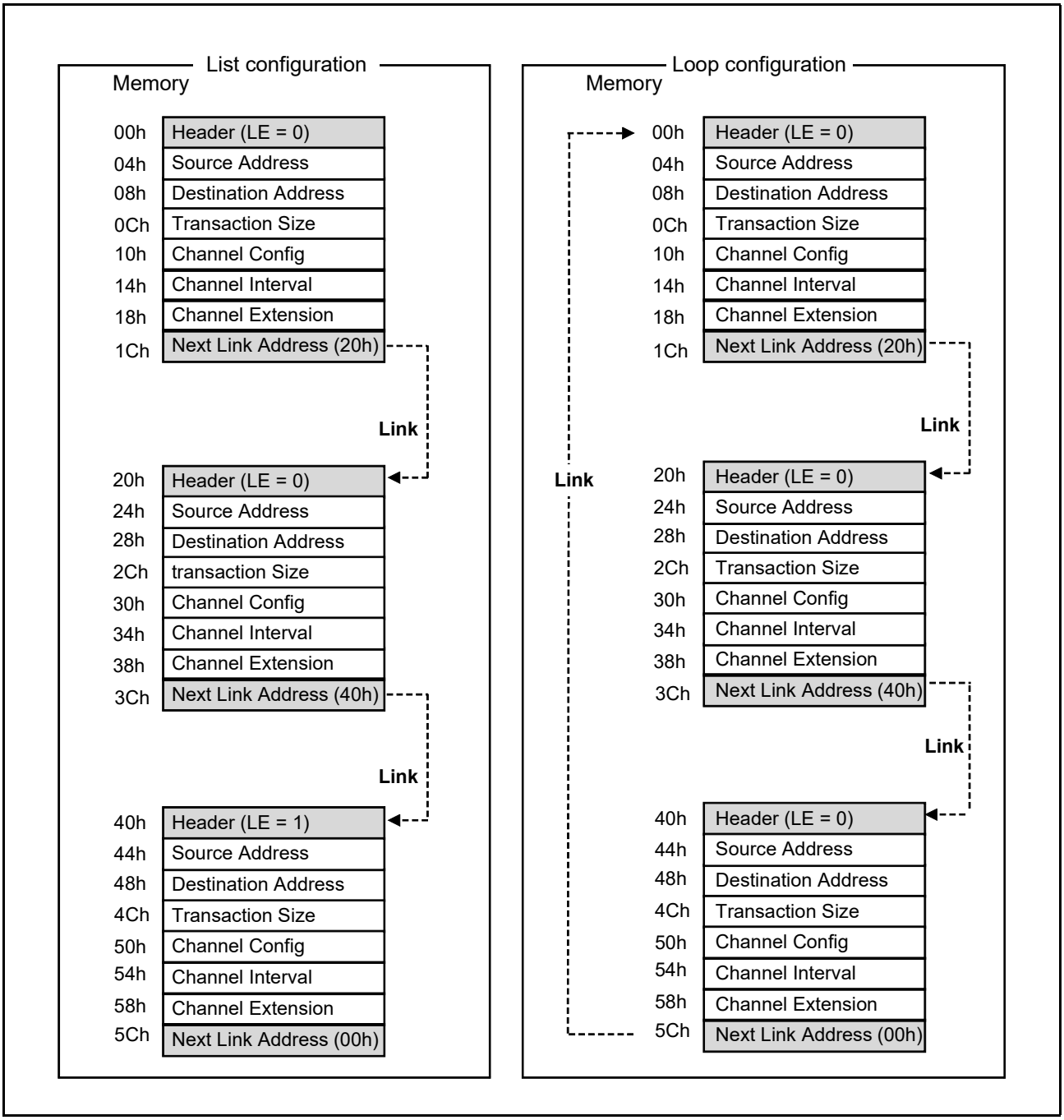


Figure 14.11 Header Area

- List configuration
  Setting 1 to the LE bit in header of the last descriptor ends the link.

- Loop configuration
  Setting the link destination of the last descriptor to the address of the previous descriptor configures the descriptors in a loop. To end the loop, change the LE bit of header to 1 before DMACAa reads a descriptor, or follow the procedure to pause a transfer.

14.3.1.3 Write-Only Mode

Setting 1 to the WONLY bit in the CHCFG_n register falls in write-only mode.

Table 14.15 Write-Only Mode Settings

WONLY (CHCFG_n)	Mode	Description
0	Normal Mode	Performs a DMA transfer with the values set for the next register set.
1	Write-Only Mode	Performs DMA write transfers only without performing DMA read transfers.

In write-only mode, read operation for DMA transfers is not performed (descriptor read is performed as same as in normal mode). In register mode, use the value set for the NxSA_n register (x = 0 when RSEL = 0, and x = 1 when RSEL = 1) as the write data. In link mode, use the value in the SA field of a descriptor as write data.
Use this mode for initializing a memory area.

14.3.2 Transfer Mode

The DMACAa supports single transfer mode and block transfer mode.
When selecting a transfer mode, use the TM bit in the CHCFG_n register corresponding for each channel.

Table 14.16 Basic Transfer Settings

Transfer Mode	TM (CHCFG_n)	Description
Single transfer	0	For a single DMA request, performs a single DMA transfer.
Block transfer	1	For a single DMA request, continues DMA transfers until the requested transfer of data is completed.

14.3.2.1 Single Transfer Mode

When the DMACAa accepts a DMA transfer request, it performs DMA transfer a single time. Transfer proceeds every time a transfer request is accepted. This operation is repeated until the amount of data reaches the transfer size loaded from the N0TB_n or N1TB_n register to the CRTB_n register (arbitration between channels is performed for each DMA transfer).

14.3.2.2 Block Transfer Mode

Once a DMA transfer request is accepted, the transfer continues until transfer of the number of bytes loaded from the N0TB_n or N1TB_n register to the DMA transfer byte register (CRTB_n register) is completed (DMA transfer completion; arbitration between channels is performed for each DMA transfer).

14.3.3 DMA Channel Priority Control

As an arbitration method between channels, fixed priority mode and round-robin mode are supported. To select mode, use the PR bit in the DCTRL register. When the PR bit is 0, fixed priority mode is selected. When the PR bit is set to 1, round-robin mode is selected.

Table 14.17 Priority Control Settings

Transfer Mode	PR (DCTRL)	Description	Applications
Fixed Priority	0	Channels 0 to 7, channels 8 to 15 are fixed priority mode. CH0 (CH8) > CH1 (CH9) > CH2 (CH10) > CH3 (CH11) > CH4 (CH12) > CH5 (CH13) > CH6 (CH14) > CH7 (CH15)	Select this mode if channels have priority.
Round-Robin	1	Controls requests in round-robin mode.	Select this mode if you want to execute requests equally.

14.3.3.1 Fixed Priority Mode

In fixed priority mode, the priority among channels 0 to 7, and channels 8 to 15 are fixed.
In addition, the priority between a group for channels 0 to 7 and a group for channels 8 to 15 is round-robin mode.
The priority levels immediately after a reset and transfer through DMA channel 0 are shown in Figure 14.12.

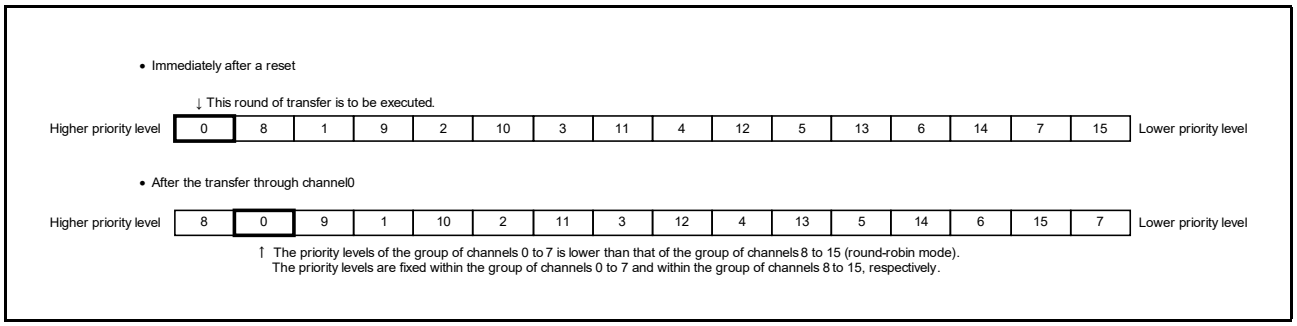


Figure 14.12 Priority Levels Immediately after a Reset and Transfer through DMA Channel 0

If DMA transfer requests are generated on multiple channels simultaneously, a DMA transfer request with a smaller channel number takes precedence.

Note: Channel 0 is handled with the highest priority, but the transfer with the next highest priority is performed because bus arbitration is performed after reading data from the transfer source of channel 0 finishes (Another read operation might interrupt the operation between reading data from and writing data to the same channel).

14.3.3.2 Round-Robin Mode

In round-robin mode, priority is changed every time a transfer of a channel is accepted so that the lowest priority is given to the channel in which the last transfer is performed.
The priority immediately after a reset is the same as the one for fixed priority mode as shown below.
The priority levels immediately after a reset and transfer through DMA channel 2 are shown in Figure 14.13.

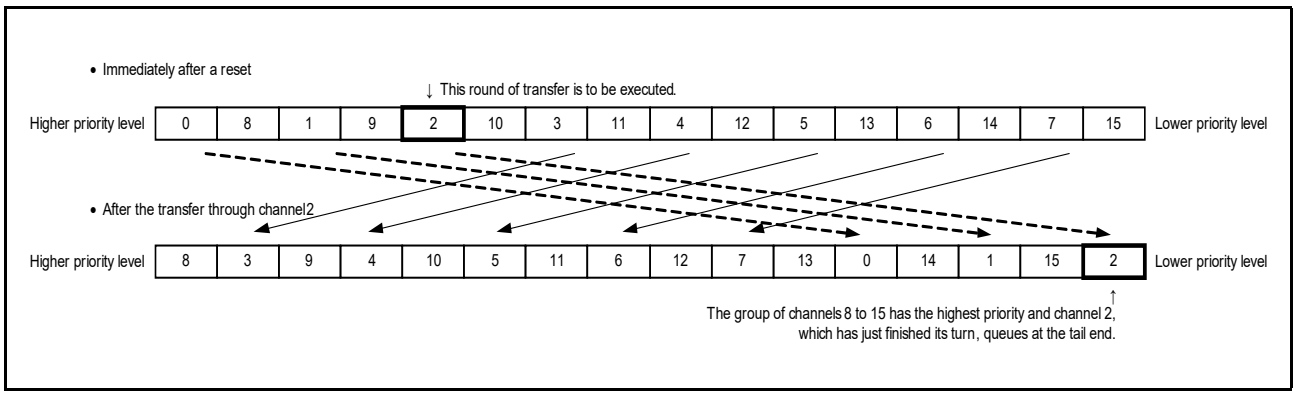


Figure 14.13 Priority Levels Immediately after a Reset and Transfer through DMA Channel 2

14.3.4 DMA Transfer Request

DMA activation requests have three types including software requests, on-chip peripheral module requests, and external interrupts.

Select the transfer request source for an on-chip peripheral module request, external interrupt, and software request with the DMAmSELn (m = 0 or 1, n = 0 to 15) register.

For details on the DMAmSELn (m = 0 or 1, n = 0 to 15) register, see section 14.2.8, DMACAa Unit 0 Source Select Register i (DMA0SELi) (i = 0 to 15), and section 14.2.9, DMACAa Unit 1 Source Select Register i (DMA1SELi) (i = 0 to 15).

14.3.4.1 Specifying Detection Operation of DMA Transfer Requests for Each Source

Detection methods for DMA transfer requests of on-chip peripheral module requests, external interrupts, and software requests might be specified according to sources.

For each DMA transfer source, set the LVL, HIEN, and LOEN bits in the CHCFG_n register according to Table 14.18, Table 14.19, and Table 14.20.

Table 14.18 Detection Operation Specification for Each Source of DMA Transfer Requests

DMA Transfer Request Source	Detection Operation Specification of DMA Transfer Requests
On-chip peripheral module request	Depends on specifications of the DMA transfer request source. (See Table 14.20.)
External interrupt	Detects the rising edge. Detects the high level.
Software request	Detects the rising edge.

Table 14.19 Method of Detecting DMA Transfer Request Signals

Mode	LVL (CHCFG_n)	HIEN (CHCFG_n)	LOEN (CHCFG_n)	Description
Edge detection	0	0	0	Disables detection.
			1	Detects the falling edge.
		1	0	Detects the rising edge.
			1	Setting prohibited
Level detection	1	0	0	Disables detection.
			1	Setting prohibited
		1	0	Detects the high level.
			1	Setting prohibited

Table 14.20 DMA Transfer Request Detection Operation Setting Table (1 / 2)

DMA Transfer Request Source	DMA Transfer Source	Transfer Source	Transfer Destination	DMAm SELn [7:0]	CHCFG_n				
					TM	LVL	HIEN	LOEN	SEL[2:0]
External Interrupt	IRQ0	Arbitrary	Arbitrary	04h	0/1	0/1*1	1*1	0*1	DMACAa0/1 ch0/8: 0h ch1/9: 1h ch2/10: 2h ch3/11: 3h ch4/12: 4h ch5/13: 5h ch6/14: 6h ch7/15: 7h
	IRQ1			05h					
	IRQ2			06h					
	IRQ3			07h					
	IRQ4			08h					
	IRQ6			0Ah					
	IRQ7			0Bh					
	IRQ9			0Dh					
	IRQ11			0Fh					
	IRQ12			10h					
	IRQ13			11h					
	IRQ14			12h					
CMT Unit 0	Compare match 0	Arbitrary	Arbitrary	15h	0/1	0	1	0	
	Compare match 1			16h					
CMT Unit 1	Compare match 0	Arbitrary	Arbitrary	17h	0/1	0	1	0	
	Compare match 1			18h					
CMTW Unit 0	Compare match	Arbitrary	Arbitrary	19h	0/1	0	1	0	
	Input capture 0			1Ah					
	Input capture 1			1Bh					
	Output compare 0			1Ch					
	Output compare 1			1Dh					
CMTW Unit 1	Compare match	Arbitrary	Arbitrary	1Eh	0/1	0	1	0	
	Input capture 0			1Fh					
	Input capture 1			20h					
	Output compare 0			21h					
	Output compare 1			22h					
DMACAa0	DMACAa0 Software trigger	Arbitrary	Arbitrary	FBh	0/1	0	1	0	
DMACAa1	DMACAa1 Software trigger	Arbitrary	Arbitrary	FCh	0/1	0	1	0	
USB	FuncDMA request 1	Arbitrary (for transmission) / D0FIFO (for reception)	D0FIFO (for transmission) / Arbitrary (for reception)	2Bh	0	1	1	0	
	Func DMA request 2			2Ch					
Ether PHY	Ether PHY Interrupt 0	Arbitrary	Arbitrary	30h	0/1	0/1	1	0	
	Ether PHY Interrupt 1	Arbitrary	Arbitrary	31h					
	Ether PHY Interrupt 2	Arbitrary	Arbitrary	32h					
ESC	Sync0 interrupt	Arbitrary	Arbitrary	49h	0/1	0	1	0	
	Sync1 interrupt			4Ah					
	Ether CAT interrupt			4Bh					
	SOF interrupt			4Ch					
	EOF interrupt			4Dh					

Table 14.20 DMA Transfer Request Detection Operation Setting Table (2 / 2)

DMA Transfer Request Source	DMA Transfer Source	Transfer Source	Transfer Destination	DMAm SELn [7:0]	CHCFG_n				
					TM	LVL	HIEN	LOEN	SEL[2:0]
RSPI Channel 0	Reception buffer full	SPDR	Arbitrary	50h	0	0	1	0	DMACAa0/1 ch0/8: 0h ch1/9: 1h ch2/10: 2h ch3/11: 3h ch4/12: 4h ch5/13: 5h ch6/14: 6h ch7/15: 7h
	Transmission buffer empty	Arbitrary	SPDR	51h	0	0	1	0	
RSPI Channel 1	Reception buffer full	SPDR	Arbitrary	54h	0	0	1	0	
	Transmission buffer empty	Arbitrary	SPDR	55h	0	0	1	0	
SCIFA Channel 0	Reception buffer full	FRDR	Arbitrary	61h	0	1	1	0	
	Transmission buffer empty	Arbitrary	FTDR	62h	0	1	1	0	
SCIFA Channel 1	Reception buffer full	FRDR	Arbitrary	65h	0	1	1	0	
	Transmission buffer empty	Arbitrary	FTDR	66h	0	1	1	0	
SCIFA Channel 2	Reception buffer full	FRDR	Arbitrary	6Eh	0	1	1	0	
	Transmission buffer empty	Arbitrary	FTDR	6Fh	0	1	1	0	
SCIFA Channel 3	Reception buffer full	FRDR	Arbitrary	72h	0	1	1	0	
	Transmission buffer empty	Arbitrary	FTDR	73h	0	1	1	0	
SCIFA Channel 4	Reception buffer full	FRDR	Arbitrary	76h	0	1	1	0	
	Transmission buffer empty	Arbitrary	FTDR	77h	0	1	1	0	
RIIC Channel 1	Data reception completed	ICDRR	Arbitrary	7Dh	0	0	1	0	
	Transmission data Empty	Arbitrary	ICDRT	7Eh	0	0	1	0	
ELC	ELCIRQ1	Arbitrary	Arbitrary	F2h	0/1	0	1	0	
	ELCIRQ2			F3h					

Note 1. Set the LVL, HIEN, and LOEN bits of the external interrupts (IRQ0 to IRQ15) as follows.
For the setting of the IRQCRi (i = 0 to 15) register, see section 12.3.1, Selecting Interrupt Request Destinations.
LVL: Set according to the edge/level setting of the IRQCRi register.
HIEN: Set to 1 regardless of the detection level of the IRQCRi register.
LOEN: Set to 0 regardless of the detection level of the IRQCRi register.

Remarks: CHCFG_n register setting values
TM Bit
0: Single transfer
1: Block transfer
LVL bit
0: Detects the edge of a DMA request.
1: Detects the level of a DMA request.

Note: Instead of an interrupt source from peripheral modules, transfer completion sources of DMACAa channels selected by the DMACAa source select register are connected to the vector number selected by the DMACAa source select register.
Note: Be sure to select edge detection with the PLSn register as the transfer completed interrupt detection type for the DMACAa.

14.3.4.2 Edge Detection

Setting the LVL bit in the CHCFG_n register to 0 detects the edge.
Setting the HIEN bit in the CHCFG_n register to 1 detects the rising edge.

14.3.4.3 Level Detection

Setting the LVL bit in the CHCFG_n register to 1 detects the level.
If a DMA transfer request is active (according to HIEN or LOEN settings) for two consecutive clocks (ICLK) or more, it is recognized as a DMA request.

14.3.5 Forced Ejection Request

When a forced ejection request is input, data that is not yet transferred in a buffer is transferred to the DMA transfer destination address. After data is flushed, the DMA transfer resumes.

The following are notes on forced ejection requests:

- If a forced ejection request conflicts with DMA transfer request input, the forced ejection takes precedence, and then the DMA transfer is performed.

- Differences from flush mode described in section 14.3.10.2, Aborting a Transfer (Buffer Flush: SBE = 1) (The EN bit is cleared to 0 when the SBE bit in the CHCFG_n register = 1).

Flush mode: DMACAa stops operation after data in a buffer is written.
Forced ejection request: A DMA transfer continues after flush operation ends.

14.3.5.1 Software Forced Ejection Request

For software forced ejection requests, use the SETSSWPRQ bit in the CHCTRL_n register.
To request a forced ejection, set 1 to the SETSSWPRQ bit. DMACAa outputs data in a buffer to the DMA transfer destination.

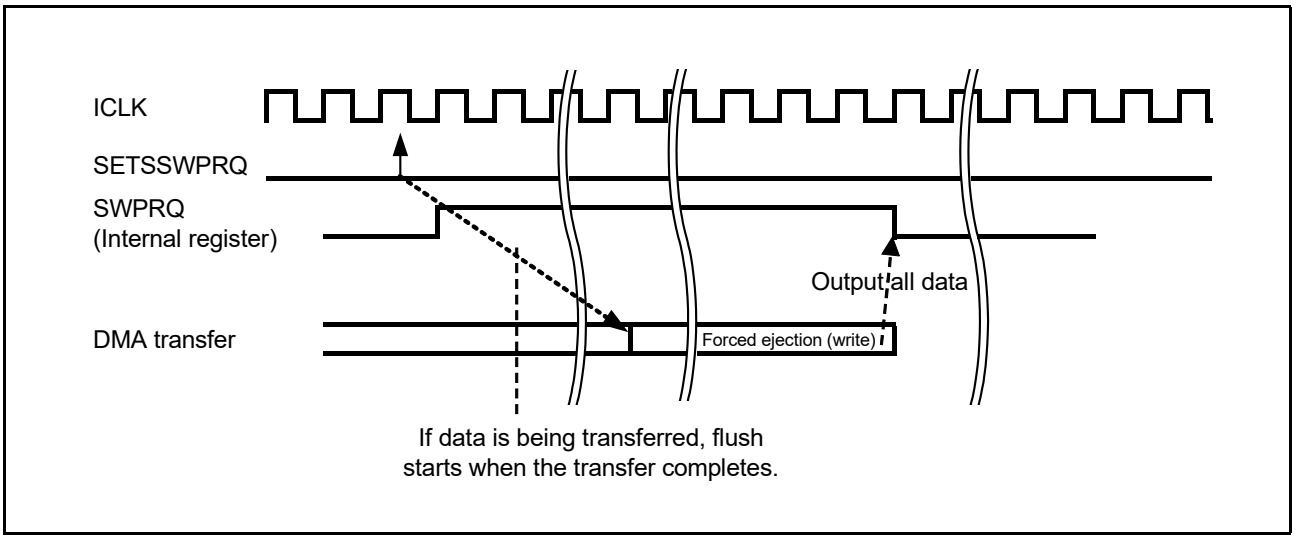


Figure 14.14 Software Forced Ejection Timing

14.3.6 Interval Count Function

By the setting for the ITVL field in the CHITVL_n register, the execution interval of DMA transfers can be adjusted. This function is used to avoid occupation of a bus by DMACAa. Until the count value becomes 0, no DMA transfer for the next DMA request is performed.

14.3.7 Differences in Operation According to the Transfer Data Size

14.3.7.1 When the Transfer Data Size on the Transfer Source is Small

As the transfer data size at the destination is large, reading from the source proceeds several times, and this is followed by writing to the destination.

14.3.7.2 When the Transfer Data Size on the Transfer Destination is Small

Because the transfer data size on the transfer source is large, after a single read operation, write operation to the transfer destination is performed a few times.

14.3.7.3 When the Size of Transfer Data on the Transfer Destination and on the Transfer Destination is the Same

Every time a DMA transfer request is detected, read operation is performed on the transfer source, and write operation is performed on the transfer destination.

14.3.8 DMA Transfer Status

The CHSTAT_n register indicates the DMA transfer status of each channel.
The TACT bit in the CHSTAT_n register indicates that DMA operation is being performed on channel n. Writing 1 to the SETEN bit in the CHCTRL_n register sets 1. The TACT bit remains 1 while accessing a descriptor, or waiting for a DMA request.
The TACT bit is cleared when the EN bit in the CHSTAT_n register is cleared (for details on clear conditions, see section 14.2.7, Channel Status Register n (CHSTAT_n)), and DMA transfers for the set number of times are finished.
The TACT bit is not cleared even when a DMA transfer finishes, but the EN bit is not cleared (when the REN bit in the CHCFG_n register = 1 in register mode, or the next descriptor access is performed in link mode).

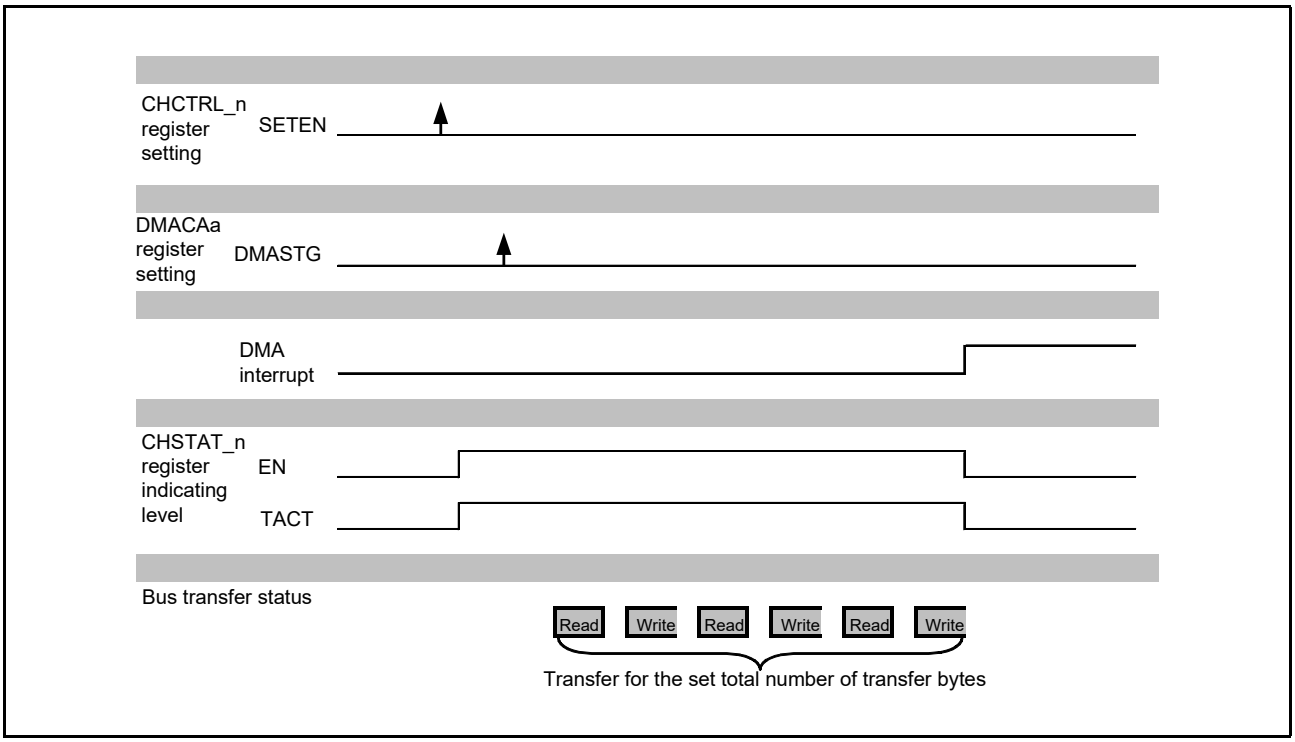


Figure 14.15 DMACAa Status Example (Software Request)

14.3.9 Suspending a Transfer

You can suspend a DMA transfer at the SETSUS bit in the CHCTRL_n register. At that time, if there is an already running bus cycle, waits for the cycle to end, and then suspends the transfer. Writing 1 to the CLRSUS bit in the CHCTRL_n register resumes from the suspended state.

To check if the transfer is suspended, set the SETSUS bit in the CHCTRL_n register, and then make sure that the SUS bit in the CHSTAT_n register, or the SUS bit in the DST_SUS register on the applicable channel is set to 1.

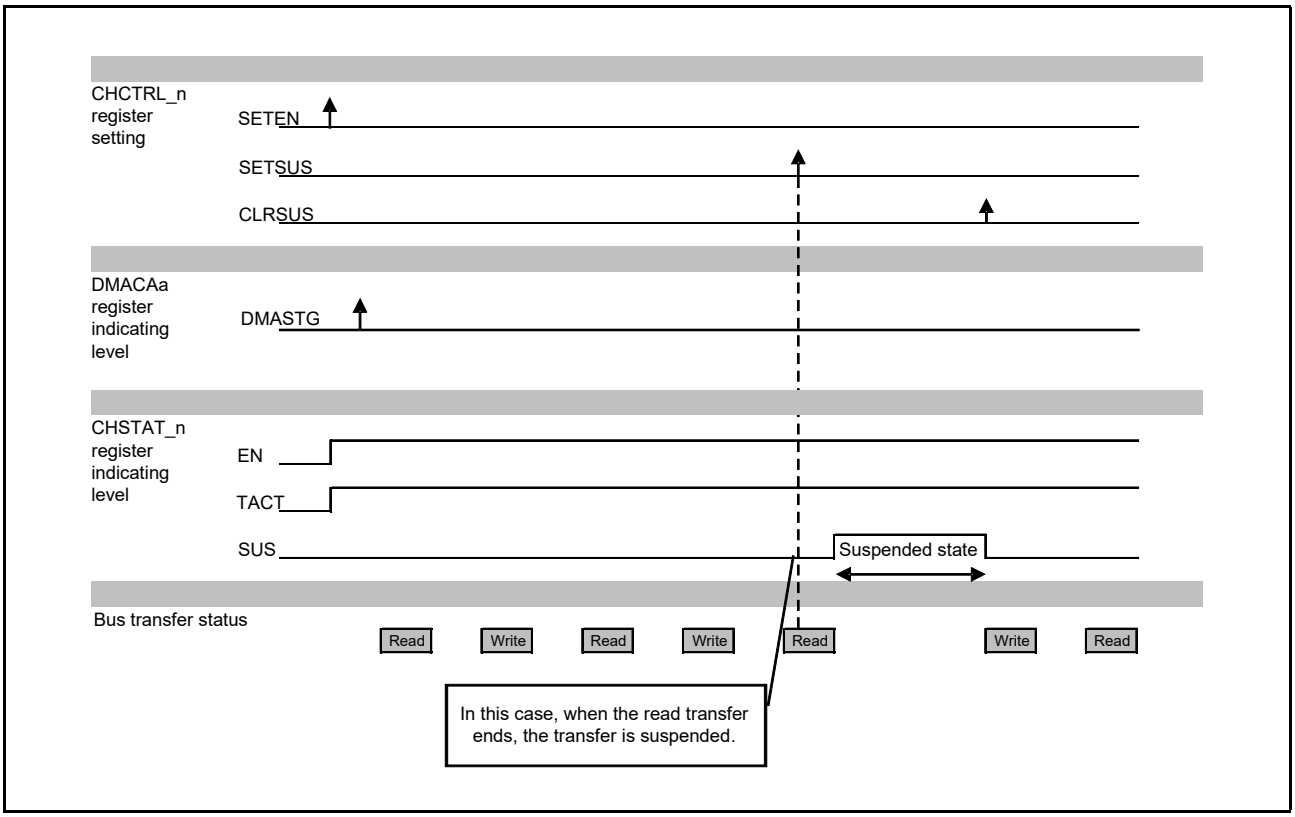


Figure 14.16 DMACAa Suspended State (Software Request Block Transfer)

14.3.10 Aborting a Transfer

During a DMA transfer, if you write 1 to the CLREN bit in the CHCTRL_n register, you can abort the DMA transfer of the channel. As processing after aborting the transfer, you can use the SBE bit in the CHCFG_n register to determine whether to flush data remaining in a buffer when a transfer is suspended. By default, SBE = 0 (do not flush data) is selected.
When this mode (flush data) is activated, if a transfer which is being performed when the CLREN bit in the CHCTRL_n register = 1 is aborted, data remaining in the buffer of DMACAa is flushed, and the operation stops.

14.3.10.1 Aborting a Transfer (No Buffer Flush: SBE = 0)

During a DMA transfer, if you write 1 to the CLREN bit in the CHCTRL_n register, you can abort and then stop the DMA transfer. After the transfer stops, write 1 to the SWRST bit in the CHCTRL_n register, and clear the contents within the DMACAa before setting the next transfer.

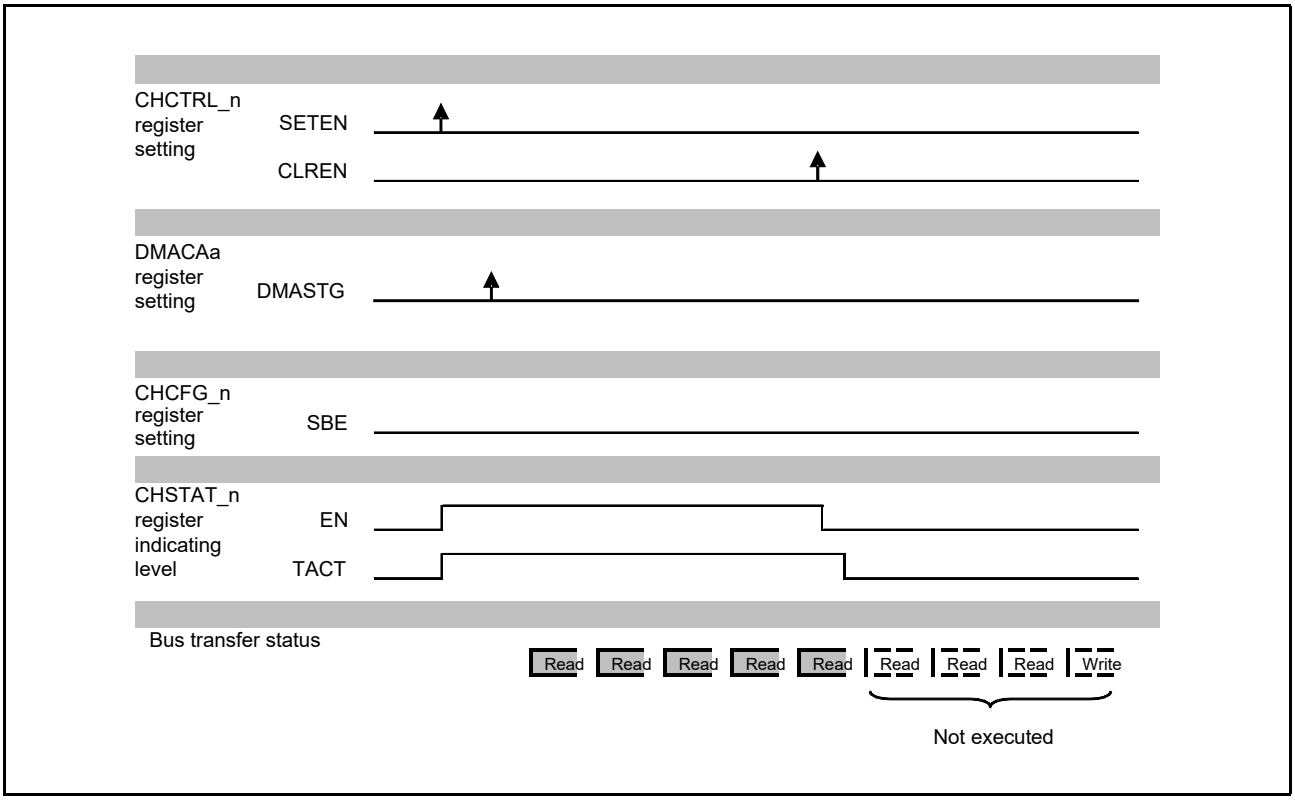


Figure 14.17 Aborting a DMA Transfer

- When the TACT bit in the CHSTAT_n register is cleared, you can confirm that the channel stops completely.
- If a DMA transfer is aborted, no DMA transfer completion interrupt is generated.
- If a DMA transfer is aborted, the transfer stops when the next read operation completes. (Note that if there is data that can be written in the buffer, the transfer stops after the data is written.)

14.3.10.2 Aborting a Transfer (Buffer Flush: SBE = 1)

During a DMA transfer, if you write 1 to the CLREN bit in the CHCTRL_n register, you can abort the DMA transfer. After the transfer stops, set the SWRST bit in the CHCTRL_n register, and clear the contents within the DMACAa before setting the next transfer.

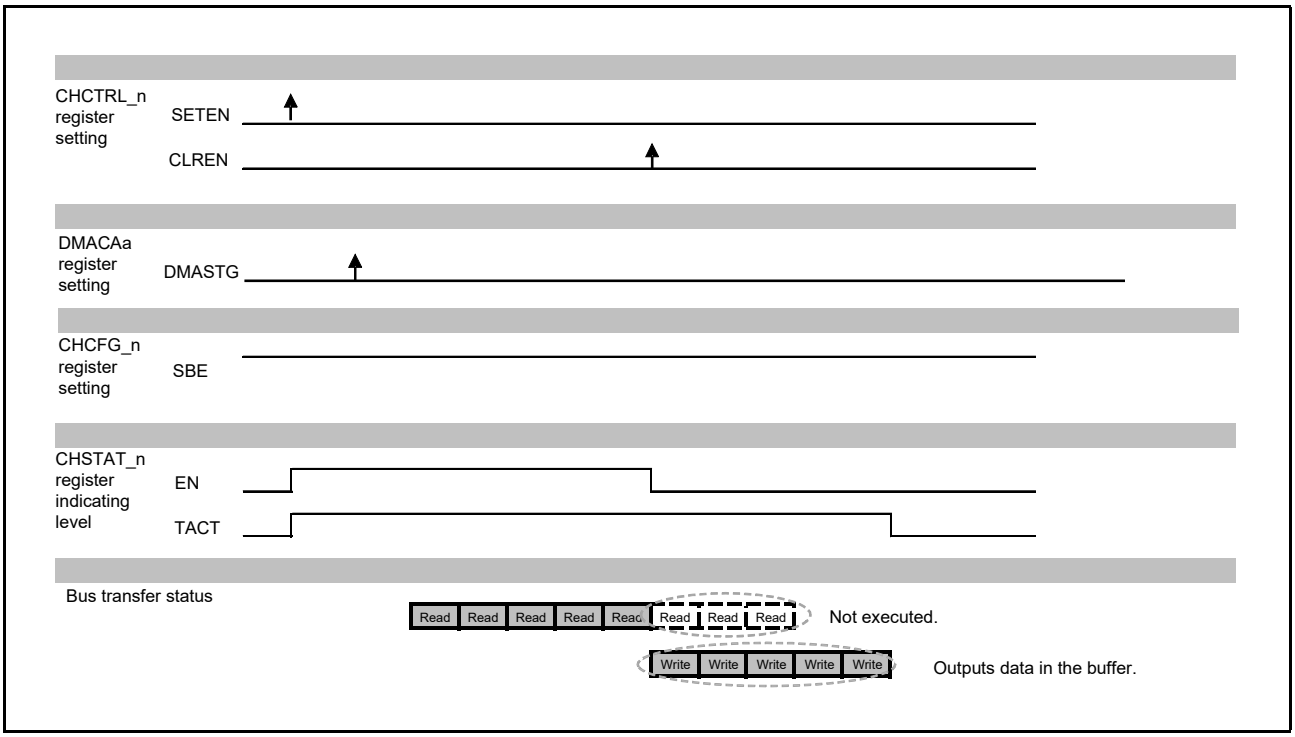


Figure 14.18 Aborting DMA Transfer (Buffer Flush Mode)

- The above figure shows an example when 1 is written to the CLREN bit in the CHCTRL_n register during the fifth read transfer in flush mode (the SBE bit in the CHCFG_n register = 1), and the transfer is aborted. It illustrates how read data is written, and the DMA transfer stops.
- When the TACT bit in the CHSTAT_n register is cleared to 0, you can confirm that the channel stops completely.

14.3.10.3 Checking If the Channel Stops

When 1 is written to the CLREN bit in the CHCTRL_n register, and the EN bit in the CHSTAT_n register is cleared to 0, if a transfer is already performed on a bus, DMACAa cannot stop immediately. To check if DMA stops completely, make sure that the EN bit is cleared to 0, and the TACT bit in the CHSTAT_n register is cleared to 0.

14.3.10.4 Procedure for Aborting a Transfer

The following is the procedure for stopping a transfer:

1. Write 1 to the CLREN bit in the CHCTRL_n register.
2. By reading the CHSTAT_n register, make sure the TACT bit is cleared to 0. When TACT = 0, DMA stops completely. When TACT = 1, perform polling until the TACT bit is cleared to 0.
3. After a transfer is aborted, if you want to perform the next DMA transfer, you must set the SWRST (software reset) bit in the CHCTRL_n register immediately before the next transfer starts.

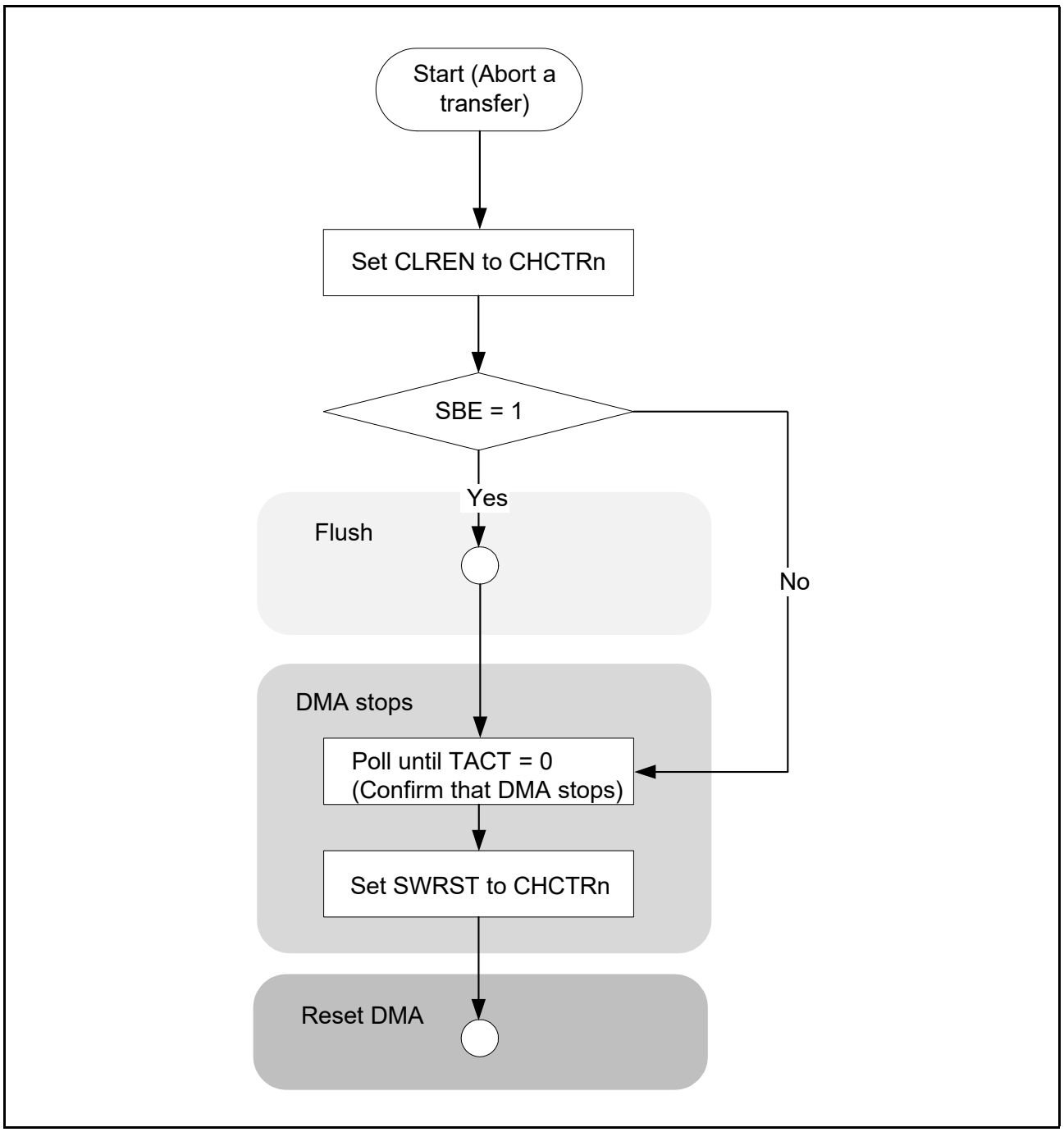


Figure 14.19 Operation Flow of Aborting a Transfer

14.4 Interrupts

14.4.1 Interrupt Sources

DMACAa has two types of interrupt sources, such as DMA transfer completion interrupts and DMA error interrupts for each channel.
Table 14.21 shows the relationship among interrupt sources, enable bits, and status flags.

Table 14.21 Interrupt Sources of DMACAa

Interrupt Source		Interrupt Enable Bit	Interrupt Status Flag	Output Condition
DMA transfer completion interrupt	DMA transfer completion	CHCFG_n.DEM	CHSTAT_n.END	When a transfer for the total number of transfer bytes loaded to the CRTB_n register completes (after a write back if write back is performed in link mode)
	Descriptor invalid	CHCFG_n.DIM		In link mode, when the DRRP and DIM bits in the CHCFG_n register = 0, and header of the read descriptor's LV = 0
DMA error interrupt		— (Mask disabled)	CHSTAT_n.ER	When a bus error occurs during a DMA transfer and descriptor access

14.4.2 DMA Transfer Completion Interrupts

A DMA transfer completion interrupt is an interrupt request signal indicating that the DMA transfer completes.
Each bit of the DMA transfer completion interrupt corresponds to each channel.

When the transfer of the total number of bytes for transfer loaded to the CRTB_n register is completed, the END bit in the CHSTAT_n register is set to 1. At that time, if the DEM bit in the CHCFG_n register = 0, a DMA transfer completion interrupt is generated (n = 15 to 0). To perform write back in link mode, an interrupt is generated after the write back.

In addition, when the DRRP bit in the CHCFG_n register = 0 in link mode, and header of the read descriptor is LV = 0, the DER bit in the CHSTAT_n register is set to 1. At that time, if the DIM bit in the CHCFG_n register = 0, a DMA transfer completion interrupt is generated.

Note: Instead of an interrupt source from peripheral modules, transfer completion sources of DMACAa channels selected by the DMACAa source select register are connected to the vector number selected by the DMACAa source select registers (DMA0SELi, DMA1SELi). (The vector number selected by the DMA source selection register is handled as the vector number of the DMA transfer completed interrupt.)
Note: Be sure to select edge detection with the PLSn register as the transfer completed interrupt detection type for the DMACAa.

For details, see section 12.3.1, Selecting Interrupt Request Destinations.

14.4.3 DMA Error Interrupt

If a bus error occurs during a DMA transfer or descriptor access, this module determines an error occurred, and stops the transfer. When a bus error occurs, the EN bit in the CHSTAT_n register for channel n where a transfer is performed is cleared to 0, and the ER bit is set to 1 (n = 15 to 0). In addition, a DMA error interrupt is generated.
DMA error interrupts cannot be masked.

Data for a series of error transfers cannot be guaranteed. You must perform the transfer from the beginning by using the following procedure.

1. Set the SWRST bit in the CHCTRL_n register to 1.
2. Set each register again.

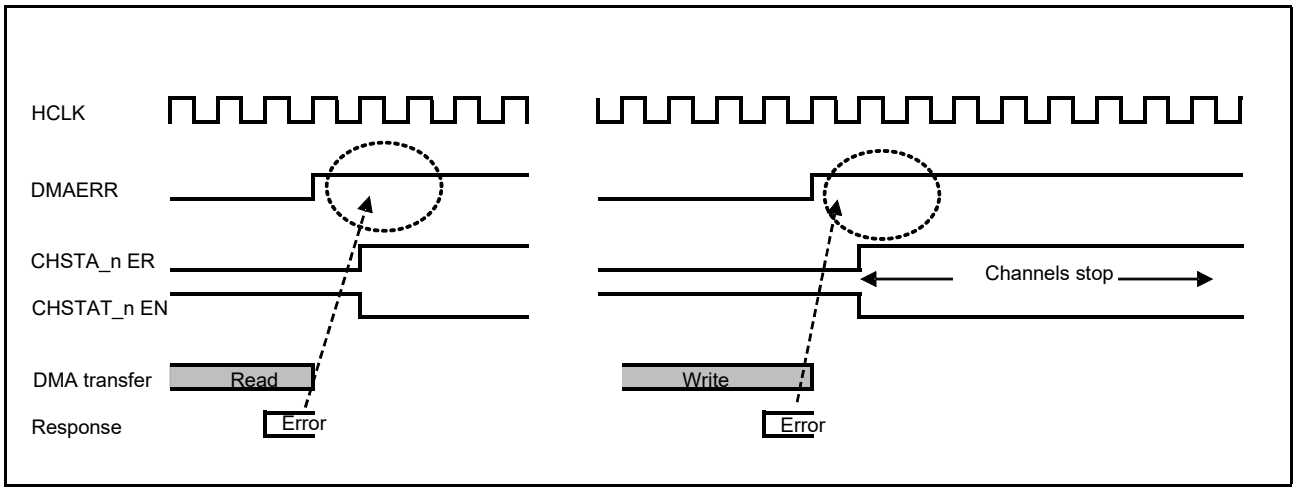


Figure 14.20 Stop Timing for Responding to a Bus Error

Note: When the CPU accesses a register of DMACAa, if a bus error occurs, no DMA error interrupt is generated.

14.5 DMA Setting Examples

This section provides examples of DMA transfers. The following table lists the transfer conditions for the setting examples described in this section.

Table 14.22 List of Transfer Conditions for the DMA Transfer Setting Examples

Setting Example	DMA Mode	Transfer Mode	Transfer Request
Setting example 1	Register mode	Block transfer mode	Software
Setting example 2	Register mode (continuous execution)	Block transfer mode	Software
Setting example 3	Link mode	Block transfer mode	Software

14.5.1 Setting Example 1 (Register Mode Software Request)

This subsection provides a setting example of DMA transfers that use software requests in register mode.

Table 14.23 DMA Transfer Setting Example 1

Item		Description	
Channel to use		DMACAa0 channel 2	
Priority control		Round-robin	
DMA mode		Register mode	
Transfer mode		Block transfer mode	
Register set to use		Next1 register set	
Transfer source/transfer destination		Transfer source	Transfer destination
	Start address	0000 0080h	0080 0080h
	Address direction	Increment	Increment
	Data size	8 bits	256 bits
Number of DMA transfer bytes		128 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		None	

Setting Example 1
DCTRL = 0000 0001h (DMA setting)

N1SA = 0000 0080h (Transfer source address)
N1DA = 0080 0080h (Transfer destination address)
N1TB = 0000 0080h (Number of transfer bytes)

CHCFG = 1045 0222h (Configuration)
CHITVL = 0000 0000h (Interval)

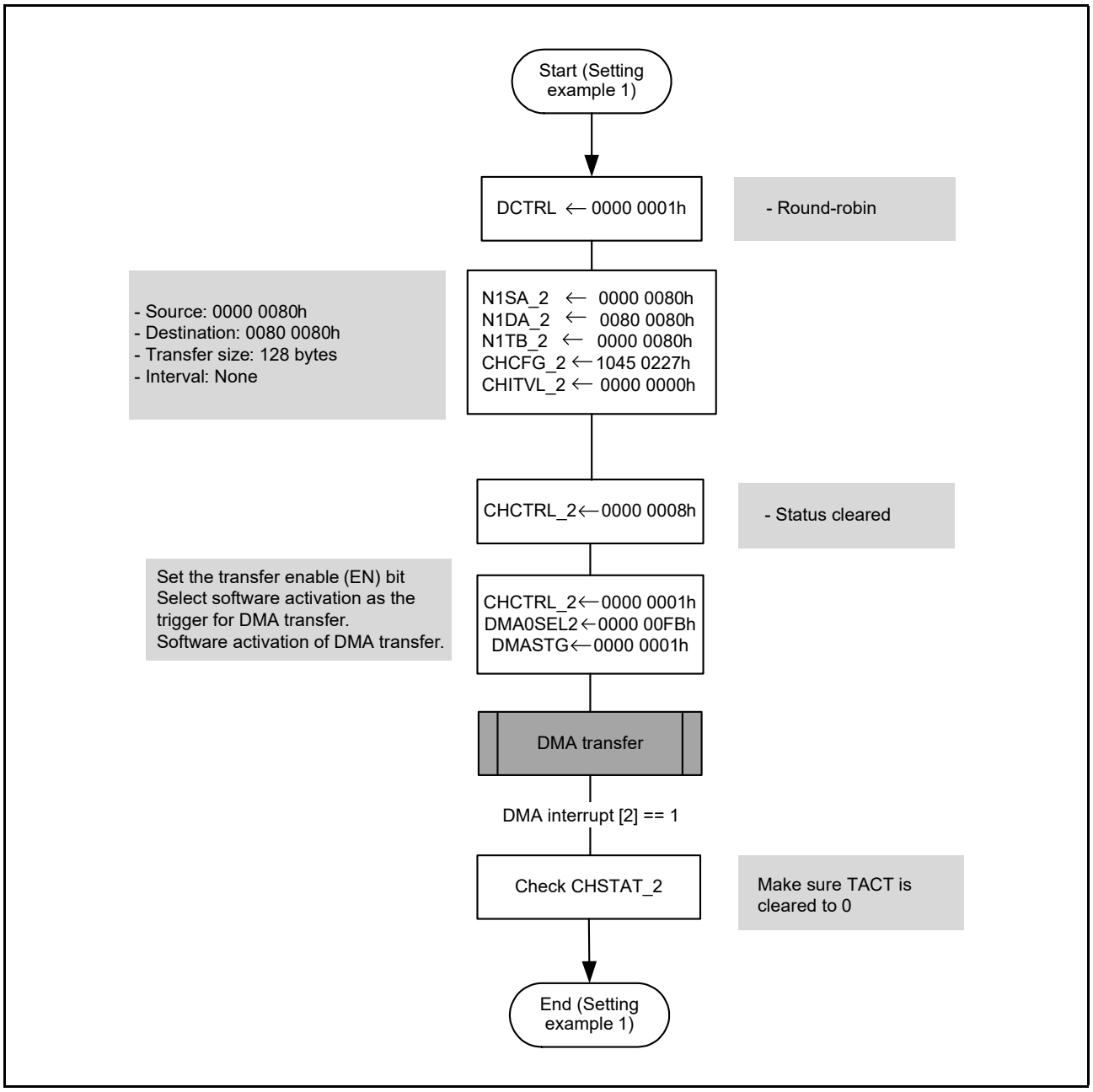


Figure 14.21 Setting Example 1

Note: DMA interrupt [2] indicates the interrupt source assigned to serve as the trigger for DMA transfer on channel 2.

14.5.2 Setting Example 2 (Register Mode Continuous Execution)

This subsection provides a setting example of DMA transfers that use Next0/1 Register Sets in series in register mode.

Table 14.24 DMA Transfer Setting Example 2

Item		Description	
Channel to use		DMACAa0 channel 1	
Priority control		Round-robin	
DMA mode		Register mode	
Transfer mode		Block transfer mode	
Register set to use		Next0 register set -> Next1 register set in series	
Transfer source/transfer destination (Next0)		Transfer source	Transfer destination
	Start address	1111 0000h	0080 0080h
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	Number of DMA transfer bytes	512 bytes	
Transfer source/transfer destination (Next1)		Transfer source	Transfer destination
	Start address	0000 0080h	1000 0000h
	Address direction	Fixed	Fixed
	Data size	32 bits	512 bits
	Number of DMA transfer bytes	2,048 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		Masks when Next0 completes.	

Setting Example 2
DCTR = 0000 0001h (DMA setting)

N0SA = 1111 0000h (Transfer source address)
N0DA = 0080 0080h (Transfer destination address)
N0TB = 0000 0200h (Number of transfer bytes)

N1SA = 0000 0080h (Transfer source address)
N1DA = 1000 0000h (Transfer destination address)
N1TB = 0000 0800h (Number of transfer bytes)

CHCFG = 6176 2007h (Configuration)
CHITVL = 0000 0000h (Interval)

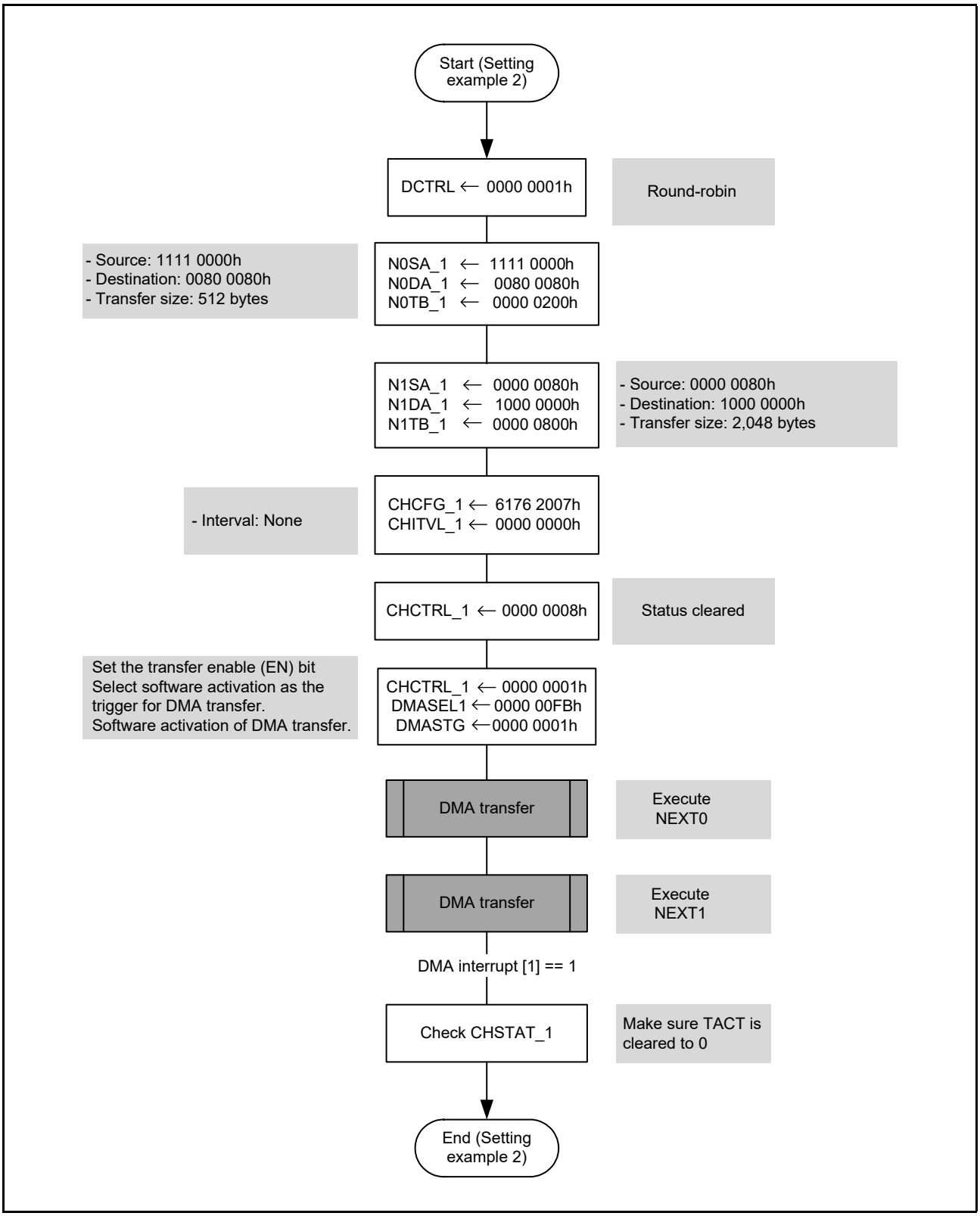


Figure 14.22 Setting Example 2

Note: DMA interrupt [1] indicates the interrupt source assigned to serve as the trigger for DMA transfer on channel 1.

14.5.3 Setting Example 3 (Link Mode)

This subsection provides a setting example when a DMA transfer is performed in link mode.

Table 14.25 DMA Transfer Setting Example 3

Item	Description
Channel to use	DMACAa0 channel 0
Priority control	Round-robin
DMA mode	Link mode
Transfer mode	Block transfer mode
Descriptor start address	0080 0000h

Table 14.26 DMA Transfer Setting Example 3 (Descriptor 1)

Item		Description	
Descriptor start address		0080 0000h	
Next descriptor start address		0080 1000h	
Transfer mode		Block transfer mode	
Transfer source/transfer destination		Transfer source	Transfer destination
	Start address	1111 0000h	0000 0080h
	Address direction	Increment	Increment
	Data size	32 bits	32 bits
Number of DMA transfer bytes		2,048 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		Masks when a DMA transfer on descriptor 1 completes.	
Descriptor format		1 (8 words)	
Descriptor header			
	Write back of the LV bit	Enable (WBD = 0)	
	Next link destination	Available (LE = 0)	
	Descriptor enabled	Enabled (LV = 1)	

Table 14.27 DMA Transfer Setting Example 3 (Descriptor 2)

Item		Item	
Descriptor start address		0080 1000h	
Next descriptor start address		0080 2000h	
Transfer mode		Block transfer mode	
Transfer source/transfer destination		Transfer source	Transfer destination
	Start address	0080 0080h	0000 0080h
	Address direction	Increment	Increment
	Data size	256 bits	256 bits
Number of DMA transfer bytes		1,024 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		Masks when a DMA transfer on descriptor 2 completes.	
Descriptor format		1 (8 words)	
Descriptor header			
	Write back of the LV bit	Enable (WBD = 0)	
	Next link destination	Available (LE = 0)	
	Descriptor enabled	Enabled (LV = 1)	

Table 14.28 DMA Transfer Setting Example 3 (Descriptor 3)

Item		Item	
Descriptor start address		0080 2000h	
Next descriptor start address		—	
Transfer mode		Block transfer mode	
Transfer source/transfer destination		Transfer source	Transfer destination
	Start address	0000 0080h	0080 2000h
	Address direction	Increment	Increment
	Data size	512 bits	512 bits
Number of DMA transfer bytes		4,096 bytes	
DMA transfer request		Software request	
DMA transfer completion interrupt output mask		Do not mask.	
Descriptor format		1 (8 words)	
Descriptor header			
	Write back of the LV bit	Enable (WBD = 0)	
	Next link destination	None (LE = 1)	
	Descriptor enabled	Enabled (LV = 1)	

Setting Example 3
DCTRL = 0000 0001h (DMA setting)
NXLA = 0080 0000h (Descriptor start address)
CHCFG = 8000 0000h (Configuration)

Table 14.29 Descriptor Settings

Item	Descriptor 1	Descriptor 2	Descriptor 3
Header	1100 0000h	1100 0000h	1300 0000h
SA (Source Address)	1111 0000h	0080 0080h	0000 0080h
DA (Destination Address)	0000 0080h	0000 0080h	0080 2000h
TB (Transaction Bytes)	0000 0800h	0000 0400h	0000 1000h
CFG (Configuration)	8142 2220h	8145 5220h	8046 6220h
ITVL (Interval)	0000 0000h	0000 0000h	0000 0000h
EXT (Extension)	0000 0000h	0000 0000h	0000 0000h
NXLA (Next Link Address)	0080 1000h	0080 2000h	0000 0000h

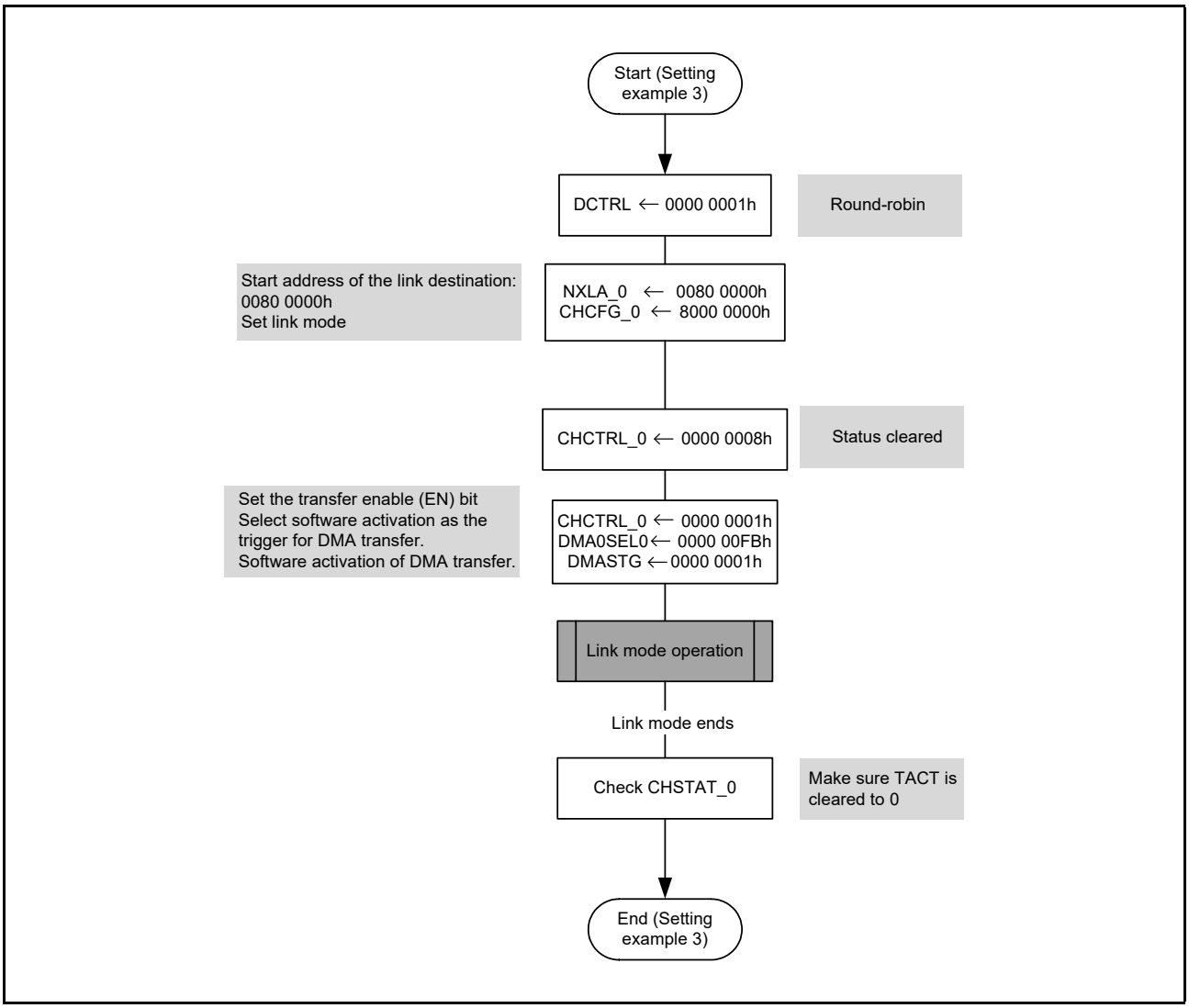


Figure 14.23 Setting Example 3

14.5.4 Next Register Continuous Execution Settings

The following figure shows a flow chart when using two Next register sets to continue DMA transfers in register mode. While performing a DMA transfer of a Next register, set the other Next register, and perform the DMA transfers in series.

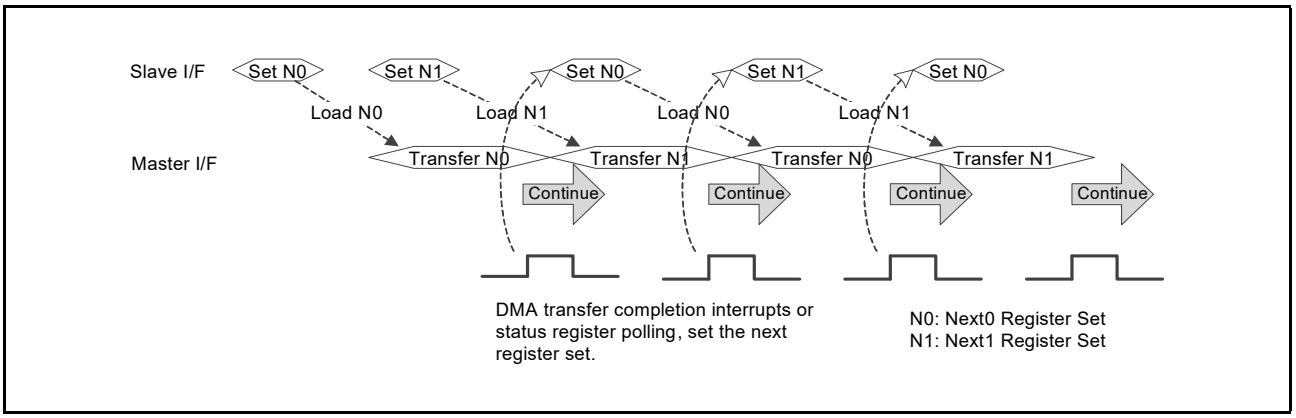


Figure 14.24 Next Register Continuous Execution Setting Image

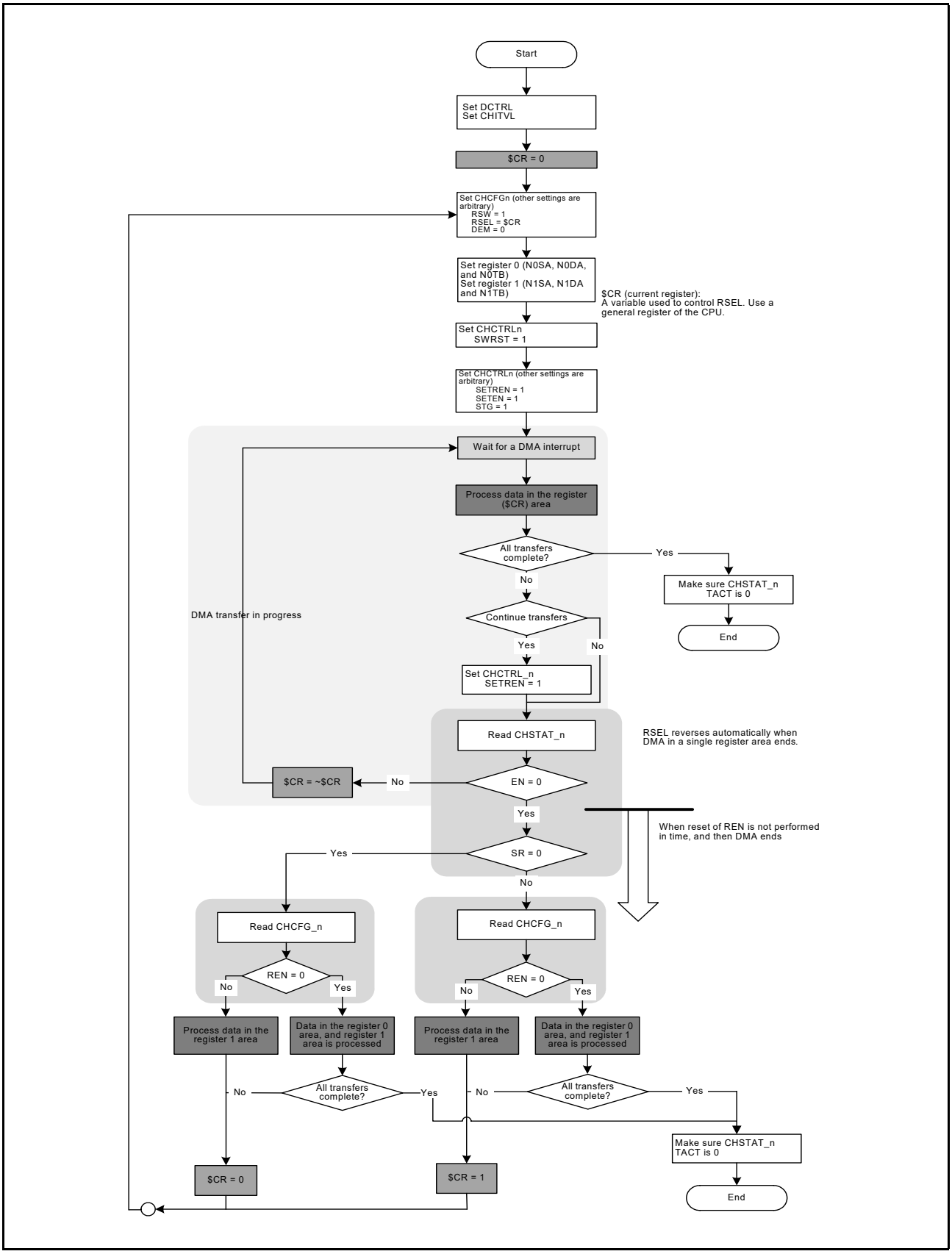


Figure 14.25 Setting Example of Next Register Continuous Execution

- Supplementary information:
  Save the register sets 0 (N0SA_n, N0DA_n, and N0TB_n registers) and 1 (N1SA_n, N1DA_n, and N1TB_n registers) in a general register (call this register value $CR for convenience).
  Every time a DMA transfer of a register set completes (a DMA transfer completion interrupt is generated), the REN bit in the CHCFG_n register is cleared to 0 automatically. To perform transfers in series, write 1 to the SETREN bit in the CHCTR_n register. By doing so, the REN bit in the CHCFG_n register is also set.

  In this mode, two Next registers are executed in series. However, if the SETREN bit is not set before a DMA transfer completes (before the next DMA transfer completion interrupt is generated), the continuous execution stops. In this case, by reading the SR and EN bits in the CHSTAT_n register, and the REN bit in the CHCFG_n register, you can check how far the transfer is performed. To resume the transfer, perform the procedure described in the above flow chart.

14.6 Usage Notes

This subsection provides notes on this module.

- When a transfer of which source and destination are in the same, or partially shared area is performed, consistency of data cannot be guaranteed. Therefore, do not perform a transfer of which source and destination address areas overlap.
- When DAD = 1 (fixed transfer source address), no SKIP transfer can be performed on the transfer destination. If a transfer is performed with this setting, operation cannot be guaranteed. Do not perform this type of transfers.

- When SAD = 1 (fixed transfer source address), no SKIP transfer can be performed on the transfer source. If a transfer is performed with this setting, operation cannot be guaranteed. Do not perform this type of transfers.

- Since access to the region from A00E 0000h to A010 0000h within the peripheral I/O register region is not supported, locations or ranges within this region should never be set as sources or destinations for transfer.

15. Event Link Controller (ELC)

15.1 Overview

The event link controller (ELC) connects (links) events generated by various peripheral modules to other modules. Event link allows direct cooperation among modules without CPU intervention.

Table 15.1 lists the specifications of the ELC, and Figure 15.1 shows a block diagram of the ELC.

Table 15.1 ELC Specifications

Item	Description
Event link function	• 16 types of event signals can be directly connected to modules. • The operation of timer modules can be selected when an event is input to the timer module. • Event link operation is possible for port B and port E. Single port*1: An event link can be set for a single bit specified in a port. Port group*1: An event link can be set for a group of single bits specified within eight I/O ports.
Low power consumption function	Module stop state can be set.

Note 1. The single port and port group specified as the input generate an event according to the change in the connected signal value.

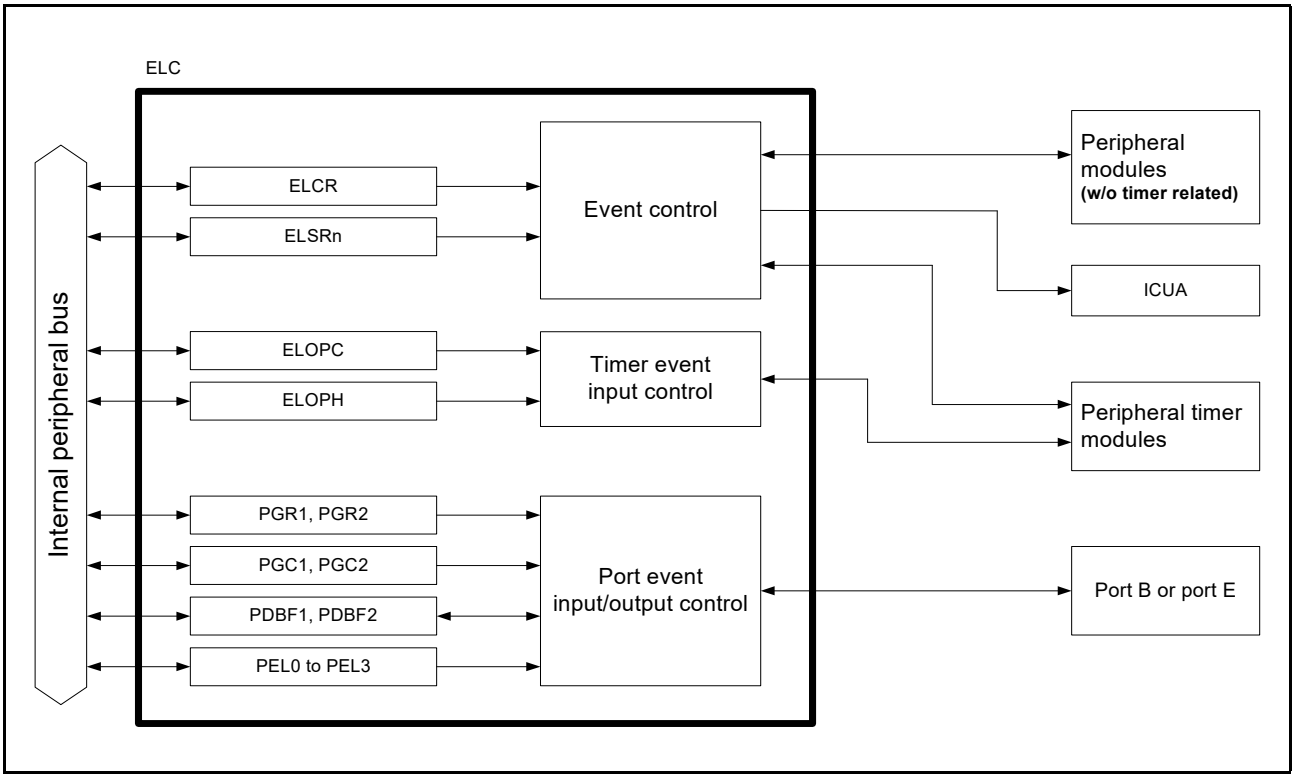


Figure 15.1 ELC Block Diagram (n = 7, 18 to 27, 33)

15.2 Register Descriptions

15.2.1 Event Link Control Register (ELCR)

The ELCR register controls operation of the ELC.

Address(es): A008 0B00h

	b7	b6	b5	b4	b3	b2	b1	b0
	ELCON	—	—	—	—	—	—	—
Value after reset:	0	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b7	ELCON	All Event Link Enable	0: ELC function is disabled. 1: ELC function is enabled	R/W

15.2.2 Event Link Setting Register n (ELSRn) (n = 7, 18 to 27, 33)

The ELSRn register specifies an event signal to be linked to for each peripheral module. Table 15.2 shows the correspondence between the ELSRn register and the peripheral modules. Table 15.3 shows the correspondence between the event signal names set in the ELSRn register and the signal numbers.

Address(es): ELSR7 A008 0B08h, ELSR18 A008 0B13h, ELSR19 A008 0B14h, ELSR20 A008 0B15h, ELSR21 A008 0B16h, ELSR22 A008 0B17h, ELSR23 A008 0B18h, ELSR24 A008 0B19h, ELSR25 A008 0B1Ah, ELSR26 A008 0B1Bh, ELSR27 A008 0B1Ch, ELSR33 A008 0B31h

	b7	b6	b5	b4	b3	b2	b1	b0
	ELS[7:0]							
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ELS[7:0]	Event Link Select	b7 b0 00000000: Event output to the corresponding peripheral module is disabled. 00011111 to 01111110: Set the number for the event signal to be linked. Settings other than above are prohibited.	R/W

Table 15.2 Correspondence between the ELSRn Register and the Peripheral Functions

Register Name	Peripheral Function (Module)
ELSR7	CMT1
ELSR18	Interrupt 1 (ELCIRQ1)
ELSR19	Interrupt 2 (ELCIRQ2)
ELSR20	Output port group 1
ELSR21	Output port group 2
ELSR22	Input port group 1
ELSR23	Input port group 2
ELSR24	Single port 0
ELSR25	Single port 1
ELSR26	Single-port 2
ELSR27	Single-port 3
ELSR33	CMTW0

Table 15.3 Correspondence between Event Signal Names Set in ELSRn.ELS[7:0] Bits and Signal Numbers

ELS[7:0] Bit Value	Peripheral modules	Name of Event Signal Set in ELSRn
1Fh	Compare match timer	CMT1 compare match 1
23h	EtherCAT slave controller	ESC Sync0
24h		ESC Sync1
52h	Serial peripheral interface	RSPI0 error (mode fault, overrun, or parity error)
53h		RSPI0 idle
54h		RSPI0 receive data full
55h		RSPI0 transmit data empty
56h		RSPI0 transmit end
63h	I/O ports	Input edge detection of input port group 1
64h		Input edge detection of input port group 2
65h		Input edge detection of single input port 0
66h		Input edge detection of single input port 1
67h		Input edge detection of single input port 2
68h		Input edge detection of single input port 3
69h	Event link controller	Software event
7Eh	Compare match timer W	CMTW channel 0 compare match
Settings other than above are prohibited.		

15.2.3 Event Link Option Setting Register C (ELOPC)

The ELOPC register determines the operation of CMT1 when an event is input. The event setting should be disabled when the ELC function is not to be used.

Address(es): A008 0B21h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	CMT1MD[1:0]		—	—
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b3, b2	CMT1MD[1:0]	CMT1 Operation Select	b3 b2 0 0: Counting is started. 0 1: The counter is cleared 1 0: Event counter 1 1: Event is disabled.	R/W
b7 to b4	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

15.2.4 Port Group Setting Register n (PGRn) (n = 1, 2)

The PGRn register specifies a group for I/O port bits. This register specifies each port bit in the same eight I/O ports as the member of a group. One to eight port bits can be specified as the members of the same group as required. Table 15.4 shows the PGRn register and corresponding ports.

Address(es): PGR1 A008 0B23h, PGR2 A008 0B24h

	b7	b6	b5	b4	b3	b2	b1	b0
	PGRn7	PGRn6	PGRn5	PGRn4	PGRn3	PGRn2	PGRn1	PGRn0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PGRn0	Port Group Setting n 0	0: The port bit is not specified as a member of the same group. 1: The port bit is specified as a member of the same group.	R/W
b1	PGRn1	Port Group Setting n 1		R/W
b2	PGRn2	Port Group Setting n 2		R/W
b3	PGRn3	Port Group Setting n 3		R/W
b4	PGRn4	Port Group Setting n 4		R/W
b5	PGRn5	Port Group Setting n 5		R/W
b6	PGRn6	Port Group Setting n 6		R/W
b7	PGRn7	Port Group Setting n 7		R/W

Table 15.4 Registers Related to Port Groups and Corresponding Port Numbers

Port Number	Port Group Setting Register (PGR)	Port Group Control Register (PGC)	Port Buffer Register (PDBF)
Port B	PGR1 register	PGC1 register	PDBF1 register
Port E	PGR2 register	PGC2 register	PDBF2 register

15.2.5 Port Group Control Register n (PGCn) (n = 1, 2)

For the output port group, the PGCn register specifies the form of outputting the signal externally via the port when the event signal is input. For the input port group, this register enables/disables overwriting of the PDBF register and specifies the conditions of event generation (edge of the externally input signal).
Refer to Table 15.4 for the PGRn register and corresponding ports.

Address(es): PGC1 A008 0B25h, PGC2 A008 0B26h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PGCOn[2:0]			—	PGCOVEn	PGCIn[1:0]	
Value after reset:	1	0	0	0	1	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	PGCIn [1:0]	Event Output Edge Select	b1 b0 0 0: Event is generated upon detection of the rising edge of the external input signal. 0 1: Event is generated upon detection of the falling edge of the external input signal. 1 X: Event is generated upon detection of both the rising and falling edges of the external input signal.	R/W
b2	PGCOVEn	PDBF Overwrite	0: Overwriting PDBFn register is disabled. 1: Overwriting PDBFn register is enabled.	R/W
b3	—	Reserved	This bit is read as 1. The write value should be 1.	R/W
b6 to b4	PGCOn [2:0]	Port Group Operation Select	b6 b4 0 0 0: 0 is output when the event is input. 0 0 1: 1 is output when the event is input. 0 1 0: The toggled (inverted) value is output when the event is input. 0 1 1: The buffer value is output when the event is input. 1 X X: The bit value is rotated out in the group (from MSB to LSB) when the event is input.	R/W
b7	—	Reserved	This bit is read as 1. The write value should be 1.	R/W

X: Don't care

15.2.6 Port Buffer Register n (PDBFn) (n = 1, 2)

The PDBFn register is an 8-bit readable/writable register used in combination with the PGRn register. Refer to section 15.3.4, I/O Port Operation upon Event Input and Event Generation for PDBFn register operations. Refer to Table 15.4 for the PDBFn register and corresponding ports.

Address(es): PDBF1 A008 0B27h, PDBF2 A008 0B28h

	b7	b6	b5	b4	b3	b2	b1	b0
	PDBFn 7	PDBFn 6	PDBFn 5	PDBFn 4	PDBFn 3	PDBFn 2	PDBFn 1	PDBFn 0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PDBFn0	Port Buffer n0	These bits handle the following operations in response to an input to or output from the port. • As an output port The value written to the PDBFn register is transferred to the PODR register. • As an input port The signal values on the external pins are transferred to the PDBFn register. Write access to the bit specified as a member of the input port group is invalid. For details, refer to section 15.3, Operation.	R/W
b1	PDBFn1	Port Buffer n1		R/W
b2	PDBFn2	Port Buffer n2		R/W
b3	PDBFn3	Port Buffer n3		R/W
b4	PDBFn4	Port Buffer n4		R/W
b5	PDBFn5	Port Buffer n5		R/W
b6	PDBFn6	Port Buffer n6		R/W
b7	PDBFn7	Port Buffer n7		R/W

15.2.7 Event Link Port Setting Register n (PELn) (n = 0 to 3)

The PELn register specifies the single port to which an event is to be linked, the port operation upon the event signal input, and the conditions of event generation. In this LSI, a total of 4 bits in port B or port E can be specified as single ports.

Address(es): PEL0 A008 0B29h, PEL1 A008 0B2Ah, PEL2 A008 0B2Bh, PEL3 A008 0B2Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	PSMn[1:0]		PSPn[1:0]		PSBn[2:0]		
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	PSBn[2:0]	Bit Number Specification	A bit number in eight I/O ports is specified.	R/W
b4, b3	PSPn[1:0]	Port Number Specification	b4 b3 0 0: Setting prohibited 0 1: Port B (corresponding to PGR1) 1 0: Port E (corresponding to PGR2) 1 1 1: Setting prohibited	R/W
b6, b5	PSMn[1:0]	Event Link Specification	• For the output port, data to be output from the port is specified. b6 b5 0 0: 0 is output when the event is input. 0 1: 1 is output when the event is input. 1 X: The toggled (inverted) value is output when the event is input. • For the input port, the edge on which the event is to be output is specified. b6 b5 0 0: Event is output upon detection of the rising edge. 0 1: Event is output upon detection of the falling edge. 1 X: Event is output upon detection of both the rising and falling edges.	R/W
b7	—	Reserved	This bit is read as 1. The write value should be 1.	R/W

X: Don't care

15.2.8 Event Link Software Event Generation Register (ELSEGR)

The ELSEGR register controls event generation by software.

Address(es): A008 0B2Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	WI	WE	—	—	—	—	—	SEG
Value after reset:	1	0	1	1	1	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	SEG	Software Event Generation	0: Normal operation 1: Software event is generated.	W
b5 to b1	—	Reserved	These bits are read as 1. The write value should be 1.	R/W
b6	WE	SEG Bit Write Enable	0: Write to SEG bit is disabled. 1: Write to SEG bit is enabled.	R/W
b7	WI	ELSEGR Register Write Disable	0: Write to ELSEGR register is enabled. 1: Write to ELSEGR register is disabled.	W

SEG Bit (Software Event Generation)

When 1 is written to this bit while the WE bit is 1, a software event is generated.
This bit is read as 0. Even if 1 is written to this bit, the data will not be stored.

WE Bit (SEG Bit Write Enable)

The SEG bit can be written to only when the WE bit is 1.
[Setting condition]
If 1 is written to this bit while the WI bit is 0, this bit becomes 1.
[Clearing condition]
If 0 is written to this bit while the WI bit is 0, this bit becomes 0.

WI Bit (ELSEGR Register Write Disable)

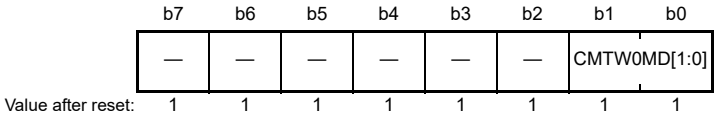
The ELSEGR register can be written to only when the value to be written to the WI bit is 0.
This bit is read as 1.

Note 1. The WE bit can only be updated by setting the WI bit and the WE bit at the same time.
Similarly, to update the SEG bit, the WI bit must be set at the same time; set WE = 1 by setting the WI bit and WE bit at the same time, and then set the WI bit and SEG bit. If WE = 1 when SEG is set, the WE bit retains the value 1.
However, setting the three bits at the same time while the current values are WI = 1, WE = 0, and SEG = 0 will not lead to the output of a software trigger. Since WE will only have the value 1 following the first time the three bits are set, a software trigger will only be generated by setting the three bits at the same time again.

15.2.9 Event Link Option Setting Register H (ELOPH)

The ELOPH register determines the operation of channel 0 in CMTW when an event is input. The event setting should be disabled when the ELC function is not to be used.

Address(es): A008 0B41h



Bit	Symbol	Bit Name	Description	R/W
b1, b0	CMTW0MD[1:0]	CMTW Operation Select	b1 b0 0 0: Counting is started. 0 1: Counting is cleared. 1 0: Event counter 1 1: Event is disabled.	R/W
b7 to b2	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

15.3 Operation

15.3.1 Relation between Interrupt Handling and Event Linking

The peripheral modules incorporated in this LSI are provided with the interrupt request status flags and the bits to enable/disable these interrupt requests. When an interrupt request is generated in a module, the corresponding interrupt request status flag is set. If the corresponding interrupt request is enabled then, the interrupt requested is issued to the CPU.
In contrast, the event link controller (ELC) uses the interrupt requests generated by various peripheral modules (excluding ESC) as event signals to connect (link) them to different modules, allowing direct cooperation between the modules without CPU intervention. Event signals can be output regardless of the setting of the corresponding interrupt request enable bit. Because ESC use interrupt signals as event signals, interrupt control must be permitted.
Figure 15.2 shows the relation between the interrupt handling and ELC.

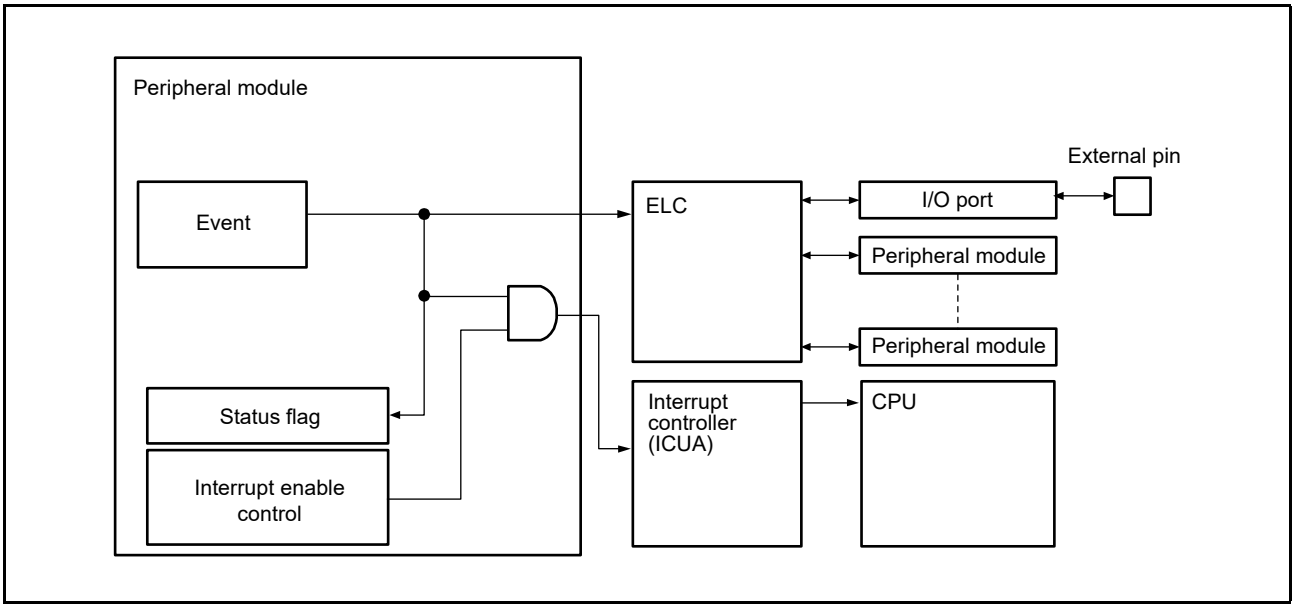


Figure 15.2 Relation between Interrupt Handling and ELC (Excluding the ESC)

15.3.2 Event Linkage

Set an event to the ELSRn register of the module to which the event is to be linked. On occurrence of the specified event, the link destination module performs the operation set in the ELOPm register (m = C, H). Only one type of event can be connected with one module. For detailed procedure for setting desired operation, see section 15.3.5, Example of Procedure for Linking Events. Table 15.5 lists the operations of modules when an event is input.

Table 15.5 Operations of Modules When Event is Input

Module	Operations When Event is Input	
CMT CMTW	Each timer operates differently depending on the setting of the ELOPm register (m = C, H) as below. • Starts counting when an event signal is input. • Clears counting when an event signal is input. Restarting count is allowed when the start bit of the timer is 1. • Counts the input events.	
I/O ports (output)	Port group	• Changes the PODR value to the value specified in the PGCn register. • Transfers the PDBFn value to the PODR register. • Rotates out the bit value.
	Single port	Changes the PODR value to the value specified in the PELn register.
I/O ports (input)	Port group	Transfers the signal value of the external pin to the PDBFn register.
	Single port	Event connection is not possible.
Interrupt controller	Issues an interrupt request to the CPU and starts DMACAa data transfer.	

15.3.3 Operation of Peripheral Timer Modules When Event is Input

The operation when an event signal is input is set by the ELOPm register (m = C, H).

(1) Counting Start Operation

When an event is input, the timer starts counting, which sets the count start bit*1 in each timer control register to 1. An event that is input while the count start bit is 1 is invalid.

(2) Counting Clear Operation

When an event is input, the timer counter is initialized. If the count start bit*1 in each timer control register is set to 1, the counting continues, thus, counting restarts.

(3) Event Counter Operation

Event input is selected as the timer clock source and the timer counts events.

Note 1. Refer to the register descriptions on starting the timer in the relevant peripheral timer module section.

15.3.4 I/O Port Operation upon Event Input and Event Generation

The I/O port operation to be performed upon event input to the port can be set and the operation causing the port to generate an event can be set.

(1) Single ports and Port Groups

There are two event link modes: event link to single ports and event link to port groups. In the former mode, events can be connected to eight I/O ports. In the latter mode, events can be connected to port groups consisting of any two or more bits in the same eight I/O ports.
A single port can be set by specifying any bit in the I/O port*1 to which an event can be connected using the PELn register. A port group can be set by specifying any one or more bits in the I/O port*1 to which an event can be connected using the PGRn register. One input port group and one output port group can be set in the same I/O port.
If the I/O port bit is specified as both a single port and a member of a port group, both functions are enabled when the relevant port is input, whereas only the port group function is enabled when the relevant port is output.
Set the PDR register to select the direction of the I/O ports.

Note 1. Port B and port E

(2) Single Input Port Operation upon Event Generation

A single input port which is specified by the PDR register generates an event when the signal value of the external pin connected to the relevant port changes. The event generation condition is specified using PELn register. An example of event linking operation by a single input port is shown as [1] in Figure 15.3.

(3) Single Output Port Operation upon Event Input

When an event is input to a single output port which is specified by the PDR register, the signal of the external pin connected to the relevant port changes according to the settings of PELn register. This changes the signal value of the external pin connected to the relevant port. An example of event linking operation by a single output port is shown as [2] in Figure 15.3.

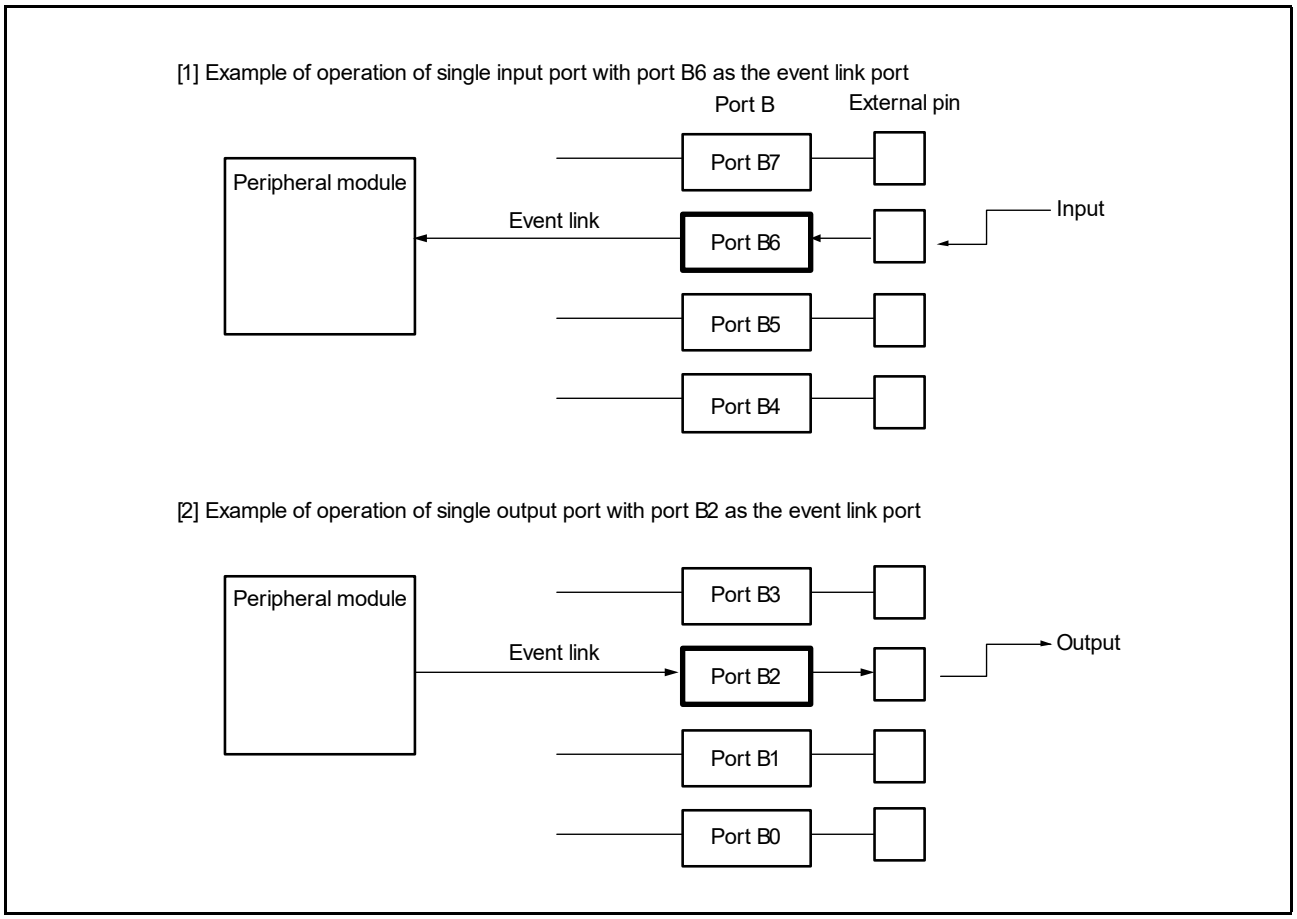


Figure 15.3 Event Linkage Related to Single Ports (Port B)

(4) Input Port Group Operation upon Event Generation

An input port group which is specified by the PDR register generates an event when the signal value of any one of the external pins connected to the relevant port group changes. The event generation condition is specified using the PCIn bit of the PGCn register.

(5) Input Port Group Operation upon Event Input

When an event is input to an input port group, the signal value of the external pin of the bit specified as a member of the input port group is transferred to the PDBFn register. If another event is input to the input port group in this state, operations are performed depending on the PGCn.PGCOVE bit setting as described below. Figure 15.4 shows the input port group operation upon an event input.

- PGCn.PGCOVEn = 0 (overwriting is disabled)
  If the value that was transferred to the PDBFn register upon the previous event input has already been read by the CPU, the signal value of the external pin is transferred to the PDBFn register. If not read, the signal value of the external pin is not transferred to the PDBFn register and the input event is invalid.
- PGCn.PGCOVEn = 1 (overwriting is enabled)
  When another event is input to an input port group, the signal value of the external pin is transferred to the PDBFn register.

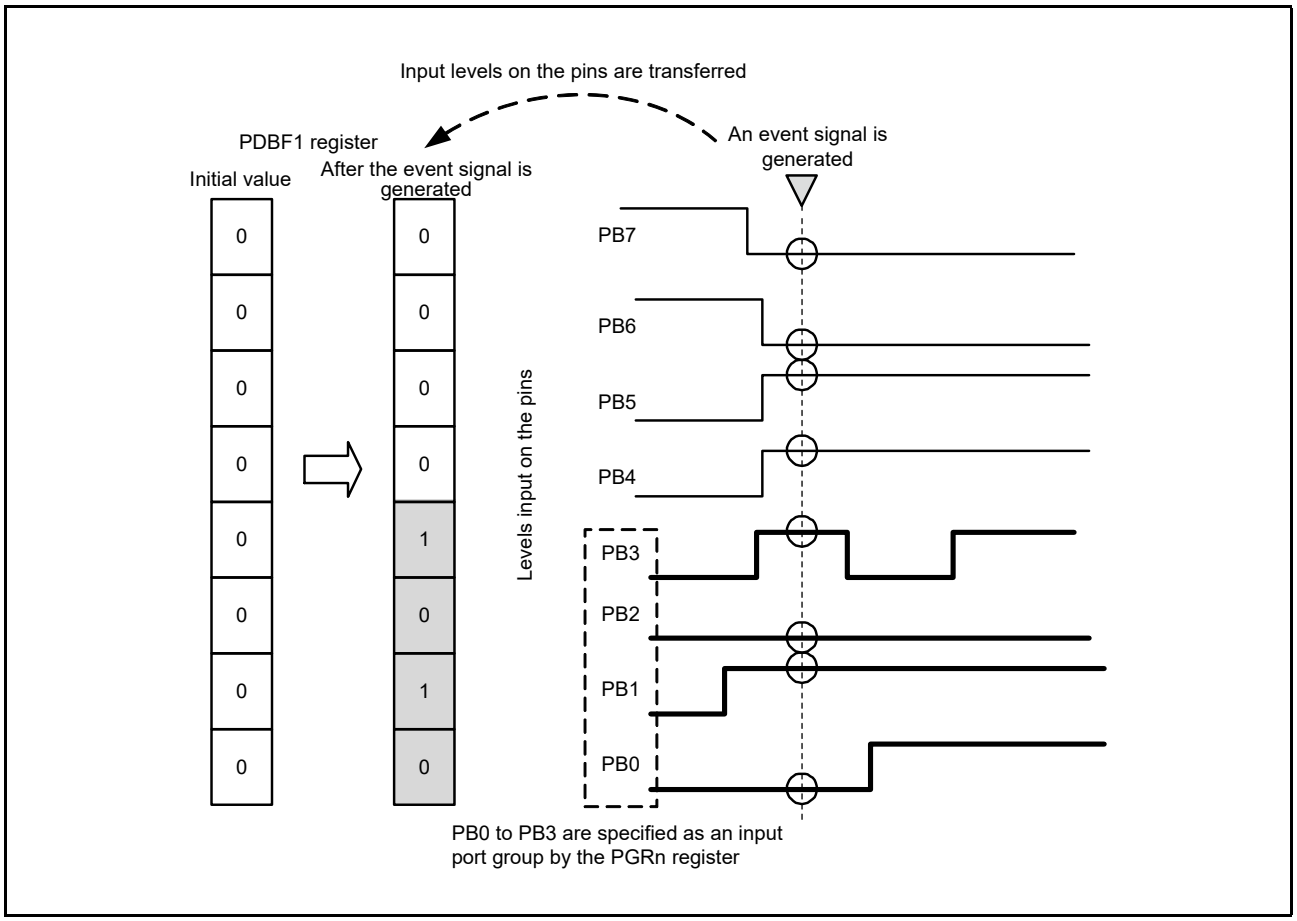


Figure 15.4 Input Port Group Operation upon Event Input (Port B)

(6) Output Port Group Operation upon Event Input

When an event is input to an output port group, the following operations are performed depending on the PGCn.PGCOn bit setting as described below.

- If an event is input to an output port group while the PGCn.PGCOn bit being 000b, 001b, or 010b, the PODR value is changed to the value which was specified in the PGCn register.
- If an event is input to an output port group while the PGCn.PGCOn bit being 011b, the PDBFn value is transferred to the PODR register of the port which was specified in the PGRn register. Example of operation of the output port group upon an event input (when PGCn.PGCOn = 011b) is shown in Figure 15.5.
- If an event is input to the output port group while the PGCn.PGCOn bit being 1XXb, the PDBFn value is transferred to the PODR register of the port which was specified in the PGRn register, and then the PODR value is rotated bit by bit from MSB to LSB. The initial value to be output to the port group should be provided in the PDBFn register in advance. Examples of bit-rotating operation of output port groups upon an event input (when PGCn.PGCOn = 1XXb) is shown in Figure 15.6.

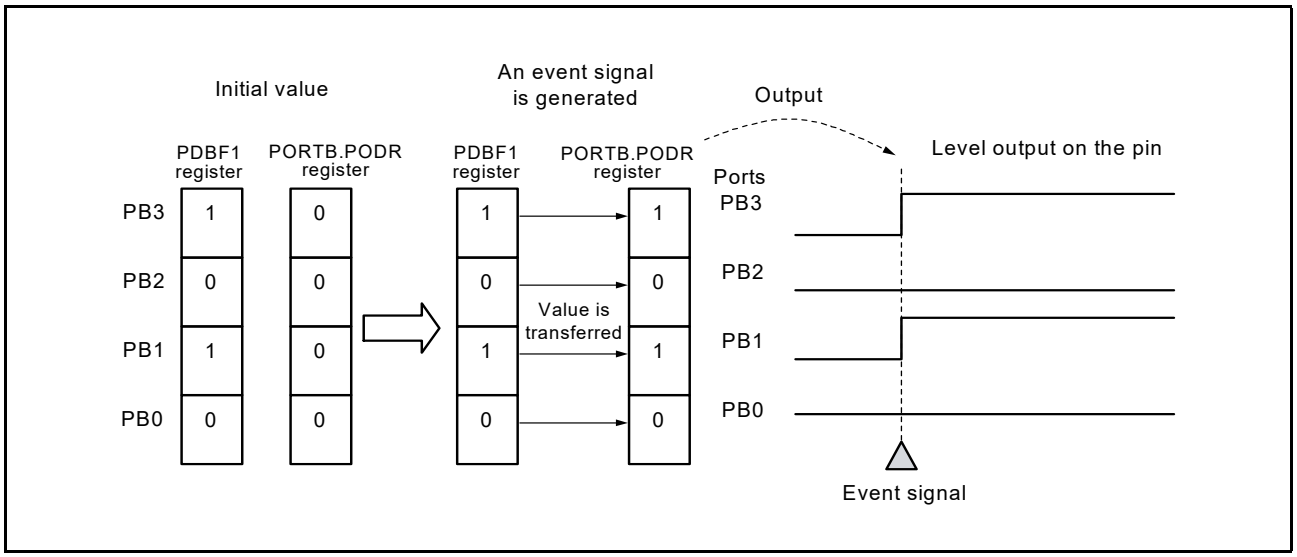


Figure 15.5 Event Linkage Related to Output Port Groups (Port B)

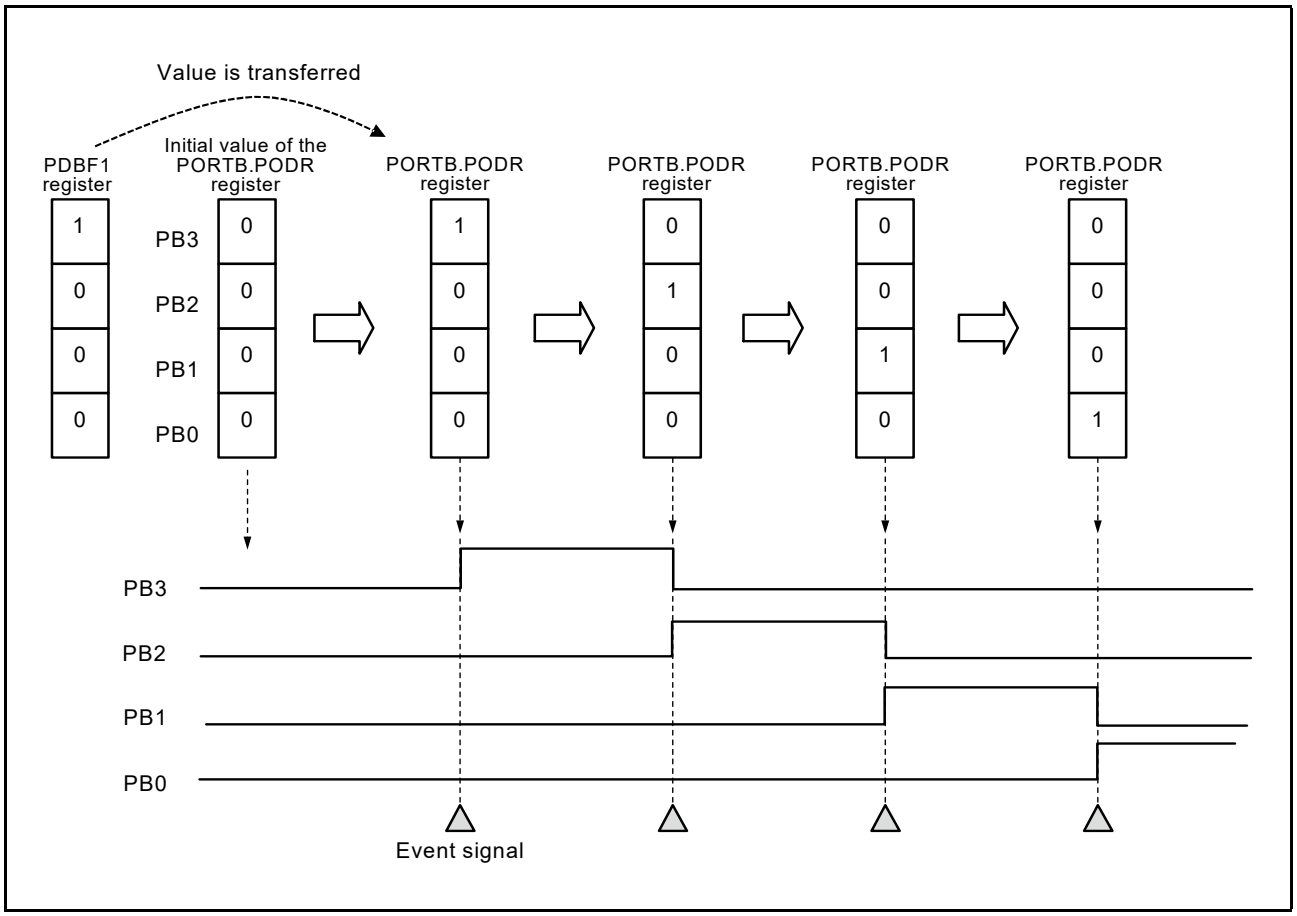


Figure 15.6 Bit-Rotating Operation of Output Port Groups (Port B)

(7) Restrictions on Writing to PODR and PDBFn Registers by a CPU

For event linkage through the I/O ports, following restrictions apply when a CPU writes the PODR and PDBFn registers.

- If port bits are specified as members of the input port group, write access to the relevant bits in the PDBFn register is invalid.
- If port bits are specified as members of the output port group, write access to the relevant bits in the PODR register is invalid.
- If a port bit is specified as a single output port and the event linkage is set (by the ELSRn register) for the port, write access to the relevant bit in the PODR register is invalid.

15.3.5 Example of Procedure for Linking Events

The following describes the procedure for linking events.

1. Set the operation of the module to which an event is to be linked.
2. If events are linked to ports, set the registers corresponding to the ports as below.

- I/O port setting
  PODR: Set the initial values of the output ports.
  PDR: Set the I/O direction of the ports.
- ELC setting
  PGRn: If ports are used as a port group, set the ports (in bit units) to be grouped.
  PGCn: Set the operation of the port group.
  PELn: If ports are used as single ports, set the ports, the operation of the ports when an event is input, and the condition when an event is generated.

Note: Setting the PDBFn register
(1) Output port groups
Set the PGCn register before setting the PDBFn register.
The value of the PGCn register can be changed if this precedes an event trigger which causes the value of the PODR register to change.
(2) Input port groups
Setting the PDBFn is not required. However, since the value after a reset is 00h, if the PDBFn register is used to confirm changes from H to L due to an event input, set the PDBFn bit for the pin for which you wish to confirm this to 1.

3. To the ELSRn register corresponding to the module to which an event signal is to be linked, set the number of the event signal.
4. If events are to be linked to timer modules, set the ELOPm register (m = C, H) corresponding to the timers as required.
5. Set the ELCR.ELCON bit to 1, which enables linkage of all the events.
6. Set the operation of the module from which an event is output, and activate the module. This allows the event output from the module to start the module to which an event is linked as preset.
7. To stop event linkage of independent modules, set 0000 0000b to the ELSRn.ELS[7:0] bits corresponding to the modules. To stop linkage of all the events, set the ELCR.ELCON bit to 0.

15.4 Usage Notes

15.4.1 Setting ELSR18 and ELSR19 Registers

For event linkage to an interrupt controller, specify the event signals to be set in the ELSR18 and ELSR19 registers from 63h to 7Eh. Setting any other values is prohibited.

15.4.2 Setting Bit-Rotating Operation of Output Port Groups

When the values of the PDBFn register are changed in the bit-rotating operation mode of the output port group, set the ELSRn register again after changing the PDBFn register value.

15.4.3 Setting Clocks

To link events, it is necessary for the ELC and the related modules to be enabled. The modules cannot operate if the related modules are in the module stop state or if the low power consumption mode causes the modules to stop (all-module stop mode).

15.4.4 Module Stop Function Setting

ELC operation can be disabled or enabled using module stop control register C (MSTPCRC). The initial setting allows the ELC to be stopped. Register access is enabled by canceling the module stop state. For details, see section 9, Low-Power Consumption Function.

16. I/O Ports

16.1 Overview

The pins of an I/O port function as general I/O port pins, I/O pins for peripheral modules, or interrupt input pins.
Each pin is also configurable as an I/O pin of a peripheral module or an input pin for an interrupt. All pins are set to non-use immediately after a reset (Hi-Z input protection), and pin functions can be switched by register settings. The setting of each pin is specified by the registers for the corresponding I/O port and peripheral modules.
Each port has the port direction register (PDR) that selects non-use, input, or output, the port output data register (PODR) that holds data for output, the port input register (PIDR) that indicates the pin states, the pull-up/pull-down control register (PCR) that controls enabling and disabling of the input pull-up/pull-down resistor, and the port mode register (PMR) that specifies the pin function of each port. For details on PMR, refer to section 17, Multi-Function Pin Controller (MPC).
Table 16.1 shows the specifications of I/O ports, and Table 16.2 lists the port functions.

Table 16.1 Specifications of I/O Ports

	Package	
Port	**196 Pins**	**Number of Pins**
PORT0	P00	1
PORT1	P10, P12, P16, P17	4
PORT2	P20 to P23, P25 to P27	7
PORT3	P33 to P35	3
PORT4	P40, P42, P44	3
PORT5	P50 to P54, P56	6
PORT6	P60 to P66	7
PORT7	P70 to P77	8
PORT8	P80 to P87	8
PORT9	P90 to P97	8
PORTA	PA0 to PA7	8
PORTB	PB0 to PB7	8
PORTC	PC0 to PC7	8
PORTD	PD5 to PD7	3
PORTE	PE0 to PE7	8
PORTF	PF5 to PF7	3
PORTG	PG2 to PG6	5
PORTJ	PJ0 to PJ7	8
PORTM	PM1 to PM7	7
PORTR	PR1	1
PORTS	PS0 to PS7	8
PORTU	PU7	1
	Total of pins	123

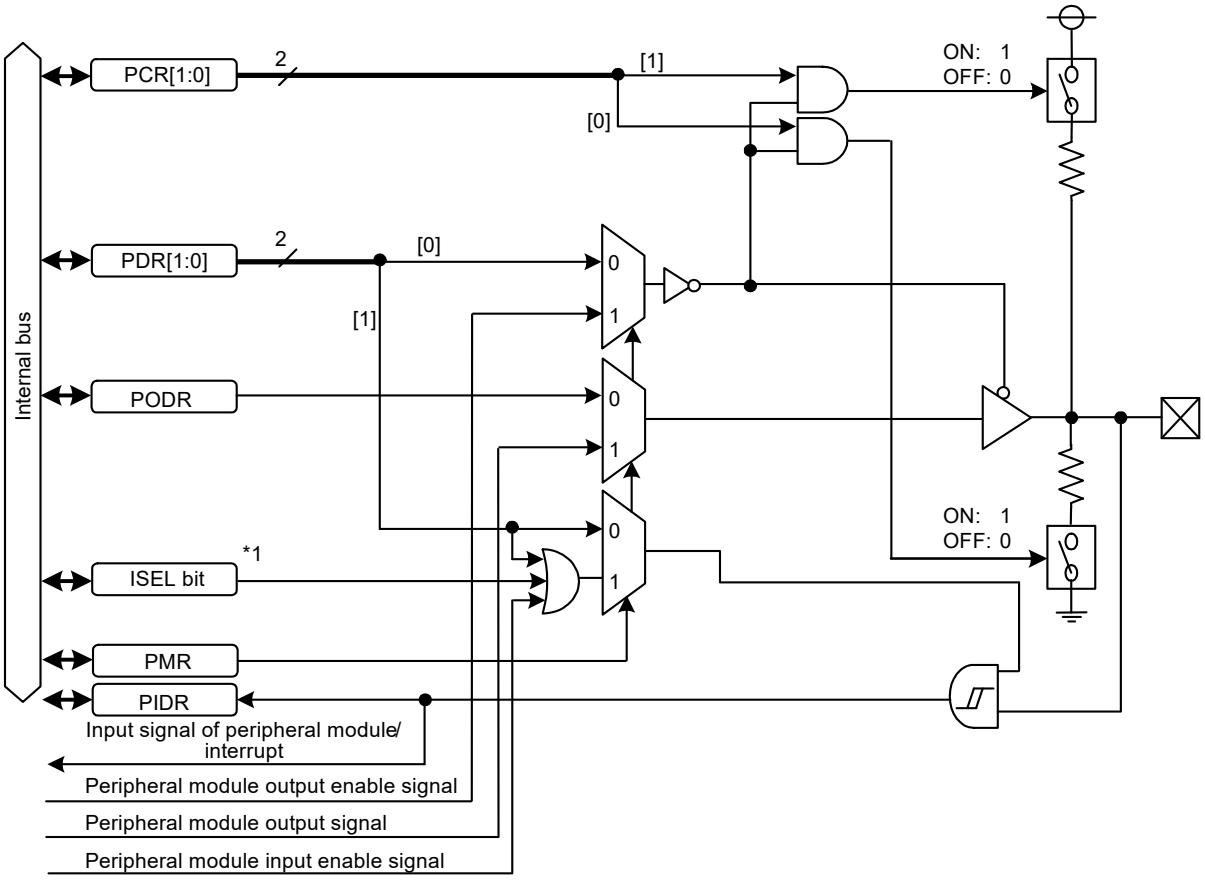
Table 16.2 Port Functions

Port	Pin	Input Pull-Up/Pull-Down	5-V Tolerant	Schmitt Input
PORT0	P00	√	—	—
PORT1	P10, P12, P16, P17	√	—	√
PORT2	P20 to P23, P25 to P27	√	—	√
PORT3	P33, P35	√	—	√
	P34	√	—	—
PORT4	P40, P42, P44	√	—	√
PORT5	P50 to P54, P56	√	—	√
PORT6	P60 to P66	√	—	√
PORT7	P70 to P77	√	—	—
PORT8	P80 to P87	√	—	√
PORT9	P90 to P97	√	—	√
PORTA	PA0 to PA7	√	—	—
PORTB	PB0 to PB7	√	—	√
PORTC	PC0 to PC7	—	√	√
PORTD	PD5 to PD7	√	—	√
PORTE	PE0 to PE7	√	—	—
PORTF	PF5 to PF7	√	—	√
PORTG	PG2 to PG6	√	—	√
PORTJ	PJ0 to PJ7	√	—	√
PORTM	PM1 to PM7	√	—	√
PORTR	PR1	√	—	√
PORTS	PS0 to PS7	√	—	√
PORTU	PU7	√	—	√

Specifying input pull-up/pull-down is available for other signals on pins that also function as general I/O pins.

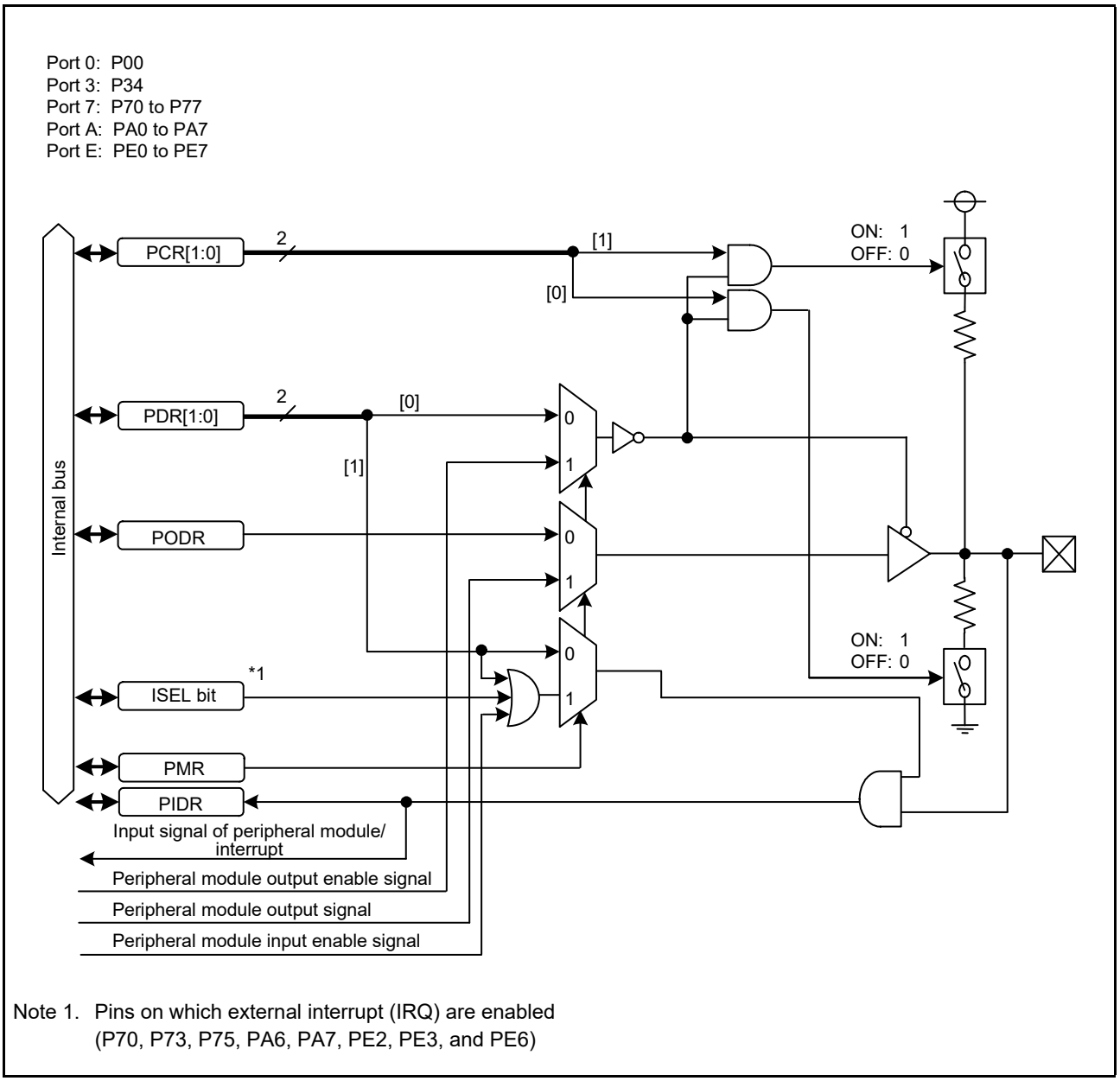
16.2 I/O Port Configuration

Port 1: P10, P12, P16, P17
Port 2: P20 to P23, P25 to P27
Port 3: P33, P35
Port 4: P40, P42, P44
Port 5: P50 to P54, P56
Port 6: P60 to P66
Port 8: P80 to P87
Port 9: P90 to P97
Port B: PB0 to PB7
Port D: PD5 to PD7
Port F: PF5 to PF7
Port G: PG2 to PG6
Port J: PJ0 to PJ7
Port M: PM1 to PM7
Port R: PR1
Port S: PS0 to PS7
Port U: PU7



Note 1. Pins on which external interrupt (NMI, IRQ) are enabled
(P10, P21, P22, P35, P44, P66, P83, P94, P95, P97, PB3, PF7, PM6, PR1, PS1, and PS6)

Figure 16.1 I/O Port Configuration (1)



Note 1. Pins on which external interrupt (IRQ) are enabled (P70, P73, P75, PA6, PA7, PE2, PE3, and PE6)

Figure 16.2 I/O Port Configuration (2)

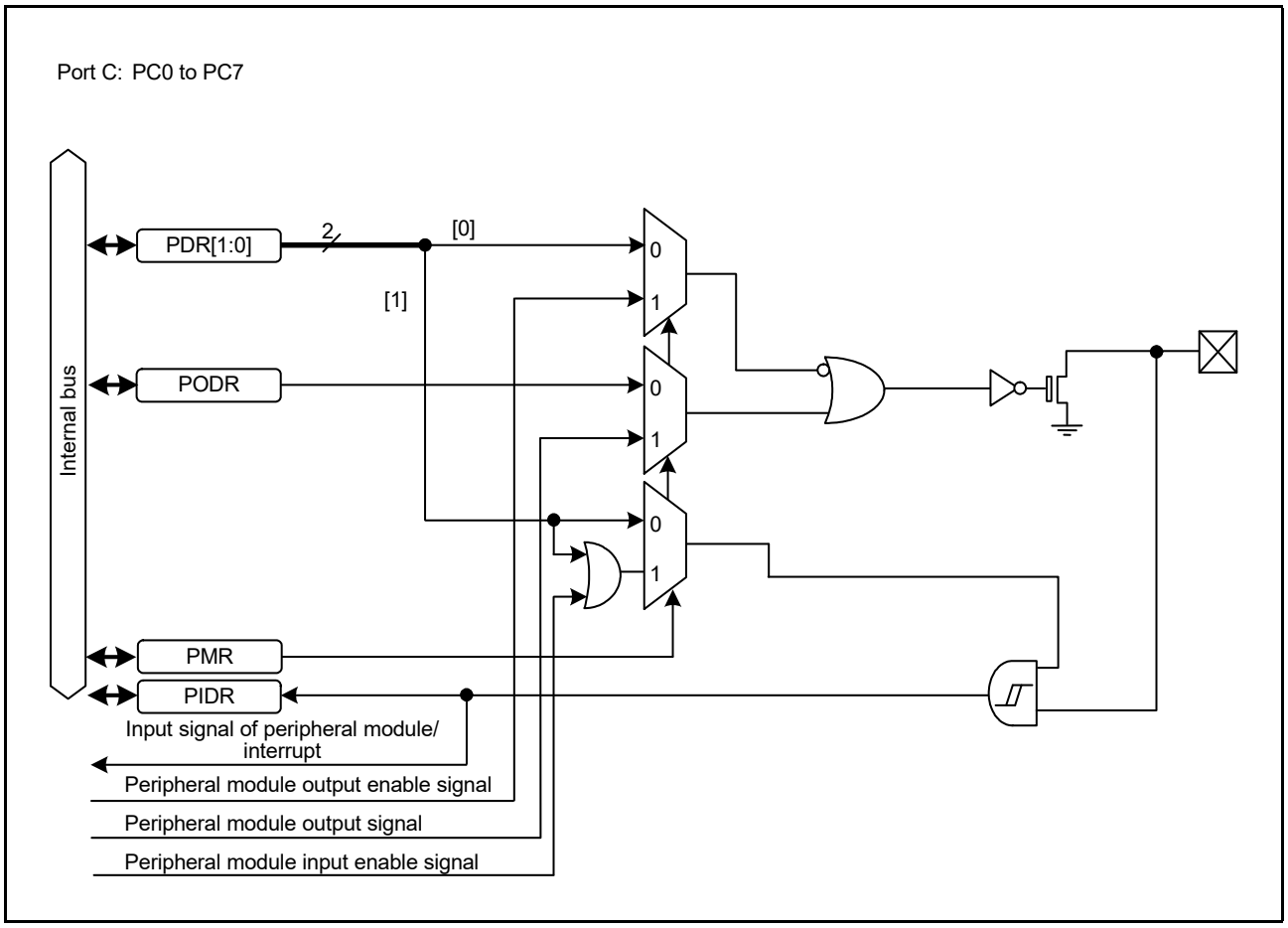


Figure 16.3 I/O Port Configuration (3)

16.3 Register Descriptions

16.3.1 Port Direction Register (PDR)

The PDR register is used to select non-use, input, or output (input enable) for individual pins of the corresponding port when the pins are configured as the general I/O pins. When 00 (non-use) is set to this register, this LSI can be protected from input Hi-Z state.

Address(es): PORT0.PDR A000 0000h, PORT1.PDR A000 0002h, PORT2.PDR A000 0004h, PORT3.PDR A000 0006h, PORT4.PDR A000 0008h, PORT5.PDR A000 000Ah, PORT6.PDR A000 000Ch, PORT7.PDR A000 000Eh, PORT8.PDR A000 0010h, PORT9.PDR A000 0012h, PORTA.PDR A000 0014h, PORTB.PDR A000 0016h, PORTC.PDR A000 0018h, PORTD.PDR A000 001Ah, PORTE.PDR A000 001Ch, PORTF.PDR A000 001Eh, PORTG.PDR A000 0020h, PORTJ.PDR A000 0024h, PORTM.PDR A000 002Ah, PORTR.PDR A000 0030h, PORTS.PDR A000 0032h, PORTU.PDR A000 0036h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	B7[1:0]		B6[1:0]		B5[1:0]		B4[1:0]		B3[1:0]		B2[1:0]		B1[1:0]		B0[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	B0[1:0]	Pm0 I/O Select	Odd bit Even bit 0 0: Non-use (Hi-Z input protection) 0 1: Setting prohibited 1 0: Input (functions as an input pin) 1 1: Output (functions as an output pin (port read enable))	R/W
b3, b2	B1[1:0]	Pm1 I/O Select		R/W
b5, b4	B2[1:0]	Pm2 I/O Select		R/W
b7, b6	B3[1:0]	Pm3 I/O Select		R/W
b9, b8	B4[1:0]	Pm4 I/O Select		R/W
b11, b10	B5[1:0]	Pm5 I/O Select		R/W
b13, b12	B6[1:0]	Pm6 I/O Select		R/W
b15, b14	B7[1:0]	Pm7 I/O Select		R/W

m = 0 to 9, A to G, J, M, R, S, U

Each bit of PORTm.PDR corresponds to each pin of port m; pin function can be specified in 2-bit units.
Write 00 (non-use) or 10 (input) to the PORTC.PDR.Bn (n = 0 to 7) bits because the PC0 to PC7 pins are input only.
The bit corresponding to a pin that does not exist is also reserved. A reserved bit is always read as 0. The write value should always be 0.

16.3.2 Port Output Data Register (PODR)

The PODR register holds the data to be output from the pins used for general I/O.

Address(es): PORT0.PODR A000 0040h, PORT1.PODR A000 0041h, PORT2.PODR A000 0042h, PORT3.PODR A000 0043h, PORT4.PODR A000 0044h, PORT5.PODR A000 0045h, PORT6.PODR A000 0046h, PORT7.PODR A000 0047h, PORT8.PODR A000 0048h, PORT9.PODR A000 0049h, PORTA.PODR A000 004Ah, PORTB.PODR A000 004Bh, PORTC.PODR A000 004Ch, PORTD.PODR A000 004Dh, PORTE.PODR A000 004Eh, PORTF.PODR A000 004Fh, PORTG.PODR A000 0050h, PORTJ.PODR A000 0052h, PORTM.PODR A000 0055h, PORTR.PODR A000 0058h, PORTS.PODR A000 0059h, PORTU.PODR A000 005Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 Output Data Store	0: Low output 1: High output	R/W
b1	B1	Pm1 Output Data Store		R/W
b2	B2	Pm2 Output Data Store		R/W
b3	B3	Pm3 Output Data Store		R/W
b4	B4	Pm4 Output Data Store		R/W
b5	B5	Pm5 Output Data Store		R/W
b6	B6	Pm6 Output Data Store		R/W
b7	B7	Pm7 Output Data Store		R/W

m = 0 to 9, A to G, J, M, R, S, U

The PORTC.PODR.Bn (n = 0 to 7) bits are reserved because the PC0 to PC7 pins are input only. The bit corresponding to a pin that does not exist is reserved. A reserved bit is always read as 0. The write value should always be 0.

16.3.3 Port Input Data Register (PIDR)

The PIDR register reflects the states of the individual input port pins.

Address(es): PORT0.PIDR A000 0060h, PORT1.PIDR A000 0061h, PORT2.PIDR A000 0062h, PORT3.PIDR A000 0063h, PORT4.PIDR A000 0064h, PORT5.PIDR A000 0065h, PORT6.PIDR A000 0066h, PORT7.PIDR A000 0067h, PORT8.PIDR A000 0068h, PORT9.PIDR A000 0069h, PORTA.PIDR A000 006Ah, PORTB.PIDR A000 006Bh, PORTC.PIDR A000 006Ch, PORTD.PIDR A000 006Dh, PORTE.PIDR A000 006Eh, PORTF.PIDR A000 006Fh, PORTG.PIDR A000 0070h, PORTH.PIDR A000 0072h, PORTM.PIDR A000 0075h, PORTR.PIDR A000 0078h, PORTS.PIDR A000 0079h, PORTU.PIDR A000 007Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
Value after reset:	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0	0: Low input 1: High input	R
b1	B1	Pm1		R
b2	B2	Pm2		R
b3	B3	Pm3		R
b4	B4	Pm4		R
b5	B5	Pm5		R
b6	B6	Pm6		R
b7	B7	Pm7		R

m = 0 to 9, A to G, J, M, R, S, U

If PORTm.PDR is set to 10 or 11, the pin states of ports m can be read with PORTm.PIDR, regardless of the values of PORTm.PMR.

The bit corresponding to a pin that does not exist is reserved. A reserved bit is read as undefined, and cannot be modified.

16.3.4 Port Mode Register (PMR)

The PMR register specifies the function of the pins of the port.

Address(es): PORT0.PMR A000 0080h, PORT1.PMR A000 0081h, PORT2.PMR A000 0082h, PORT3.PMR A000 0083h, PORT4.PMR A000 0084h, PORT5.PMR A000 0085h, PORT6.PMR A000 0086h, PORT7.PMR A000 0087h, PORT8.PMR A000 0088h, PORT9.PMR A000 0089h, PORTA.PMR A000 008Ah, PORTB.PMR A000 008Bh, PORTC.PMR A000 008Ch, PORTD.PMR A000 008Dh, PORTE.PMR A000 008Eh, PORTF.PMR A000 008Fh, PORTG.PMR A000 0090h, PORTJ.PMR A000 0092h, PORTM.PMR A000 0095h, PORTR.PMR A000 0098h, PORTS.PMR A000 0099h, PORTU.PMR A000 009Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	B7	B6	B5	B4	B3	B2	B1	B0
Value after reset: *1	0	0	0	0	0	0	0	0

Note 1. The PMR register value for port 3 after a reset is 18h.

Bit	Symbol	Bit Name	Description	R/W
b0	B0	Pm0 Pin Mode Control	0: Uses the pin as a general I/O pin 1: Uses the pin as an I/O port for peripheral functions	R/W
b1	B1	Pm1 Pin Mode Control		R/W
b2	B2	Pm2 Pin Mode Control		R/W
b3	B3	Pm3 Pin Mode Control		R/W
b4	B4	Pm4 Pin Mode Control		R/W
b5	B5	Pm5 Pin Mode Control		R/W
b6	B6	Pm6 Pin Mode Control		R/W
b7	B7	Pm7 Pin Mode Control		R/W

m = 0 to 9, A to G, J, M, R, S, U

Each bit of PORTm.PMR corresponds to each pin of port m; pin function can be specified in 1-bit units. Be sure to write 0 (general I/O port) to these bits. The bit corresponding to a pin that does not exist is reserved. A reserved bit is always read as 0. The write value should always be 0.

16.3.5 Pull-Up/Pull-Down Control Register (PCR)

The PCR register enables or disables an input pull-up or pull-down resistor for each pin of the port.
When a pin for the general port or peripheral module is in the input state, the input pull-up resistor connected to the pin with the corresponding bit in PORTm.PCR set to 10 is enabled, and the input pull-down resistor connected to the pin with the corresponding bit in PORTm.PCR set to 01 is enabled.
When a pin is set as a general port output pin or a peripheral module output pin, the pull-up resistor for the pin is disabled regardless of the settings of PCR. Note that the PC0 to PC7 pins do not have his function.
The pull-up and pull-down resistors are also disabled in the reset state.

Address(es): PORT0.PCR A000 0100h, PORT1.PCR A000 0102h, PORT2.PCR A000 0104h, PORT3.PCR A000 0106h, PORT4.PCR A000 0108h, PORT5.PCR A000 010Ah, PORT6.PCR A000 010Ch, PORT7.PCR A000 010Eh, PORT8.PCR A000 0110h, PORT9.PCR A000 0112h, PORTA.PCR A000 0114h, PORTB.PCR A000 0116h, PORTD.PCR A000 011Ah, PORTE.PCR A000 011Ch, PORTF.PCR A000 011Eh, PORTG.PCR A000 0120h, PORTJ.PCR A000 0124h, PORTM.PCR A000 012Ah, PORTR.PCR A000 0130h, PORTS.PCR A000 0132h, PORTU.PCR A000 0136h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	B7[1:0]		B6[1:0]		B5[1:0]		B4[1:0]		B3[1:0]		B2[1:0]		B1[1:0]		B0[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	B0[1:0]	Pm0 Input Pull-Up/Pull-Down Resistor Control	Odd bit Even bit 0 0: Disables an input pull-up and pull-down resistors 0 1: Enables an input pull-down resistor 1 0: Enables an input pull-up resistor 1 1: Setting prohibited	R/W
b3, b2	B1[1:0]	Pm1 Input Pull-Up/Pull-Down Resistor Control		R/W
b5, b4	B2[1:0]	Pm2 Input Pull-Up/Pull-Down Resistor Control		R/W
b7, b6	B3[1:0]	Pm3 Input Pull-Up/Pull-Down Resistor Control		R/W
b9, b8	B4[1:0]	Pm4 Input Pull-Up/Pull-Down Resistor Control		R/W
b11, b10	B5[1:0]	Pm5 Input Pull-Up/Pull-Down Resistor Control		R/W
b13, b12	B6[1:0]	Pm6 Input Pull-Up/Pull-Down Resistor Control		R/W
b15, b14	B7[1:0]	Pm7 Input Pull-Up/Pull-Down Resistor Control		R/W

m = 0 to 9, A, B, D to G, J, M, R, S, U

The bit corresponding to a pin that does not exist is also reserved. A reserved bit is always read as 00b. The write value should always be 00b.

16.4 Handling of Unused Pins

Table 16.3 lists the details of handling of unused pins.

Table 16.3 Handling of Unused Pins

Pin Name	Handling
ERROROUT#, RSTOUT#	Keep these pins open.
TCK	Connect this pin to VSS via a resistor (pulling down).
TRST#	Connect this pin to VSS via a resistor (pulling down), or input the same signal as that on the RES# pin.
TMS, Port 34 (TDI)	Connect these pins to VCCQ33 via a resistor (pulling up).
Port 0 to Port 9, Port A to Port G, Port J, Port M, Port R, Port S, Port U (except Port 34)*1	Keep these pins open, connect them to VCCQ33 via a resistor (pulling up), or connect them to VSS via a resistor (pulling down).
USB0_DP, USB0_DM	Keep these pins open.
USB_RREF	Keep these pins open.

Note 1. When handling them as unused pins, set the corresponding bits of the port direction register (PDR) to "Non-use (Hi-Z input protection)" which is the value after reset release.

17. Multi-Function Pin Controller (MPC)

17.1 Overview

This LSI configures an I/O pin or an interrupt pin of peripheral functions to be multiplexed with multiple ports. The multi-function pin controller (MPC) is a module that selects I/O pins and interrupt pins for the peripheral function to use from multiple ports, and then assigns the function to the selected pins.
Table 17.1 lists the multiplexed pin configurations. Selecting a single function for multiple pins is prohibited.

Table 17.1 List of Multiplexed Pin Configurations (1 / 6)

Module/Function	Channel	Pin Function	Allocation Port
Debugging interface		TDI (input)	P34
		TDO (output)	P33
		TRACECLK (output)	P10
			P70
		TRACECTL (output)	P00
			P71
		TRACEDATA0 (output)	P72
			PE0
		TRACEDATA1 (output)	P73
			PE1
		TRACEDATA2 (output)	P74
			PE2
		TRACEDATA3 (output)	P75
			PE3
		TRACEDATA4 (output)	P76
			PE4
		TRACEDATA5 (output)	P77
			PE5
		TRACEDATA6 (output)	PA0
			PE6
		TRACEDATA7 (output)	PA1
			PE7

Table 17.1 List of Multiplexed Pin Configurations (2 / 6)

Module/Function	Channel	Pin Function	Allocation Port
Interrupt	NMI	NMI (input)	P35
	IRQ0	IRQ0 (input)	P10
			P70
	IRQ1	IRQ1 (input)	P21
			PS1
	IRQ2	IRQ2 (input)	P22
			PE2
	IRQ3	IRQ3 (input)	P73
			PB3
			PE3
	IRQ4	IRQ4 (input)	P94
	IRQ6	IRQ6 (input)	PA6
			PE6
			PM6
	IRQ7	IRQ7 (input)	P97
			PA7
			PF7
	IRQ9	IRQ9 (input)	PR1
	IRQ11	IRQ11 (input)	P83
	IRQ12	IRQ12 (input)	P44
	IRQ13	IRQ13 (input)	P75
			P95
	IRQ14	IRQ14 (input)	P66
			PS6
Compare match timer W	CMTW0	TOC0 (output)	PG2
		TIC0 (input)	PC7
	CMTW1	TOC1 (output)	PG4
		TIC1 (input)	PG3
	CMTW2	TOC2 (output)	P71
		TIC2 (input)	P72
	CMTW3	TOC3 (output)	P92
		TIC3 (input)	P93

Table 17.1 List of Multiplexed Pin Configurations (3 / 6)

Module/Function	Channel	Pin Function	Allocation Port
FIFO on-chip serial communication interface (SCIFA)	SCIFA0	RXD0 (input)	P42
		TXD0 (output)	P23
			P40
		SCK0 (input/output)	P22
		CTS0# (input/output)	P21
			P44
		RTS0# (output)	P27
	SCIFA1	RXD1 (input)	P73
			PE6
		TXD1 (output)	P72
			PE5
		SCK1 (input/output)	P71
			PE7
		CTS1# (input/output)	P74
			PE3
			PR1
		RTS1# (output)	P70
			PE4
	SCIFA2	RXD2 (input)	P92
			PA4
			PS6
		TXD2 (output)	P91
			PA5
			PS7
		SCK2 (input/output)	P93
			PA3
		CTS2# (input/output)	P95
			PA6
		RTS2# (output)	P94
			PA7
	SCIFA3	RXD3 (input)	PB4
		TXD3 (output)	PB3
	SCIFA4	RXD4 (input)	P84
		TXD4 (output)	P83
			P90
I^2C bus interface	RIIC1	SCL1 (input/output)	PC6
		SDA1 (input/output)	PC7

Table 17.1 List of Multiplexed Pin Configurations (4 / 6)

Module/Function	Channel	Pin Function	Allocation Port
EtherCAT slave controller (MDIO controller)	MDIOC0	CLKOUT25M0 (output)	P85
		ETH0_TXC (input)	PC2
		ETH0_TXEN (output)	P82
		ETH0_TXD0 (output)	PJ3
		ETH0_TXD1 (output)	PJ2
		ETH0_TXD2 (output)	PJ1
		ETH0_TXD3 (output)	PJ0
		ETH0_INT (input)	P52
			PA5
		ETH0_RXC (input)	PC3
		ETH0_RXDV (input)	P80
		ETH0_RXER (input)	P81
			PB4
		ETH0_RXD0 (input)	PJ4
		ETH0_RXD1 (input)	PJ5
		ETH0_RXD2 (input)	PJ6
		ETH0_RXD3 (input)	PJ7
		PHYLINK0 (input)	P50
		ETH_MDC (output)	PB6
		ETH_MDIO (input/output)	PB5
	MDIOC1	CLKOUT25M1 (output)	P54
		ETH1_TXC (input)	P87
		ETH1_TXEN (output)	PF5
		ETH1_TXD0 (output)	P86
		ETH1_TXD1 (output)	PD7
		ETH1_TXD2 (output)	PD6
		ETH1_TXD3 (output)	PD5
		ETH1_INT (input)	P53
			PA4
		ETH1_RXC (input)	PB2
		ETH1_RXDV (input)	PB0
		ETH1_RXER (input)	PB1
		ETH1_RXD0 (input)	PF6
		ETH1_RXD1 (input)	PB7
		ETH1_RXD2 (input)	PC0
		ETH1_RXD3 (input)	PC1
		PHYLINK1 (input)	P51

Table 17.1 List of Multiplexed Pin Configurations (5 / 6)

Module/Function	Channel	Pin Function	Allocation Port
EtherCAT slave controller	ESC0	CATLEDRUN (output)	PM4
		CATIRQ (output)	PU7
		CATLEDSTER (output)	PM5
		CATLEDERR (output)	PM1
		CATLINKACT0 (output)	P83
			PM6
		CATLINKACT1 (output)	P84
			PM7
		CATSYNC1 (output)	PM2
		CATSYNC0 (output)	PB4
			PM3
		CATLATCH1 (input)	PM2
		CATLATCH0 (input)	PB4
			PM3
		CATI2CCLK (input/output)	PC4
		CATI2CDATA (input/output)	PC5
		PHYRESETOUT# (output)	P17
			PB3
USB2.0 host/ function module	USB	USB_VBUSIN (input)	PC6
		USB_VBUSEN (output)	P66
		USB_OVRCUR (input)	P70
CAN module	CAN1	CRXD1 (input)	PC7
		CTXD1 (output)	P61
			P66
			PB3
Serial peripheral interface	RSPI0	RSPCK0 (input/output)	P77
			PE7
		MOSI0 (input/output)	PA0
			PE5
		MISO0 (input/output)	PA1
			PE6
		SSL00 (input/output)	P75
			PE4
		SSL01 (output)	P76
			PE3
		SSL02 (output)	PA2
			PE2
		SSL03 (output)	P74
			PE1
	RSPI1	RSPCK1 (input/output)	PG2
		MOSI1 (input/output)	PG4
		MISO1 (input/output)	PG3
		SSL10 (input/output)	PG5
		SSL11 (output)	PG6

Table 17.1 List of Multiplexed Pin Configurations (6 / 6)

Module/Function	Channel	Pin Function	Allocation Port
SPI multi-IO bus controller		SPBCLK (output)	P62
		SPBMO/SPBIO0 (input/output)	P63
		SPBMI/SPBIO1 (input/output)	P64
		SPBIO2 (input/output)	P65
		SPBIO3 (input/output)	P61
		SPBSSL (output)	P60

17.2 Register Descriptions

The registers and bits of unsupported pins, depending on the package, are reserved. The write value to the reserved bits is the value after a reset.

17.2.1 Write-Protect Register (PWPR)

The PWPR register enables or disables writing to the PFS register and the PFSWE bit of the PWPR register.

Address(es): A000 02FFh

	b7	b6	b5	b4	b3	b2	b1	b0
	B0WI	PFSWE	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	PFSWE	PFS Register Write Enable	0: Writing to the PFS register is disabled. 1: Writing to the PFS register is enabled.	R/W
b7	B0WI	PFSWE Bit Write Disable	0: Writing to the PFSWE bit is enabled. 1: Writing to the PFSWE bit is disabled.	R/W

PFSWE Bit (PFS Register Write Enable)

Writing to the PmnPFS register (m = 0 to 9, A to G, J, M, R, S, U, and n = 0 to 7) is enabled only when the PFSWE bit is set to 1.

To set the PFSWE bit to 1, write 0 to the B0WI bit, and then set 1 to the PFSWE bit.

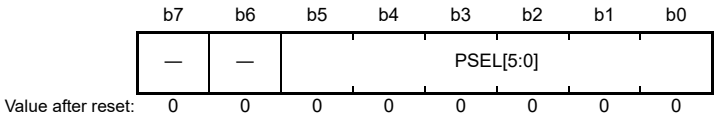
B0WI Bit (PFSWE Bit Write Disable)

Only when the B0WI bit is set to 0, writing to the PFSWE bit is enabled.

17.2.2 P0n Pin Function Control Register (P0nPFS) (n = 0)

The P0n pin function control register (P0nPFS) selects the function of the pin to use.
The P0nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): P00PFS A000 0200h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.2.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.2 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

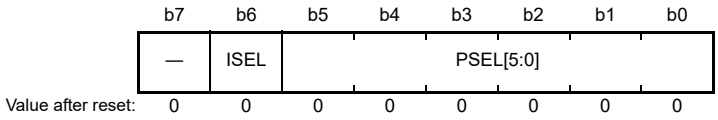
	Pin
PSEL[5:0] Setting	**P00**
000000b (Value after reset)	Hi-Z
100111b	TRACECTL

Note: Do not set values other than those listed above.

17.2.3 P1n Pin Function Control Register (P1nPFS) (n = 0, 7)

The P1n pin function control register (P1nPFS) selects the function of the pin to use.
The P1nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): P10PFS A000 0208h, P17PFS A000 020Fh



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.3.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ input pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.3 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

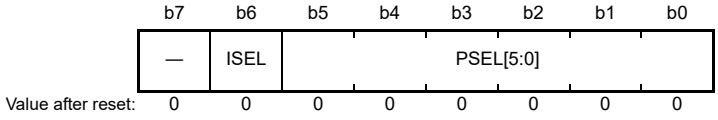
	Pin	
PSEL[5:0] Setting	**P10**	**P17**
000000b (Value after reset)	Hi-Z	
010110b	—	PHYRESETOUT#
100111b	TRACECLK	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.4 P2n Pin Function Control Register (P2nPFS) (n = 1 to 3, 7)

The P2n pin function control register (P2nPFS) selects the function of the pin to use. The P2nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): P21PFS A000 0211h, P22PFS A000 0212h, P23PFS A000 0213h, P27PFS A000 0217h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.4.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.4 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

	Pin			
PSEL[5:0] Setting	**P21**	**P22**	**P23**	**P27**
000000b (Value after reset)	Hi-Z			
001010b	—	—	—	RTS0#
001011b	CTS0#	SCK0	TXD0	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.5 P3n Pin Function Control Register (P3nPFS) (n = 3 to 5)

The P3n pin function control register (P3nPFS) selects the function of the pin to use. The P3nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written. No peripheral functions are assigned to P35. To use P35 as the NMI pin, see section 12.3.4, NMI Pin Interrupts.

Address(es): P33PFS A000 021Bh, P34PFS A000 021Ch, P35PFS A000 021Dh

	b7	b6	b5	b4	b3	b2	b1	b0	
	—	ISEL	PSEL[5:0]						
Value after reset:	0	0	0	0	0	0	0	0	*1
	0	0	1	0	0	1	1	1	*2

Note 1. P35PFS
Note 2. P33PFS, P34PFS

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.5.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin or NMI input pin. 1: Use as the IRQn input pin or NMI input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ input pin or the NMI input pin (P35). This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.5 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

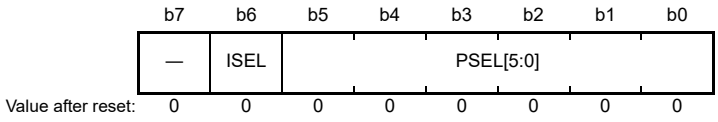
	Pin	
PSEL[5:0] Setting	**P33**	**P34**
000000b (Value after reset)*1	Hi-Z	
100111b (Value after reset)*2	TDO	TDI

Note 1. For pin P35
Note 2. For pins P33 and P34
Note: Do not set values other than those listed above.

17.2.6 P4n Pin Function Control Register (P4nPFS) (n = 0, 2, 4)

The P4n pin function control register (P4nPFS) selects the function of the pin to use. The P4nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): P40PFS A000 0220h, P42PFS A000 0222h, P44PFS A000 0224h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.6.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.6 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

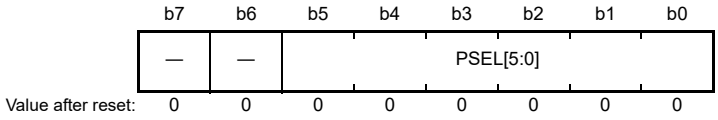
	Pin		
PSEL[5:0] Setting	**P40**	**P42**	**P44**
000000b (Value after reset)	Hi-Z		
001010b	TXD0	RXD0	CTS0#

Note: Do not set values other than those listed above.

17.2.7 P5n Pin Function Control Register (P5nPFS) (n = 0 to 4)

The P5n pin function control register (P5nPFS) selects the function of the pin to use. The P5nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): P50PFS A000 0228h, P51PFS A000 0229h, P52PFS A000 022Ah, P53PFS A000 022Bh, P54PFS A000 022Ch



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.7.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.7 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

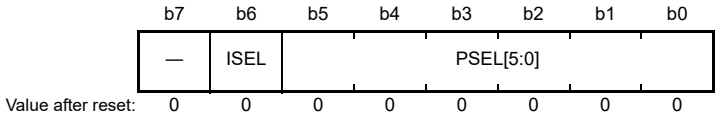
	Pin				
PSEL[5:0] Setting	**P50**	**P51**	**P52**	**P53**	**P54**
000000b (Value after reset)	Hi-Z				
010001b	PHYLINK0	PHYLINK1	ETH0_INT	ETH1_INT	CLKOUT25M1

Note: Do not set values other than those listed above.

17.2.8 P6n Pin Function Control Register (P6nPFS) (n = 0 to 6)

The P6n pin function control register (P6nPFS) selects the function of the pin to use. The P6nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): P60PFS A000 0230h, P61PFS A000 0231h, P62PFS A000 0232h, P63PFS A000 0233h, P64PFS A000 0234h, P65PFS A000 0235h, P66PFS A000 0236h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.8.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.8 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

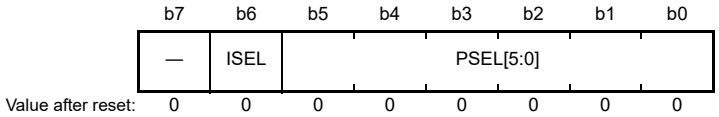
	Pin						
PSEL[5:0] Setting	**P60**	**P61**	**P62**	**P63**	**P64**	**P65**	**P66**
000000b (Value after reset)	Hi-Z						
010000b	—	CTXD1	—	—	—	—	CTXD1
011001b	—	—	—	—	—	—	USB_VBUSEN
011011b	SPBSSL	SPBIO3	SPBCLK	SPBMO/SPBIO0	SPBMI/SPBIO1	SPBIO2	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.9 P7n Pin Function Control Register (P7nPFS) (n = 0 to 7)

The P7n pin function control register (P7nPFS) selects the function of the pin to use. The P7nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): P70PFS A000 0238h, P71PFS A000 0239h, P72PFS A000 023Ah,P73PFS A000 023Bh, P74PFS A000 023Ch, P75PFS A000 023Dh, P76PFS A000 023Eh, P77PFS A000 023Fh



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.9.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.9 Register Settings for the Input/Output Function in the 196-pin FBGA Pins

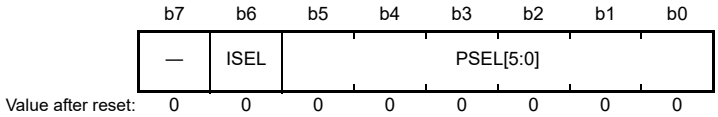
PSEL[5:0] Setting	Pin							
	P70	P71	P72	P73	P74	P75	P76	P77
000000b (Value after reset)	Hi-Z							
001010b	RTS1#	SCK1	TXD1	RXD1	CTS1#	—	—	—
001101b	—	—	—	—	SSL03	SSL00	SSL01	RSPCK0
011001b	USB_OVRCUR	—	—	—	—	—	—	—
011101b	—	TOC2	TIC2	—	—	—	—	—
100111b	TRACECLK	TRACECTL	TRACEDATA0	TRACEDATA1	TRACEDATA2	TRACEDATA3	TRACEDATA4	TRACEDATA5

Note: —: Do not set.
Do not set values other than those listed above.

17.2.10 P8n Pin Function Control Register (P8nPFS) (n = 0 to 7)

The P8n pin function control register (P8nPFS) selects the function of the pin to use. The P8nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of a pint without the IRQn function is reserved. A value after a reset must be written.

Address(es): P80PFS A000 0240h, P81PFS A000 0241h, P82PFS A000 0242h, P83PFS A000 0243h, P84PFS A000 0244h, P85PFS A000 0245h, P86PFS A000 0246h, P87PFS A000 0247h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.10.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.10 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

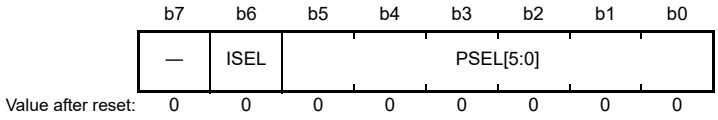
PSEL[5:0] Setting	Pin							
	P80	P81	P82	P83	P84	P85	P86	P87
000000b (Value after reset)	Hi-Z							
001011b	—	—	RTS3#	TXD4	RXD4	SCK4	—	—
010001b	ETH0_RXDV	ETH0_RXER	ETH0_TXEN	—	—	CLKOUT25M 0	ETH1_TXD0	ETH1_TXC
010101b	—	—	—	CATLINKACT 0	CATLINKACT 1	—	—	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.11 P9n Pin Function Control Register (P9nPFS) (n = 0 to 5, 7)

The P9n pin function control register (P9nPFS) selects the function of the pin to use. The P9nPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of a pint without the IRQn function is reserved. A value after a reset must be written.

Address(es): P90PFS A000 0248h, P91PFS A000 0249h, P92PFS A000 024Ah, P93PFS A000 024Bh, P94PFS A000 024Ch, P95PFS A000 024Dh, P97PFS A000 024Fh



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.11.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.11 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

	Pin						
PSEL[5:0] Setting	**P90**	**P91**	**P92**	**P93**	**P94**	**P95**	**P97**
000000b (Value after reset)	Hi-Z						
001011b	—	TXD2	RXD2	SCK2	RTS2#	CTS2#	—
001100b	TXD4	—	—	—	—	—	—
011101b	—	—	TOC3	TIC3	—	—	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.12 PAn Pin Function Control Register (PAnPFS) (n = 0 to 7)

The PAn pin function control register (PAnPFS) selects the function of the pin to use. The PAnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): PA0PFS A000 0250h, PA1PFS A000 0251h, PA2PFS A000 0252h, PA3PFS A000 0253h, PA4PFS A000 0254h, PA5PFS A000 0255h, PA6PFS A000 0256h, PA7PFS A000 0257h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.12.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.12 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

PSEL[5:0] Setting	Pin							
	PA0	PA1	PA2	PA3	PA4	PA5	PA6	PA7
000000b (Value after reset)	Hi-Z							
001010b	—	—	—	SCK2	RXD2	TXD2	CTS2#	RTS2#
001101b	MOSI0	MISO0	SSL02	—	—	—	—	—
010001b	—	—	—	—	—	ETH0_INT	—	—
010010b	—	—	—	—	ETH1_INT	—	—	—
100111b	TRACEDATA6	TRACEDATA7	—	—	—	—	—	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.13 PBn Pin Function Control Register (PBnPFS) (n = 0 to 7)

The PBn pin function control register (PBnPFS) selects the function of the pin to use. The PBnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): PB0PFS A000 0258h, PB1PFS A000 0259h, PB2PFS A000 025Ah, PB3PFS A000 025Bh, PB4PFS A000 025Ch, PB5PFS A000 025Dh, PB6PFS A000 025Eh, PB7PFS A000 025Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.13.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.13 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

PSEL[5:0] Setting	Pin							
	PB0	PB1	PB2	PB3	PB4	PB5	PB6	PB7
000000b (Value after reset)	Hi-Z							
001010b	—	—	—	TXD3	RXD3	—	—	—
010000b	—	—	—	CTXD1	—	—	—	—
010001b	ETH1_RXDV	ETH1_RXER	ETH1_RXC	—	—	ETH_MDIO	ETH_MDC	ETH1_RXD1
010010b	—	—	—	—	ETH0_RXER	—	—	—
010101b	—	—	—	PHYRESETOUT#	CATSYNC0	—	—	—
010110b	—	—	—	—	CATLATCH0	—	—	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.14 PCn Pin Function Control Register (PCnPFS) (n = 0 to 7)

The PCn pin function control register (PCnPFS) selects the function of the pin to use. The PCnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): PC0PFS A000 0260h, PC1PFS A000 0261h, PC2PFS A000 0262h, PC3PFS A000 0263h, PC4PFS A000 0264h, PC5PFS A000 0265h, PC6PFS A000 0266h, PC7PFS A000 0267h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.14.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.14 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

PSEL[5:0] Setting	Pin							
	PC0	PC1	PC2	PC3	PC4	PC5	PC6	PC7
000000b (Value after reset)	Hi-Z							
001111b	—	—	—	—	—	—	SCL1	SDA1
010000b	—	—	—	—	—	—	—	CRXD1
010001b	ETH1_RXD2	ETH1_RXD3	ETH0_TXC	ETH0_RXC	—	—	—	—
010110b	—	—	—	—	CATI2CCLK	CATI2CDATA	—	—
011001b	—	—	—	—	—	—	USB_VBUSIN	—
011101b	—	—	—	—	—	—	—	TIC0

Note: —: Do not set.
Do not set values other than those listed above.

17.2.15 PDn Pin Function Control Register (PDnPFS) (n = 5 to 7)

The PDn pin function control register (PDnPFS) selects the function of the pin to use. The PDnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): PD5PFS A000 026Dh, PD6PFS A000 026Eh, PD7PFS A000 026Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.15.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.15 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

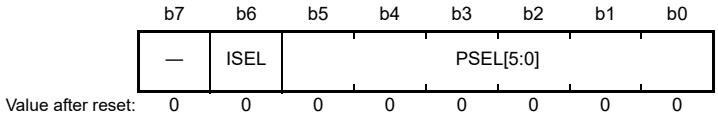
	Pin		
PSEL[5:0] Setting	**PD5**	**PD6**	**PD7**
000000b (Value after reset)	Hi-Z		
010001b	ETH1_TXD3	ETH1_TXD2	ETH1_TXD1

Note: Do not set values other than those listed above.

17.2.16 PEn Pin Function Control Register (PEnPFS) (n = 0 to 7)

The PEn pin function control register (PEnPFS) selects the function of the pin to use. The PEnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): PE0PFS A000 0270h, PE1PFS A000 0271h, PE2PFS A000 0272h, PE3PFS A000 0273h, PE4PFS A000 0274h, PE5PFS A000 0275h, PE6PFS A000 0276h, PE7PFS A000 0277h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.16.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.16 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

PSEL[5:0] Setting	Pin							
	PE0	PE1	PE2	PE3	PE4	PE5	PE6	PE7
000000b (Value after reset)	Hi-Z							
001100b	—	—	—	CTS1#	RTS1#	TXD1	RXD1	SCK1
001110b	—	SSL03	SSL02	SSL01	SSL00	MOSI0	MISO0	RSPCK0
100111b	TRACEDATA0	TRACEDATA1	TRACEDATA2	TRACEDATA3	TRACEDATA4	TRACEDATA5	TRACEDATA6	TRACEDATA7

Note: —: Do not set.
Do not set values other than those listed above.

17.2.17 PFn Pin Function Control Register (PFnPFS) (n = 5 to 7)

The PFn pin function control register (PFnPFS) selects the function of the pin to use. The PFnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): PF5PFS: A000 027Dh, PF6PFS: A000 027Eh, PF7PFS: A000 027Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each pin function, see Table 17.17.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.17 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

	Pin		
PSEL[5:0] Setting	**PF5**	**PF6**	**PF7**
000000b (Value after reset)	Hi-Z		
010001b	ETH1_TXEN	ETH1_RXD0	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.18 PGn Pin Function Control Register (PGnPFS) (n = 2 to 6)

The PGn pin function control register (PGnPFS) selects the function of the pin to use. The PGnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): PG2PFS A000 0282h, PG3PFS A000 0283h, PG4PFS A000 0284h, PG5PFS A000 0285h, PG6PFS A000 0286h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.18.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.18 Register Settings for the Input/Output Function in the 196-pin FBGA Pins

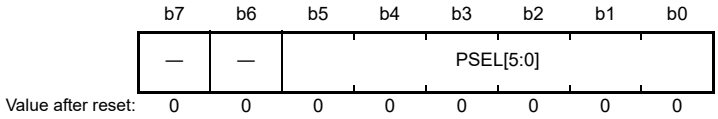
	Pin				
PSEL[5:0] Setting	**PG2**	**PG3**	**PG4**	**PG5**	**PG6**
000000b (Value after reset)	Hi-Z				
001101b	RSPCK1	MISO1	MOSI1	SSL10	SSL11
011101b	TOC0	TIC1	TOC1	—	—

Note: —: Do not set.
Do not set values other than those listed above.

17.2.19 PJn Pin Function Control Register (PJnPFS) (n = 0 to 7)

The PJn pin function control register (PJnPFS) selects the function of the pin to use. The PJnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): PJ0PFS A000 0290h, PJ1PFS A000 0291h, PJ2PFS A000 0292h, PJ3PFS A000 0293h, PJ4PFS A000 0294h, PJ5PFS A000 0295h, PJ6PFS A000 0296h, PJ7PFS A000 0297h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.19.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.19 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

PSEL[5:0] Setting	Pin							
	PJ0	PJ1	PJ2	PJ3	PJ4	PJ5	PJ6	PJ7
000000b (Value after reset)	Hi-Z							
010001b	ETH0_TXD3	ETH0_TXD2	ETH0_TXD1	ETH0_TXD0	ETH0_RXD0	ETH0_RXD1	ETH0_RXD2	ETH0_RXD3

Note: Do not set values other than those listed above.

17.2.20 PMn Pin Function Control Register (PMnPFS) (n = 1 to 7)

The PMn pin function control register (PMnPFS) selects the function of the pin to use. The PMnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): PM1PFS A000 02A9h, PM2PFS A000 02AAh, PM3PFS A000 02ABh, PM4PFS A000 02ACh, PM5PFS A000 02ADh, PM6PFS A000 02AEh, PM7PFS A000 02AFh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	ISEL	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.20.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.20 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

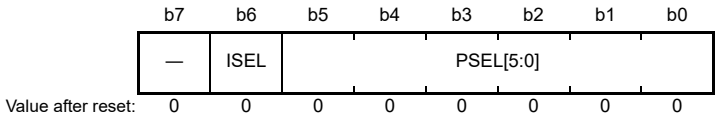
	Pin						
PSEL[5:0] Setting	**PM1**	**PM2**	**PM3**	**PM4**	**PM5**	**PM6**	**PM7**
000000b (Value after reset)	Hi-Z						
010101b	—	CATSYNC1	CATSYNC0	—	—	—	—
010110b	CATLEDERR	CATLATCH1	CATLATCH0	CATLEDRUN	CATLEDSTER	CATLINKACT0	CATLINKACT1

Note: —: Do not set.
Do not set values other than those listed above.

17.2.21 PRn Pin Function Control Register (PRnPFS) (n = 1)

The PRn pin function control register (PRnPFS) selects the function of the pin to use. The PRnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): PR1PFS A000 02C1h



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.21.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.21 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

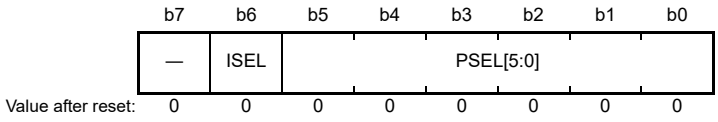
	Pin
PSEL[5:0] Setting	**PR1**
000000b (Value after reset)	Hi-Z
001011b	CTS1#

Note: Do not set values other than those listed above.

17.2.22 PSn Pin Function Control Register (PSnPFS) (n = 1, 6, 7)

The PSn pin function control register (PSnPFS) selects the function of the pin to use. The PSnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection. The ISEL bit of the pin without the IRQn function is reserved. A value after a reset must be written.

Address(es): PS1PFS A000 02C9h, PS6PFS A000 02CEh, PS7PFS A000 02CFh



Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.22.	R/W
b6	ISEL	Interrupt Input Function Select	0: Do not use as the IRQn input pin. 1: Use as the IRQn input pin.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

ISEL Bit (Interrupt Input Function Select)

Set this bit to 1 to use it as the IRQ pin. This bit can be used with peripheral functions. Note that IRQn (external pin interrupt) with the same number cannot be enabled for two or more pins.

Table 17.22 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

	Pin		
PSEL[5:0] Setting	**PS1**	**PS6**	**PS7**
000000b (Value after reset)	Hi-Z		
001100b	—	RXD2	TXD2

Note: —: Do not set.
Do not set values other than those listed above.

17.2.23 PUn Pin Function Control Register (PUnPFS) (n = 7)

The PUn pin function control register (PUnPFS) selects the function of the pin to use. The PUnPFS register is protected by the write protect register (PWPR). Modifications to this register can only be made after releasing protection.

Address(es): PU7PFS A000 02DFh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PSEL[5:0]					
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b5 to b0	PSEL[5:0]	Pin Function Select	These bits select the peripheral function. For details on each function, see Table 17.23.	R/W
b7, b6	—	Reserved	Thes bits are read as 0. The write value should be 0.	R/W

PSEL[5:0] Bits (Pin Function Select)

These bits select the peripheral function to assign to a pin.

Table 17.23 Register Settings for the Input/Output Function in the 196-pin FBGA Pin

	Pin
PSEL[5:0] Setting	**PU7**
000000b (Value after reset)	Hi-Z
010110b	CATIRQ

Note: Do not set values other than those listed above.

17.3 Usage Notes

17.3.1 Procedure for Specifying the Pin Input/Output Function

To specify the pin input/output function:

1. Set the port direction register (PDR) of the applicable pin to 00, and clear the port mode register (PMR) to 0 to set them as a general I/O port.
2. For each peripheral module, set the I/O signal to assign to the applicable pin.
3. Clear the PWPR.B0WI bit to 0, and then set the PWPR.PFSWE bit to 1. By doing so, make the Pmn pin function control register (PmnPFS) (m = 0 to 9, A to G, J, M, R, S, U, n = 0 to 7) writable.
4. By using the PmnPFS.PSEL[5:0] bits, set the pin input/output function.
5. Clear the PWPR.PFSWE bit to 0 to disable writing to the PmnPFS register.
6. Set the applicable bit of the PMR register corresponding to the selected pin to 1 as necessary to switch to the pin input/output function of the peripheral function.
7. Set the PDR register to 10 as necessary to enable reading the port status.

17.3.2 Notes on MPC Register Setting

1. Settings of the Pmn pin function control register (PmnPFS) (m = 0 to 9, A to G, J, M, R, S, U, n = 0 to 7) should be made only while the PMR register for the target pin is cleared to 0. If the PmnPFS is set while the applicable bit of the PMR register is 1, unexpected edges might be input for the input function. Besides, unexpected pulses might be output for the output function.
2. When setting the PmnPFS.ISEL bit to use the IRQ or NMI pin interrupt, follow the procedure described in [For IRQ pins] of section 12.3.3, External Pin Interrupts and section 12.3.4, NMI Pin Interrupts. If the PmnPFS.ISEL bit is set using a different procedure, unexpected edges may be input, leading to malfunction.
3. Only the allowed functions should be specified for the PmnPFS register. If a value that is not allowed for the register is specified, correct operation is not guaranteed.
4. Do not assign a single function to multiple pins through the MPC settings.
5. Points to note regarding the port mode register (PMR), port direction register (PDR), and Pmn pin function control register (PmnPFS) settings for pins that have multiplexed pin functions are listed in Table 17.24. Ensure that the bit corresponding to the applicable pin of the PMR register is 0 when changes to the PSEL[5:0] bits are made.

Table 17.24 Register Settings

Item	PMR.Bn	PDR.Bn[1:0]	PmnPFS		Note
			ISEL	PSEL[5:0]	
After a reset is canceled	0 *1	00	0	000000b *1	In the disabled (Hi-Z input protection) state after a reset is canceled.
Not used	0	00	0	N	
General I/O port	0	10/11 *2	0/1 *3	N	Set the PmnPFS.ISEL bit to 1 if these are multiplexed with interrupt inputs.
Peripheral functions	1	00/10 *4	0/1 *3	Peripheral functions (See Table 17.2 to Table 17.23.)	Set the PmnPFS.ISEL bit to 1 if it is multiplexed with interrupt inputs. Set the PDR.Bn[1:0] bits to 10 if it is multiplexed with the port read function (reading the pin status of the port in the PIDR.Bn bit).
Interrupt input (NMI, IRQ0 to 4, 6, 7, 9, 11 to 14)	0	10	1	N	

N: Setting is not required.

Note 1. Values after reset for PORT3.PMR, P33PFS.PSEL[5:0], and P34PFS.PSEL[5:0] are different.
For details on the PSEL[5:0] bits, see section 17.2.5, P3n Pin Function Control Register (P3nPFS) (n = 3 to 5).
For details on PORT3.PMR, see section 16.3.4, Port Mode Register (PMR).

Note 2. Setting the PDR.Bn[1:0] bits to 10 makes the register function as a general input port.
Setting the PDR.Bn[1:0] bits to 11 makes the register function as a general output port.

Note 3. If the PmnPFS.ISEL bit is set to 0, the register does not function as an IRQ pin.
Setting the PmnPFS.ISEL bit to 1 makes the register function as an IRQ pin (when the IRQ function is multiplexed).

Note 4. If the PDR.Bn[1:0] bits is set to 00, the port read function (reading the pin status of the port with the PIDR.Bn bit) cannot be used.
Setting the PDR.Bn[1:0] bits to 10 enables reading of the port pin status.

17.3.3 Usage Notes on Port Read Function

When a peripheral module for a pin which is bidirectional or an output is in use and the value of the PDR bits for the given pin is changed from 00 (initial value) to 10 (input enabled), the state of the pin can be read (port read function) from the PIDR register while the peripheral function is in use.

When a peripheral module for a pin which is an input or any of the input/output pin functions listed in Table 17.25 is in use, input is always enabled and the pin function can be used in parallel with port reading without changing the setting of the PDR register.

A shoot-through current flows when an external pin enters the Hi-Z state. An external pin must be pulled down or up when in the Hi-Z state.

Table 17.25 List of Modules and Associated Pin Functions for Which Input is Always Enabled

Module/Function	Channel	Pin Function
FIFO internal serial communication interface (SCIFA)	SCIFA0	RXD0 (Input)
		SCK0 (Input/output)
		CTS0# (Input/output)
		RTS0# (Output)
	SCIFA1	RXD1 (Input)
		SCK1 (Input/output)
		CTS1# (Input/output)
		RTS1# (Output)
	SCIFA2	RXD2 (Input)
		SCK2 (Input/output)
		CTS2# (Input/output)
		RTS2# (Output)
	SCI3FA	RXD3 (Input)
	SCI4FA	RXD4 (Input)
EtherCAT slave controller (MDIO controller)	MDIO0	ETH_MDIO (Input/output)
EtherCAT slave controller	ESC0	CATI2CCLK (Input/output)
Serial peripheral interface	RSPI0	RSPCK0 (Input/output)
		MOSI0 (Input/output)
		MISO0 (Input/output)
		SSL00 (Input/output)
	RSPI1	RSPCK1 (Input/output)
		MOSI1 (Input/output)
		MISO1 (Input/output)
		SSL10 (Input/output)
		SSL11 (Output)
SPI multi I/O bus controller		SPBMO/SPBIO0 (Input/output)
		SPBMI/SPBIO1 (Input/output)
		SPBIO2 (Input/output)
		SPBIO3 (Input/output)

18. Compare Match Timer (CMT)

The compare match timer (CMT) consists of a two-channel 16-bit timer, and can generate interrupts at set intervals by using its 16-bit counter.

18.1 Overview

Table 18.1 lists the specifications for the CMT.
Figure 18.1 shows a block diagram of the CMT.

Table 18.1 CMT Specifications

Item	Description
Number of internal channels	Two channels × three units
Timer counter (per channel)	16-bit up counter (Counted according to the count enable signal output by the prescaler.) Returned to 0000h after compare match.
Prescaler (per channel)	9-bit counter (Linked with enabling/disabling of timer counter operation.) • Outputs four types of count enable signals. The type can be selected from PCLKD/8, PCLKD/32, PCLKD/128, and PCLKD/512.
Event link function (only channel 1 of unit 0)	One of the following three operations is possible depending on the received event: • Count start • Event count • Count clear This function can issue a compare match event.
Reset	Asynchronous reset

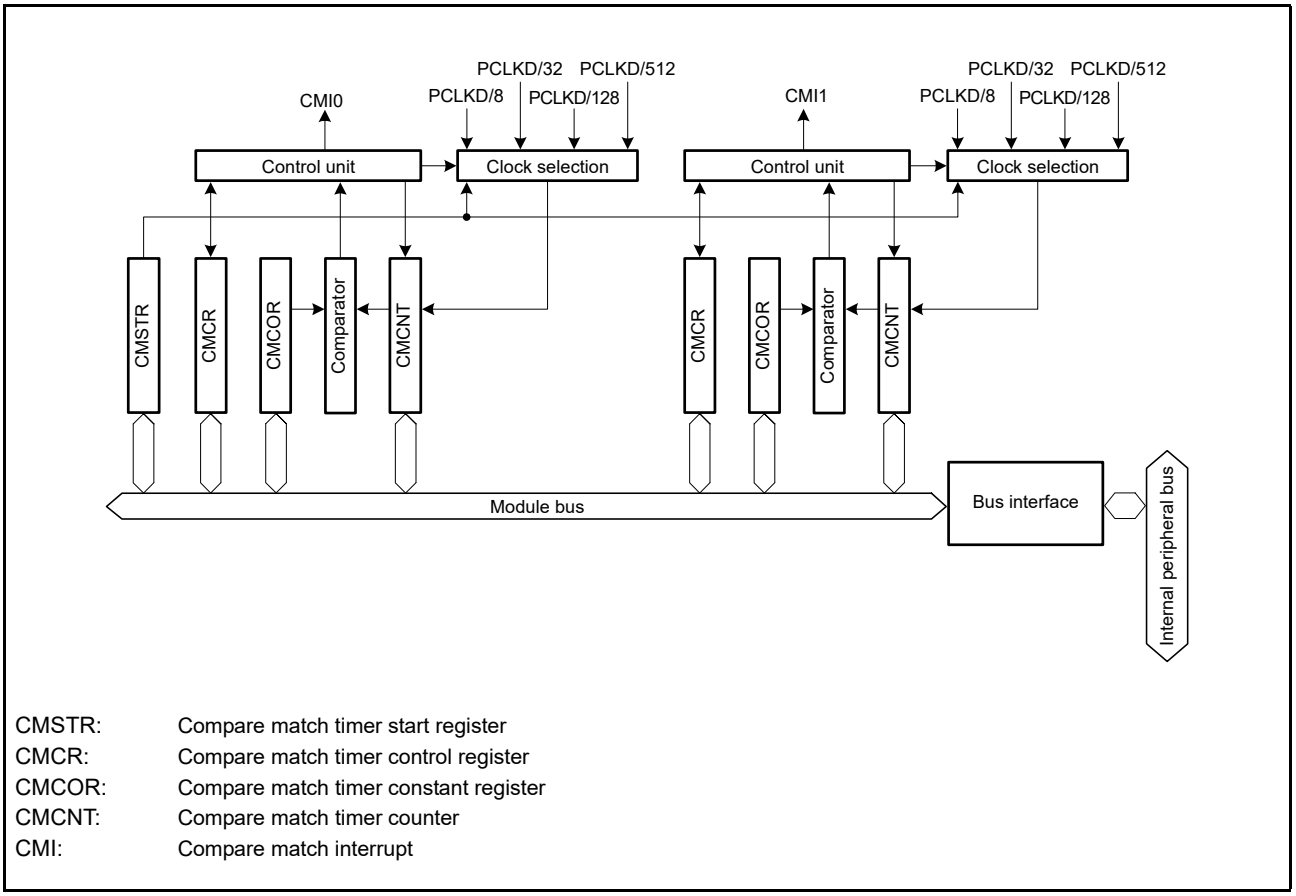


Figure 18.1 Block Diagram of CMT (Units 0 to 2)

18.2 Register Descriptions

18.2.1 Compare Match Timer Start Register 0 (CMSTR0)

The CMSTR0 register sets starting or stopping of the CMT0.CMCNT and CMT1.CMCNT counters of unit 0.

Address(es): A008 0000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR1	STR0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR0	Count Start 0	This bit selects starting or stopping of the CMT0.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT0.CMCNT counter is stopped. 1: The CMT0.CMCNT counter is started.	R/W
b1	STR1	Count Start 1	This bit selects starting or stopping of the CMT1.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT1.CMCNT counter is stopped. 1: The CMT1.CMCNT counter is started.	R/W
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

18.2.2 Compare Match Timer Start Register 1 (CMSTR1)

The CMSTR1 register sets starting or stopping of the CMT2.CMCNT and CMT3.CMCNT counters of unit 1.

Address(es): A008 0020h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR3	STR2
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR2	Counter Start 2	This bit selects starting or stopping of the CMT2.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT2.CMCNT counter is stopped. 1: The CMT2.CMCNT counter is started.	R/W
b1	STR3	Counter Start 3	This bit selects starting or stopping of the CMT3.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT3.CMCNT counter is stopped. 1: The CMT3.CMCNT counter is started.	R/W
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

18.2.3 Compare Match Timer Start Register 2 (CMSTR2)

The CMSTR2 register sets starting or stopping of the CMT4.CMCNT and CMT5.CMCNT counters of unit 2.

Address(es): A008 0040h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR5	STR4
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR4	Counter Start 4	This bit selects starting or stopping of the CMT4.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT4.CMCNT counter is stopped. 1: The CMT4.CMCNT counter is started.	R/W
b1	STR5	Counter Start 5	This bit selects starting or stopping of the CMT5.CMCNT counter operation. The corresponding prescaler starts or stops according to the setting of this bit. 0: The CMT5.CMCNT counter is stopped. 1: The CMT5.CMCNT counter is started.	R/W
b15 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

18.2.4 Compare Match Timer Control Register (CMCR)

The CMCRn register specifies a clock used for count-up operation.

Address(es): CMT0: A008 0002h, CMT1: A008 0008h, CMT2: A008 0022h, CMT3: A008 0028h, CMT4: A008 0042h, CMT5: A008 0048h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	CMIE	—	—	—	—	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	These bits select a clock to be input to the CMTn.CMCNT counter from the internal clocks obtained by dividing the frequency of the low-speed peripheral module clock (PCLKD). Setting the CMSTRm.STRn bit to 1 starts count-up operation of the corresponding CMCNT counter by using the clock selected in the CKS[1:0] bits (m = 0 to 2, n = 0 to 5). b1 b0 0 0: PCLKD/8 0 1: PCLKD/32 1 0: PCLKD/128 1 1: PCLKD/512	R/W
b5 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b6	CMIE	Compare Match Interrupt Enable	This bit selects whether to enable or disable generation of a compare match interrupt (CMIn) when the values in the CMCNT counter and in the CMCOR register match (n = 0 to 5). 0: Compare match interrupt (CMIn) disabled 1: Compare match interrupt (CMIn) enabled	R/W
b7	—	Reserved	When read, the value returned is undefined. The write value should be 0.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

18.2.5 Compare Match Timer Counter (CMCNT)

The CMCNT counter (the main unit of the compare match timer) is a readable/writable up-counter.
When an internal clock is selected by the CMCR.CKS[1:0] bits and the CMSTRm.STRn (m = 0 to 2, n = 0 to 5) bit is set to 1, the CMCNT counter starts counting up using the selected clock.
When the value in the CMCNT counter and the value in the CMCOR register match, the CMCNT counter is cleared to 0000h. At the same time, a compare match interrupt (CMIn) is generated (n = 0 to 5).

Address(es): CMT0: A008 0004h, CMT1: A008 000Ah, CMT2: A008 0024h, CMT3: A008 002Ah, CMT4: A008 0044h, CMT5: A008 004Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

18.2.6 Compare Match Timer Constant Register (CMCOR)

The CMCOR register is a readable/writable register to set a cycle for compare match with the CMCNT counter.
The cycle for compare matches is as follows.

Compare-match cycle = (setting of the CMCOR register + 1) × counter-clock cycle*1

Note 1. This is a clock cycle set by the CMCR.CKS[1:0] bits.

Address(es): CMT0: A008 0006h, CMT1: A008 000Ch, CMT2: A008 0026h, CMT3: A008 002Ch, CMT4: A008 0046h, CMT5: A008 004Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

18.3 Operation

18.3.1 Periodic Count Operation

When an internal clock is selected by the CMCRn.CKS[1:0] bits and the CMSTRm.STRn (m = 0 to 2, n = 0 to 5) bit is set to 1, the CMCNT counter starts counting up using the selected clock.
When the value in the CMCNT counter and the value in the CMCOR register match, the CMCNT counter is cleared to 0000h, and then a compare match interrupt (CMIn) is generated. The CMCNT counter then starts counting up again from 0000h. Figure 18.2 shows the operation of the CMCNT counter.

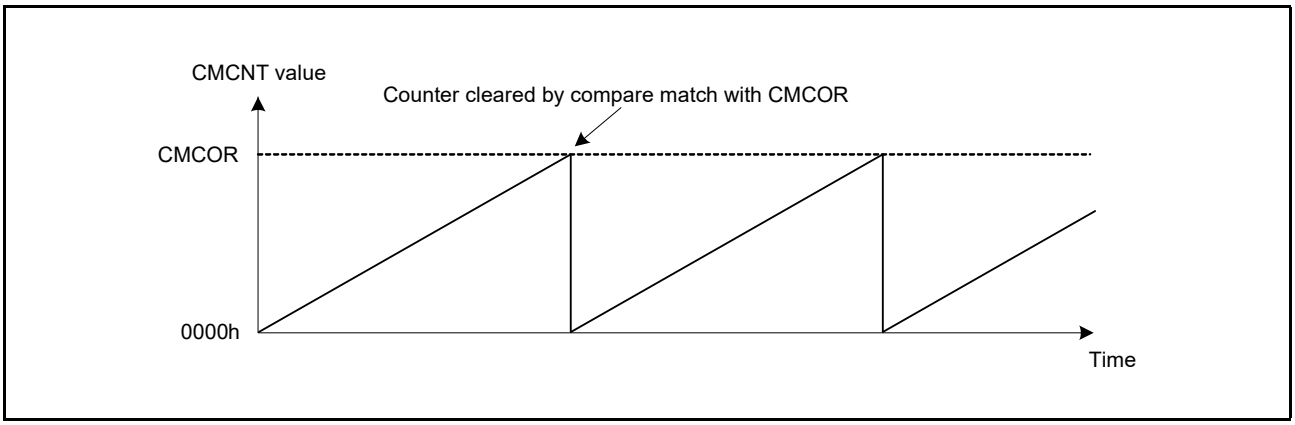


Figure 18.2 Counter Operation

18.3.2 CMCNT Count Timing

As the count clock, one of four internal clocks (PCLKD/8, PCLKD/32, PCLKD/128, and PCLKD/512) obtained by dividing the low-speed peripheral module clock (PCLKD) can be selected with the CMCR.CKS[1:0] bits. Figure 18.3 shows the timing of the CMCNT counter.

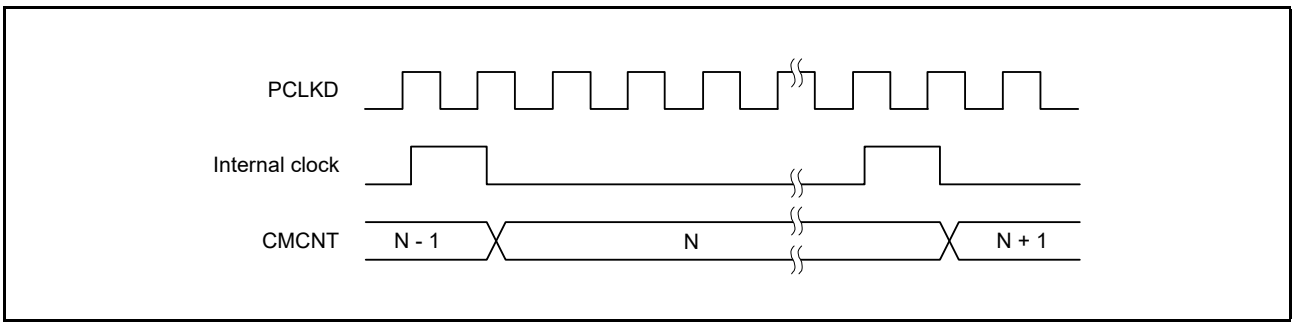


Figure 18.3 CMCNT Count Timing

18.4 Interrupts

18.4.1 Interrupt Sources

The CMT has channels and each of them to which a different vector address is allocated has a compare match interrupt (CMIn). The priority of channels can be changed by the interrupt controller settings.

Table 18.2 CMT Interrupt Sources

Name	Interrupt Sources
CMI0	Compare match between the CMT0.CMCNT counter and CMT0.CMCOR register
CMI1	Compare match between the CMT1.CMCNT counter and CMT1.CMCOR register
CMI2	Compare match between the CMT2.CMCNT counter and CMT2.CMCOR register
CMI3	Compare match between the CMT3.CMCNT counter and CMT3.CMCOR register
CMI4	Compare match between the CMT4.CMCNT counter and CMT4.CMCOR register
CMI5	Compare match between the CMT5.CMCNT counter and CMT5.CMCOR register

18.4.2 Timing of Compare Match Interrupt Generation

When the CMCNT counter and the CMCOR register match, a compare match interrupt (CMIn) is generated.
A compare match signal is generated at the last state in which the values match (the timing when the CMCNT counter updates the matched count value). That is, after a match between the CMCOR register and the CMCNT counter, the compare match signal is not generated until the CMCNT counter input clock is generated (n = 0 to 5).
Figure 18.4 shows the timing of a compare match interrupt.

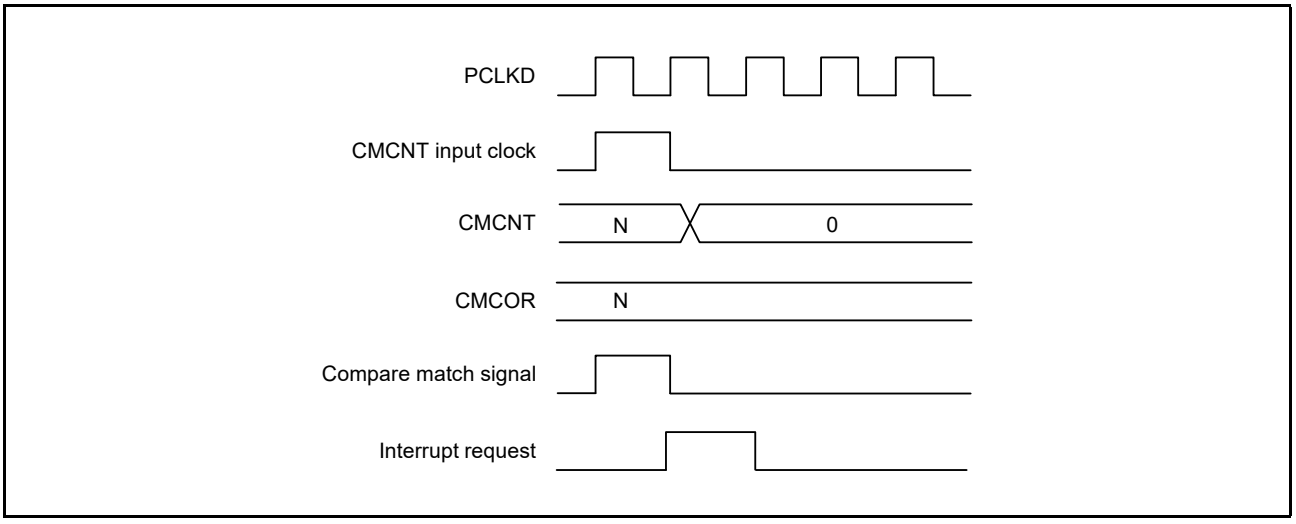


Figure 18.4 Timing a Compare Match Interrupt is Set

18.5 Event Link Operations

18.5.1 Event Issuance to ELC

The CMT issues an interrupt request when compare match occurs. Then, the CMT can use the event link controller (ELC) to perform link operation to a preset module using the interrupt request signal as the event signal.
The event signal can be output regardless of the setting of the corresponding interrupt request enable bit (CMT1.CMCR.CMIE).

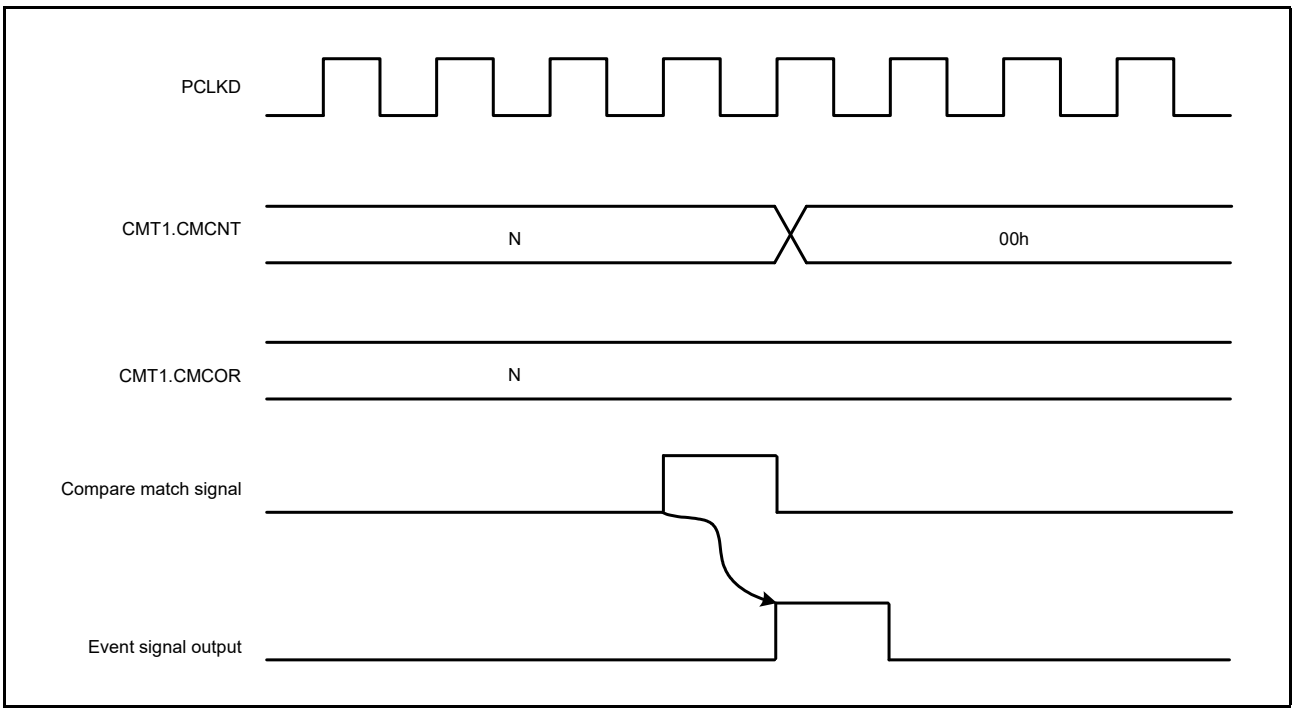


Figure 18.5 Timing of Event Issuance

18.5.2 CMT Operation When Receiving an Event from ELC

The CMT can perform either of the following three operations upon the event preset in the event link controller (ELC).

(1) Count Start

When the CMT count start operation is selected in the ELC and an event is received, the STR1 bit in the corresponding CMSTR0 (compare match timer start register 0) is set to 1, starting the count operation.
However, if the specified event occurs while the CMSTR0.STR1 bit is 1, the event is invalid.

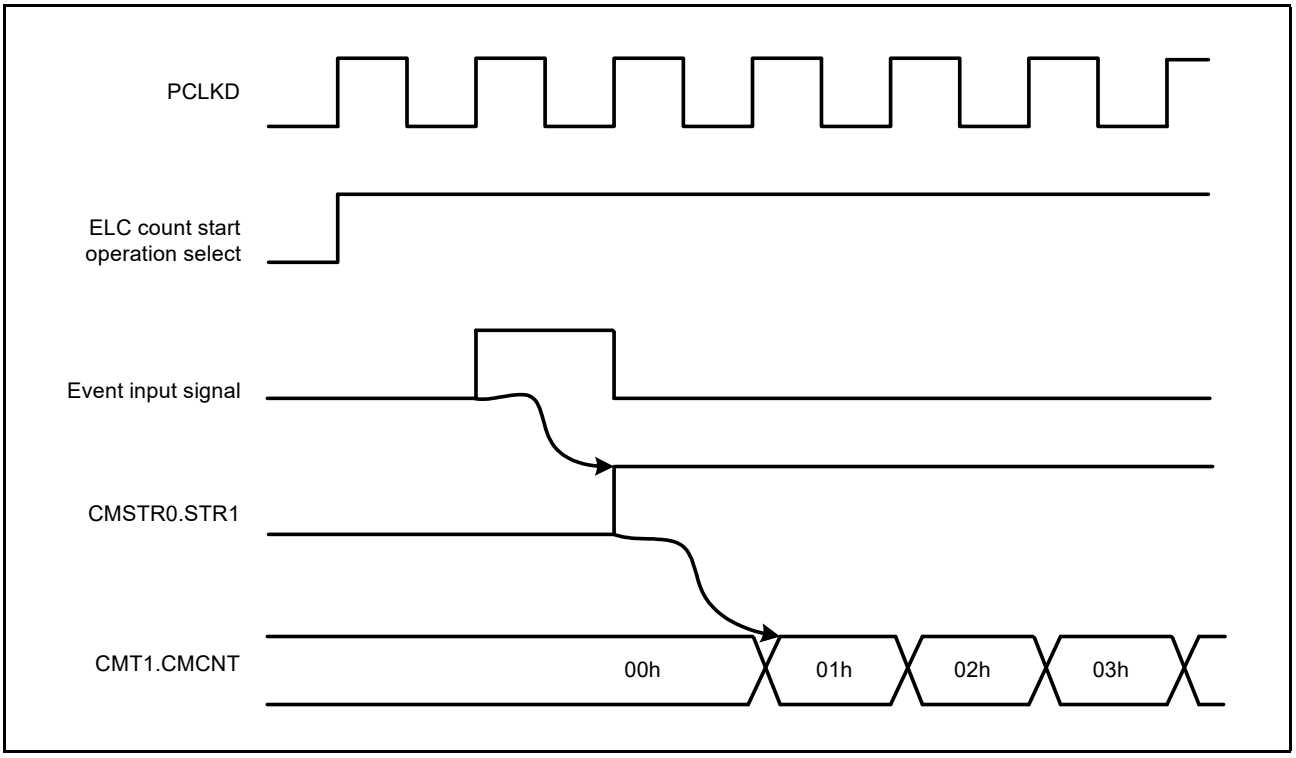


Figure 18.6 Count Start Operation at Reception of an Event

(2) Event Count

When the CMT event count operation is selected in the ELC and an event is received, the CMT1.CMCNT (compare match timer counter) is incremented, regardless of the setting of the CKS[1:0] bits in CMT1.CMCR (compare match timer control register). The STR1 bit in the CMSTR0 (compare match timer start register) must be set to 1 before receiving an event.

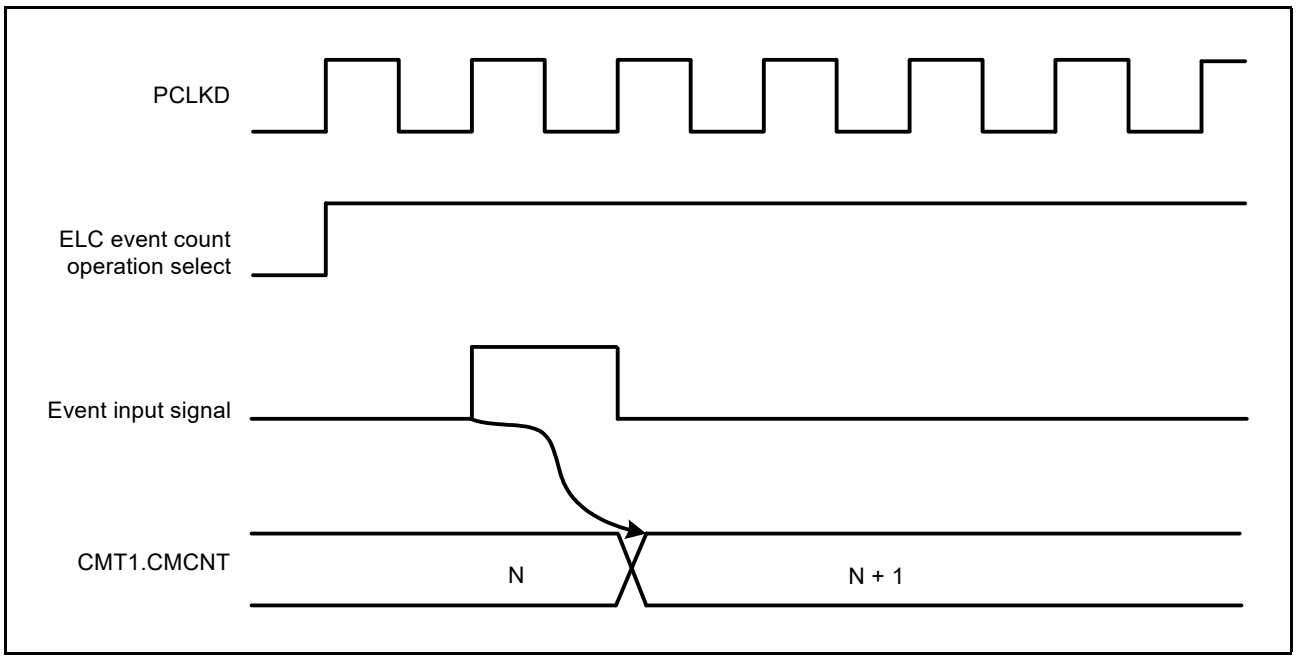


Figure 18.7 Event Count Operation at Reception of an Event

(3) Count Clear

When the CMT count clear operation is selected in the ELC and an event is received, the CMT1.CMCNT (compare match timer counter) value is modified to the initial value. If the STR1 bit in the CMSTR0 (compare match timer start register 0) is 1 here, the count operation can be continued, i.e., count operation can be restarted.

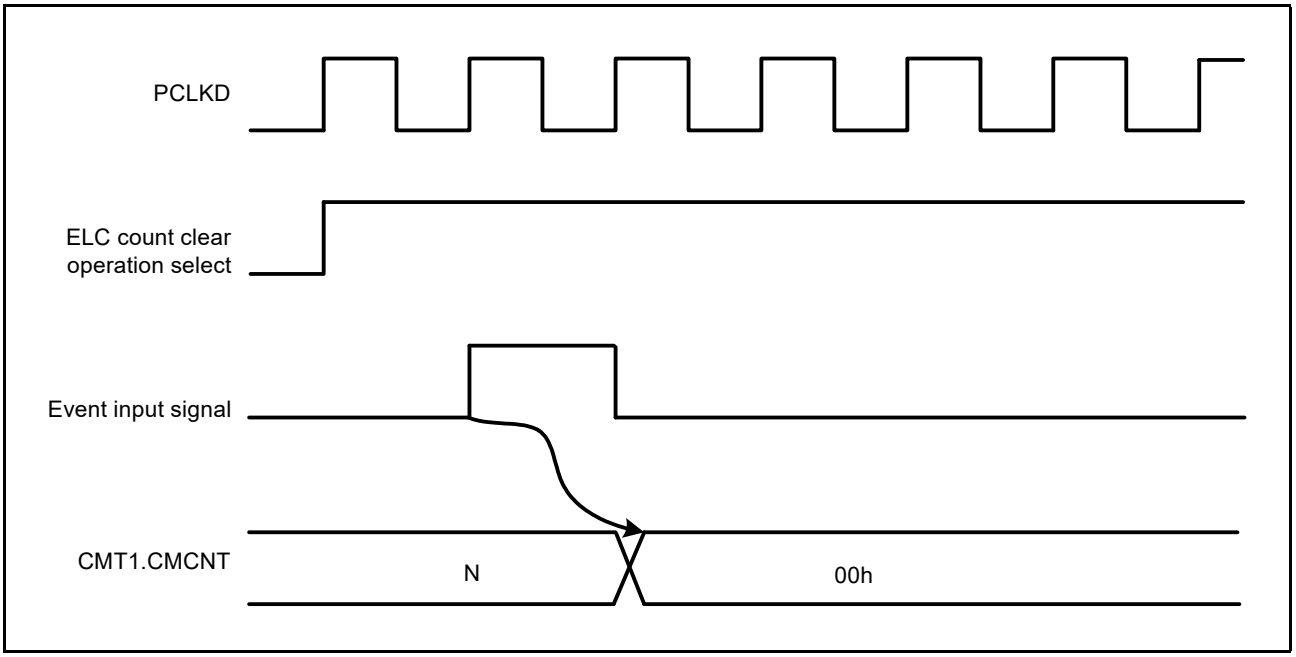


Figure 18.8 Count Clear Operation at Reception of an Event

18.5.3 Notes on CMT Event Link Operation

Note the following when using the CMT with event link operation.

(1) Count Start

When an event occurs during the write access to the STR1 bit in the CMSTR0 (compare match timer start register 0), that write access is not performed, and setting 1 according to the event occurrence takes priority.

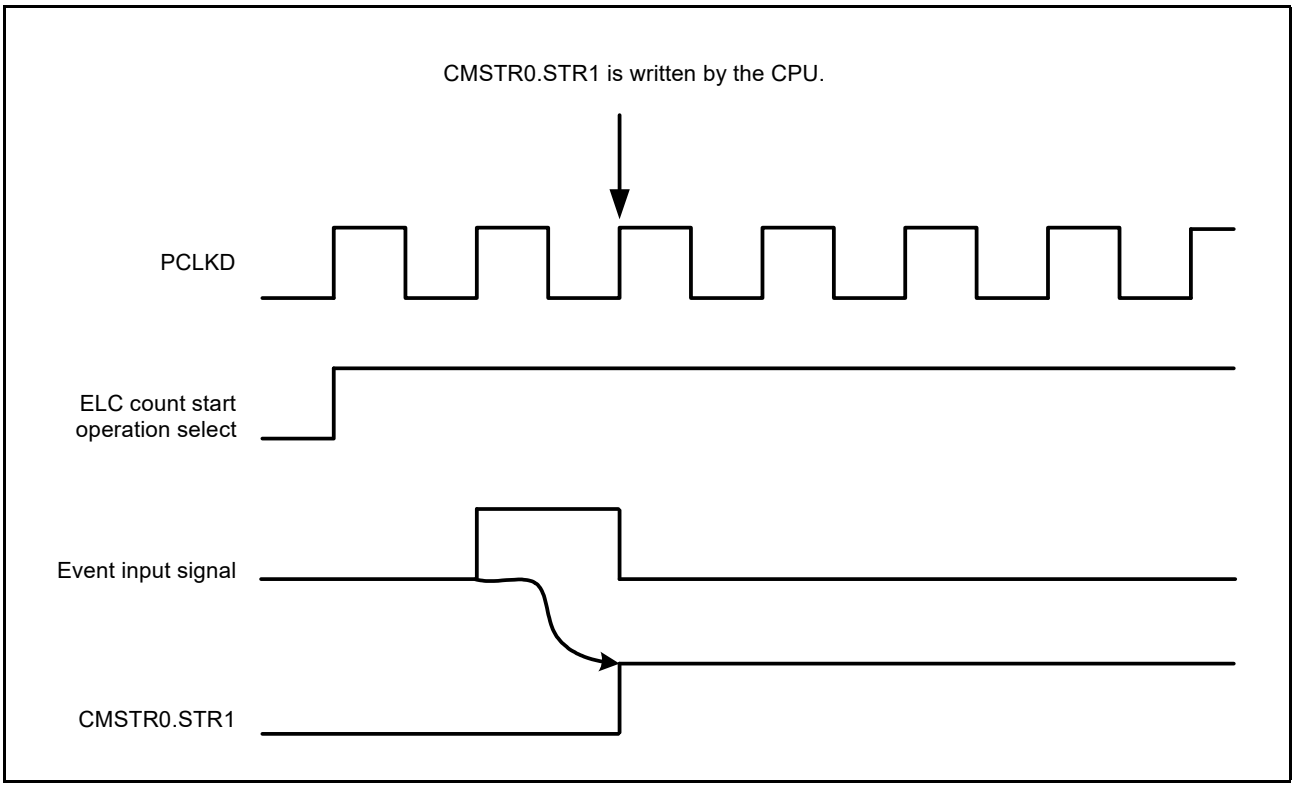


Figure 18.9 Conflict between Event Reception and Register Access at Count Start Operation

(2) Event Count

When an event occurs during the write access to the CMT1.CMCNT (compare match timer counter), that write access is not performed, and event count operation according to the event occurrence takes priority.

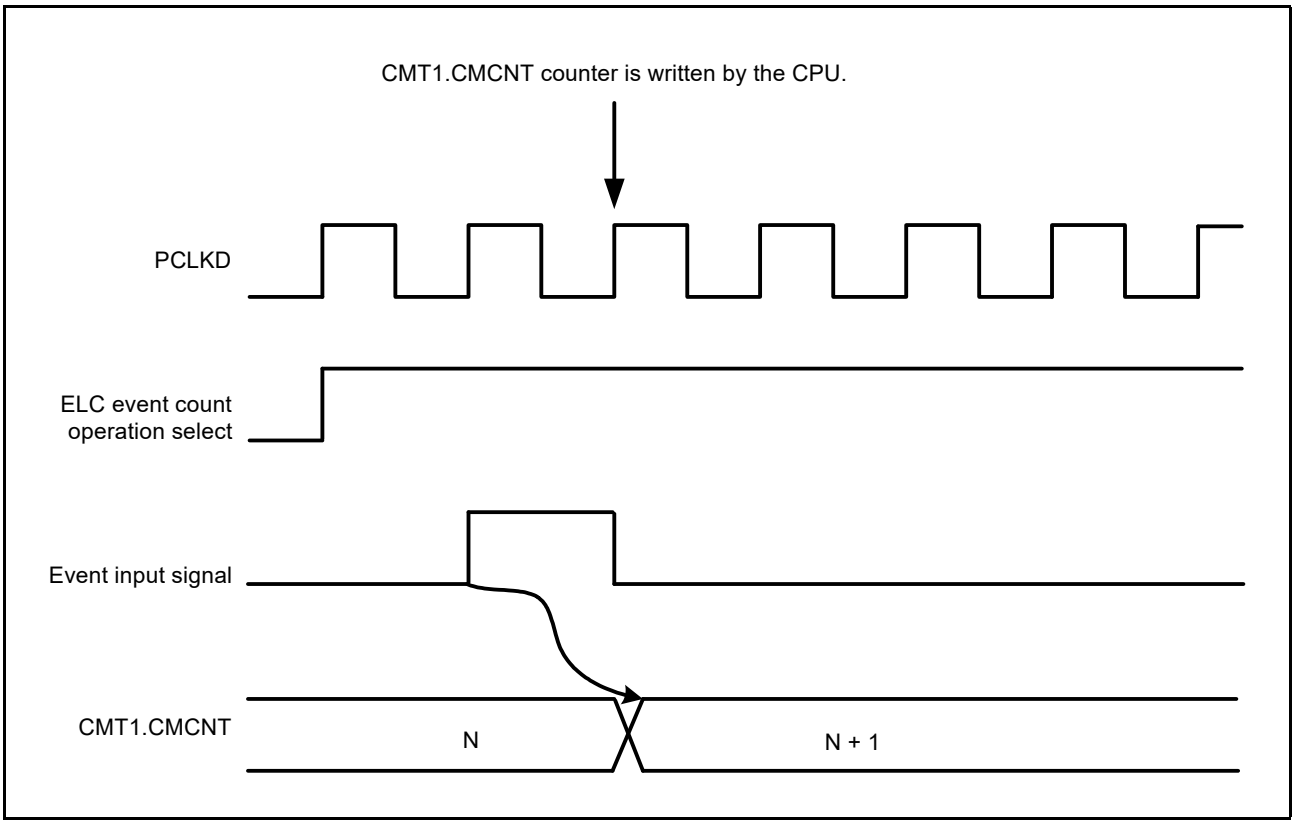


Figure 18.10 Conflict between Event Reception and Register Access at Event Count Operation

(3) Count Clear

When an event occurs during the write access to the CMT1.CMCNT (compare match timer counter), that write access is not performed, and count value initialization according to the event occurrence takes priority.

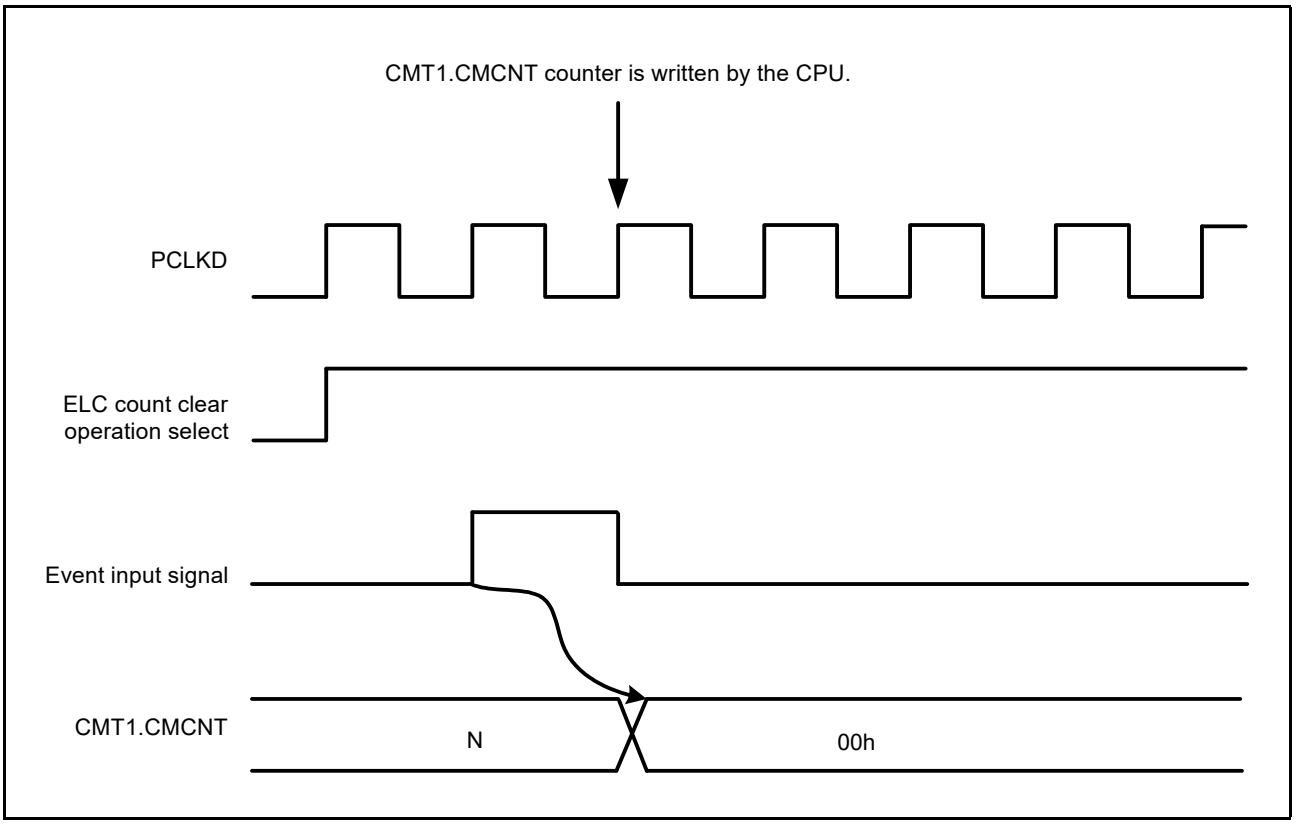


Figure 18.11 Conflict between Event Reception and Register Access at Count Clear Operation

18.6 Usage Notes

18.6.1 Setting the Module Stop Function

The CMT can be enabled or disabled using the module stop control register. After a reset, the CMT is in the module-stop state. The registers can be accessed by canceling the module-stop state. For details, see section 9, Low-Power Consumption Function.

18.6.2 Conflict between Write and Compare-Match Processes of CMCNT

When the compare match signal is generated while writing to the CMCNT counter, clearing the CMCNT counter is given priority over writing to it. In this case, the CMCNT counter is not written to. Figure 18.12 shows the timing to clear the CMCNT counter.

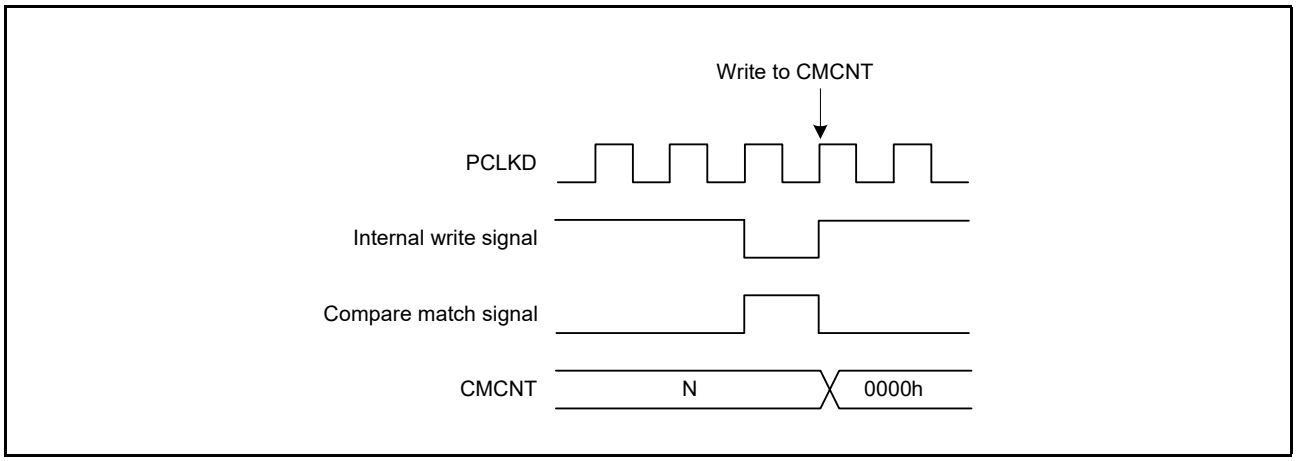


Figure 18.12 Conflict between Write and Compare Match Processes of CMCNT

18.6.3 Conflict between Write and Count-Up Processes of CMCNT

If count-up occurs during the write access to the CMCNT counter, that writing has priority over the count-up. Figure 18.13 shows the timing to write the CMCNT counter.

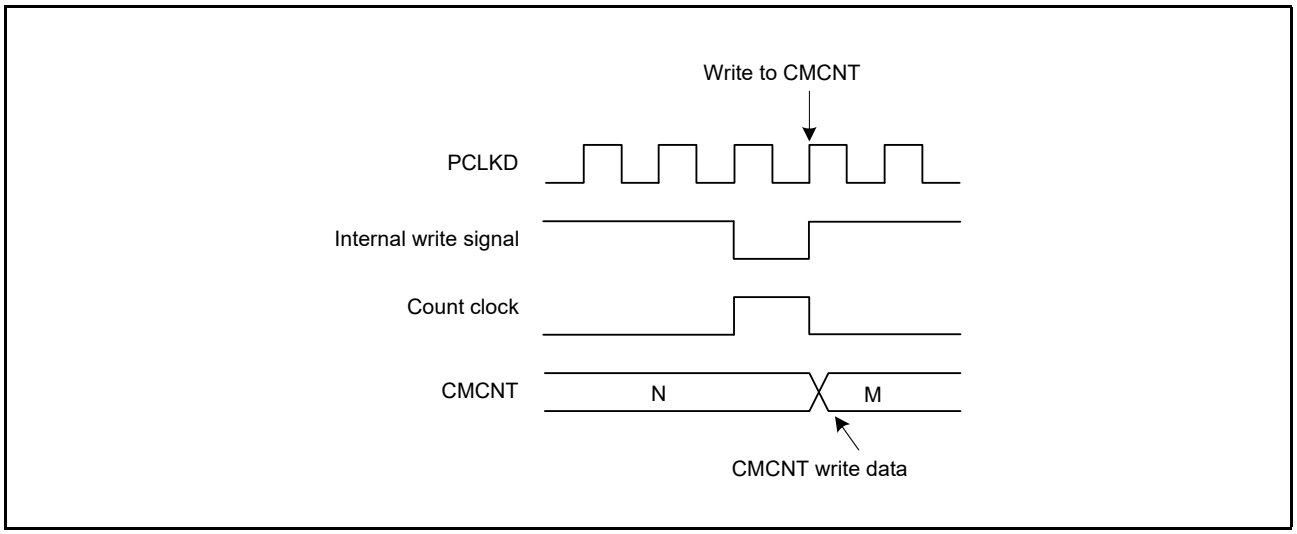


Figure 18.13 Conflict between Write and Count-Up Processes of CMCNT

Table 18.3 Summary of Conflicted Operations among Event Link Operation, Register Access, and Counter Status

Event Link Operation	Register Access	CMCNT Status	Operation to be Performed
Count start	Writing to CMSTR0.STR1	Stopped	Count start
		Compare match	Count start
		Count up	Count start
Event count	Writing to CMCNT	—	Event count
	Writing to CMCOR	Compare match	Compare match
Count clear	Writing to CMCNT	Other than compare match	Count clear
	Writing to CMCNT	Compare match	Compare match
	(No register access)	Compare match	Compare match
(No event)	Writing to CMCNT	Compare match	Compare match
		Count up	Writing to CMCNT
	Writing to CMCOR	Compare match	Compare match

19. Compare Match Timer W (CMTW)

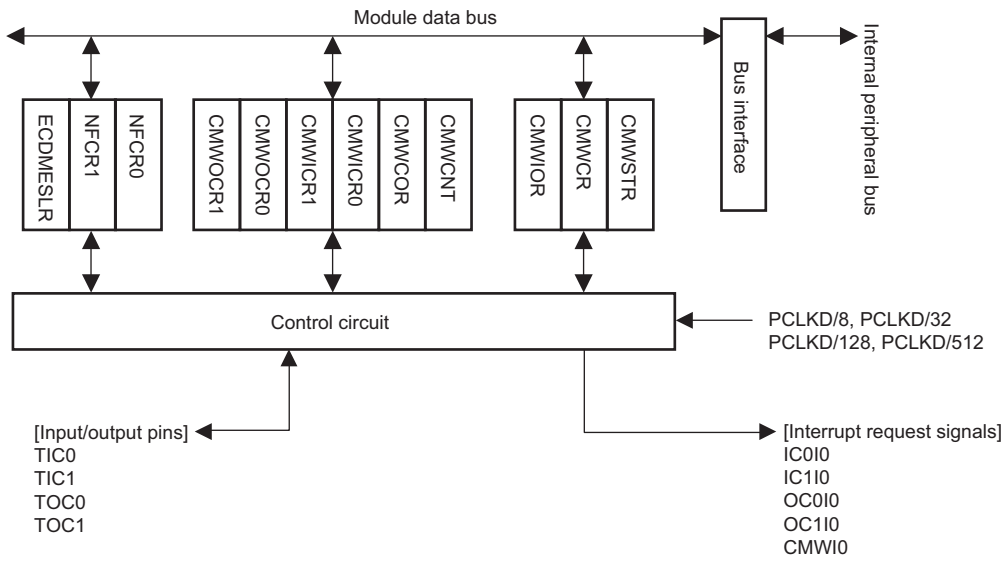
This LSI includes two units with one channel of 32-bit compare match timer W (CMTW).
CMTW has a 32-bit counter and can generate interrupts each time a set period elapses.

19.1 Overview

Table 19.1 shows the specifications of the CMTW. Figure 19.1 shows a block diagram of the CMTW.

Table 19.1 Specifications of CMTW

Item	Function
Number of channel	One channel × two units
Timer counter	16-bit/32-bit selectable up-counter Counting starts after the output of a counting enable signal from the prescaler. The counter returns to 0000 0000h after a compare match.
Prescaler	9-bit counter (operates to enable and disable the timer counter) Outputs four signals to enable counting. Selectable either of PCLKD/8, PCLKD/32, PCLKD/128, and PCLKD/512
Input capture	Up to two input capture input signals available.
Output compare	Up to two output compare output signals available.
Compare match	One compare match available.
Interrupt	• Compare match interrupt • Input capture 0 and 1 interrupts • Output compare 0 and 1 interrupts
Event link	One of the following three operations is enabled after an event signal is received: • Counting start • Event counting • Counting clear Also, the following event signal can be issued: • Compare match event
Low-power consumption function	The module-stop state can be set for each unit.



CMWSTR: Timer start register
CMWCR: Timer control register
CMWIOR: Timer I/O control register
CMWCNT: Timer counter
CMWCOR: Compare match constant register
CMWICR0: Input capture register 0
CMWICR1: Input capture register 1
CMWOCR0: Output compare register 0
CMWOCR1: Output compare register 1
NFCR0: Digital noise filter control register 0
NFCR1: Digital noise filter control register 1
ECDMESLR: ECM dynamic mode error output select register

Figure 19.1 Block Diagram of CMTW (Unit 0)

Table 19.2 shows the CMTW pin configuration.

Table 19.2 Input/Output Pins of CMTW

Unit	Pin Name	I/O	Description
CMTW0	TIC0	Input	Input capture input 0
	TIC1	Input	Input capture input 1
	TOC0	Output	Output compare output 0
	TOC1	Output	Output compare output 1
CMTW1	TIC2	Input	Input capture input 2
	TIC3	Input	Input capture input 3
	TOC2	Output	Output compare output 2
	TOC3	Output	Output compare output 3

19.2 Register Descriptions

19.2.1 Timer Start Register (CMWSTR)

The CMWSTR register is used to start or stop the CMWCNT counter.

Address(es): CMTW0.CMWSTR A008 0300h, CMTW1.CMWSTR A008 0380h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STR
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	STR	Counter Start	0: The CMWCNT counter stops counting. (The value immediately before a stop of counting is retained and counting is stopped.) 1: The CMWCNT counter starts counting.	R/W
b15 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

STR Bit (Counter Start)

Specifies whether the timer counter operates or is stopped. The relevant prescaler operates or is stopped according to the settings of STR bit.

19.2.2 Timer Control Register (CMWCR)

The CMWCR register selects the counter clearing source and the counter input clock, and enables or disables interrupts. The CMWCR register should be set while the timer counter (CMWCNT) operation is stopped.

Address(es): CMTW0.CMWCR A008 0304h, CMTW1.CMWCR A008 0384h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CCLR[2:0]			—	—	—	CMS	—	OC1IE	OC0IE	IC1IE	IC0IE	CMWIE	—	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: PCLKD/8 0 1: PCLKD/32 1 0: PCLKD/128 1 1: PCLKD/512	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	CMWIE	Compare Match Interrupt Enable	0: Disables a compare match interrupt request (CMWI). 1: Enables a compare match interrupt request (CMWI).	R/W
b4	IC0IE	Input Capture 0 Interrupt Enable	0: Disables an interrupt request by the input capture 0 bit (IC0I). 1: Enables an interrupt request by the input capture 0 bit (IC0I).	R/W
b5	IC1IE	Input Capture 1 Interrupt Enable	0: Disables an interrupt request by the input capture 1 bit (IC1I). 1: Enables an interrupt request by the input capture 1 bit (IC1I).	R/W
b6	OC0IE	Output Compare 0 Interrupt Enable	0: Disables an interrupt request by the output compare 0 bit (OC0I). 1: Enables an interrupt request by the output compare 0 bit (OC0I).	R/W
b7	OC1IE	Output Compare 1 Interrupt Enable	0: Disables an interrupt request by the output compare 1 bit (OC1I). 1: Enables an interrupt request by the output compare 1 bit (OC1I).	R/W
b8	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b9	CMS	Timer Counter Size	0: 32 bits 1: 16 bits	R/W
b12 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15 to b13	CCLR[2:0]	Counter Clear	b15 b13 0 0 0: The CMWCNT counter is cleared by CMWCOR register compare match. 0 0 1: The CMWCNT counter is not cleared. 0 1 0: The CMWCNT counter is not cleared. 0 1 1: The CMWCNT counter is not cleared. 1 0 0: The CMWCNT counter is cleared by CMWICR0 register input capture. 1 0 1: The CMWCNT counter is cleared by CMWICR1 register input capture. 1 1 0: The CMWCNT counter is cleared by CMWOCR0 register compare match. 1 1 1: The CMWCNT counter is cleared by CMWOCR1 register compare match.	R/W

CKS[1:0] Bits (Clock Select)

Select the clock to be input to the CMWCNT counter among four internal clocks obtained by dividing the peripheral clock (PCLKD). When the CMWSTR.STR bit is set to 1, the CMWCNT counter starts counting up based on the clock selected with the CMWCNT.CKS[1:0] bits.

CMWIE Bit (Compare Match Interrupt Enable)

Enables or disables compare match interrupt (CMWI) request generation when the CMWCNT counter and CMWCOR register values match.

IC0IE Bit (Input Capture 0 Interrupt Enable)

Enables or disables input capture interrupt 0 (IC0I) request generation when input capture is generated in the CMWICR0 register.

IC1IE Bit (Input Capture 1 Interrupt Enable)

Enables or disables input capture interrupt 1 (IC1I) request generation when input capture is generated in the CMWICR1 register.

OC0IE Bit (Output Compare 0 Interrupt Enable)

Enables or disables compare match interrupt 0 (OC0I) request generation when the CMWCNT counter and CMWOCR0 register values match.

OC1IE Bit (Output Compare 1 Interrupt Enable)

Enables or disables compare match interrupt 1 (OC1I) request generation when the CMWCNT counter and CMWOCR1 register values match.

CMS Bit (Timer Counter Size)

Selects either 16 or 32 bits as the size of the timer counter (CMWCNT). The size selected with the CMS bit is valid in the compare match constant register (CMWCOR), input capture registers (CMWICR0 and CMWICR1), and output compare registers (CMWOCR0 and CMWOCR1).

CCLR[2:0] Bits (Counter Clear)

Selects the CMWCNT counter clearing source.

19.2.3 Timer I/O Control Register (CMWIOR)

The CMWIOR register controls the CMWCOR, CMWICR0, CMWICR1, CMWOCR0, and CMWOCR1 registers. CMWIOR should be set while the timer counter (CMWCNT) operation is stopped.

Address(es): CMTW0.CMWIOR A008 0308h, CMTW1.CMWIOR A008 0388h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CMWE	—	OC1E	OC0E	OC1[1:0]		OC0[1:0]		—	—	IC1E	IC0E	IC1[1:0]		IC0[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	IC0[1:0]	Input Compare Control 0	b1 b0 0 0: Input capture at the rising edge on the TIC0 pin. 0 1: Input capture at the falling edge on the TIC0 pin. 1 0: Input capture at both edges on the TIC0 pin. 1 1: Setting prohibited	R/W
b3, b2	IC1[1:0]	Input Capture Control 1	b3 b2 0 0: Input capture at the rising edge on the TIC1 pin. 0 1: Input capture at the falling edge on the TIC1 pin. 1 0: Input capture at both edges on the TIC1 pin. 1 1: Setting prohibited	R/W
b4	IC0E	Input Capture Enable 0	0: Disables the input capture operation of the CMWICR0 register. 1: Enables the input capture operation of the CMWICR0 register.	R/W
b5	IC1E	Input Capture Enable 1	0: Disables the input capture operation of the CMWICR1 register. 1: Enables the input capture operation of the CMWICR1 register.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b9, b8	OC0[1:0]	Output Compare Control 0	b9 b8 0 0: Retains the output value.*1 0 1: Initially outputs 0 and toggles the output value upon compare match. 1 0: Initially outputs 1 and toggles the output value upon compare match. 1 1: Setting prohibited	R/W
b11, b10	OC1[1:0]	Output Compare Control 1	b11 b10 0 0: Retains the output value.*1 0 1: Initially outputs 0 and toggles the output value upon compare match. 1 0: Initially outputs 1 and toggles the output value upon compare match. 1 1: Setting prohibited	R/W
b12	OC0E	Compare Match Enable 0	0: Disables the compare match operation using the CMWOCR0 register. 1: Enables the compare match operation using the CMWOCR0 register.	R/W
b13	OC1E	Compare Match Enable 1	0: Disables the compare match operation using the CMWOCR1 register. 1: Enables the compare match operation using the CMWOCR1 register.	R/W
b14	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b15	CMWE	Compare Match Enable	0: Disables the compare match operation using the CMWCOR register. 1: Enables the compare match operation using the CMWCOR register.	R/W

Note 1. After reset, 0 is output until the CMWIOR register is set.

IC0[1:0] Bits (Input Compare Control 0)

Selects the input capture operation of the CMWICR0 register.

IC1[1:0] Bits (Input Capture Control 1)

Selects the input capture operation of the CMWICR1 register.

IC0E Bit (Input Capture Enable 0)

Enables or disables the input capture operation of the CMWICR0 register.

IC1E Bit (Input Capture Enable 1)

Enables or disables the input capture operation of the CMWICR1 register.

OC0[1:0] Bits (Output Compare Control 0)

Sets the output compare operation using the CMWOCR0 register.

OC1[1:0] Bits (Output Compare Control 1)

Sets the output compare operation using the CMWOCR1 register.

OC0E Bit (Compare Match Enable 0)

Enables or disables the compare match operation using the CMWOCR0 register.

OC1E Bit (Compare Match Enable 1)

Enables or disables the compare match operation using the CMWOCR1 register.

CMWE Bit (Compare Match Enable)

Enables or disables the compare match operation using the CMWCOR register.

19.2.4 Timer Counter (CMWCNT)

The CMWCNT counter is used as a readable/writable up-counter.

Before starting counter operation, the timer control register (CMWCR) should be set. When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in the CMWCNT counter are valid. Since access to this register is in 32-bit units, when writing, write 32-bit values to it with 0000h in the higher-order bits. If a value other than 0000h is set in the higher-order bits, a value greater than 0000 FFFFh may be read when this register is read.

When the STR bit is set to 1, the CMWCNT counter starts counting. When the STR bit is set to 0, the CMWCNT counter retains the value immediately before a stop of counting and stops counting.

Address(es): CMTW0.CMWCNT A008 0310h, CMTW1.CMWCNT A008 0390h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

19.2.5 Compare Match Constant Register (CMWCOR)

The CMWCOR register is a readable/writable register that specifies the time up to a compare match between the timer counter (CMWCNT) value and CMWCOR value. When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in this register are valid. Since access to this register is in 32-bit units, when writing, write 32-bit values to it with 0000h in the higher-order bits.
The cycle for compare matches is as follows.

Compare-match cycle = (setting of the CMWCOR register + 1) × counter-clock cycle*1

Note 1. This is a clock cycle set by the CMWCR.CKS[1:0] bits.

Address(es): CMTW0.CMWCOR A008 0314h, CMTW1.CMWCOR A008 0394h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

19.2.6 Input Capture Registers 0 and 1 (CMWICR0 and CMWICR1)

The CMWICR0 and CMWICR1 registers are read-only registers in which the CMWCNT counter value is stored when an input capture is generated.
When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 in these registers are valid. Writing to these registers is invalid.

Address(es): CMTW0.CMWICR0 A008 0318h, CMTW0.CMWICR1 A008 031Ch,
CMTW1.CMWICR0 A008 0398h, CMTW1.CMWICR1 A008 039Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

19.2.7 Output Compare Registers 0 and 1 (CMWOCR0 and CMWOCR1)

The CMWOCR0 and CMWOCR1 registers are readable/writable registers that set the value to be compared when an output compare is generated.

When the 16-bit counter size is selected with the CMS bit in the timer control register (CMWCR), bits 15 to 0 of these registers become valid. Since access to this register is in 32-bit units, when writing, write 32-bit values to it with 0000h in the higher-order bits.

Address(es): CMTW0.CMWOCR0 A008 0320h, CMTW0.CMWOCR1 A008 0324h, CMTW1.CMWOCR0 A008 03A0h, CMTW1.CMWOCR1 A008 03A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

19.2.8 Digital Noise Filter Control Register 0 (NFCR0)

The NFCR0 register controls digital noise filters for input capture signals (TICn, n = 0, 1) of CMTW0.
The NFCR0 register should be set while the CMTW0.CMWCNT counter operation is stopped.

Address(es): CMTW.NFCR0 A008 0400h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	NFCS0[1:0]		NF1EN	NF0EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NF0EN	Digital Noise Filter Enable 0	0: The digital noise filter for the TIC0 pin is disabled. 1: The digital noise filter for the TIC0 pin is enabled.	R/W
b1	NF1EN	Digital Noise Filter Enable 1	0: The digital noise filter for the TIC1 pin is disabled. 1: The digital noise filter for the TIC1 pin is enabled.	R/W
b3, b2	NFCS0 [1:0]	Digital Noise Filter Clock Select 0	b3 b2 0 0: PCLKD/1 0 1: PCLKD/8 1 0: PCLKD/32 1 1: PCLKD/64	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NF0EN Bit (Digital Noise Filter Enable 0)

This bit enables or disables the noise filter for the TIC0 pin. Before changing the value of this bit, select the output compare function for the relevant pin in the timer I/O control register. Changing the value of the bit when the output compare function is not selected may lead to the internal generation of an unexpected edge.

NF1EN Bit (Digital Noise Filter Enable 1)

This bit enables or disables the noise filter for the TIC1 pin. Before changing the value of this bit, select the output compare function for the relevant pin in the timer I/O control register. Changing the value of the bit while the output compare function is not selected may lead to the internal generation of an unexpected edge.

NFCS0[1:0] Bits (Digital Noise Filter Clock Select 0)

These bits select the sampling clock for the digital noise filter of CMTW0.
The input-capture signal is sampled on rising edges of the selected clock signal. If the sampled levels match three times in a row, the given level is regarded as the input-capture signal. If the levels do not match, the existing value is retained.
After setting these bits, wait for two selected sampling periods before selecting the input capture function.

19.2.9 Digital Noise Filter Control Register 1 (NFCR1)

The NFCR1 register controls digital noise filters for input capture signals (TICn, n = 2, 3) of CMTW1.
The NFCR1 register should be set while the CMTW1.CMWCNT counter operation is stopped.

Address(es): CMTW.NFCR1 A008 0404h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	NFCS1[1:0]		NF3EN	NF2EN
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	NF2EN	Digital Noise Filter Enable 2	0: The digital noise filter for the TIC2 pin is disabled. 1: The digital noise filter for the TIC2 pin is enabled.	R/W
b1	NF3EN	Digital Noise Filter Enable 3	0: The digital noise filter for the TIC3 pin is disabled. 1: The digital noise filter for the TIC3 pin is enabled.	R/W
b3, b2	NFCS1 [1:0]	Digital Noise Filter Clock Select 1	b3 b2 0 0: PCLKD/1 0 1: PCLKD/8 1 0: PCLKD/32 1 1: PCLKD/64	R/W
b31 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NF2EN Bit (Digital Noise Filter Enable 2)

This bit enables or disables the noise filter for the TIC2 pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register before doing so.

NF3EN Bit (Digital Noise Filter Enable 3)

This bit enables or disables the noise filter for the TIC3 pin. Since changing the value of the bit may lead to the internal generation of an unexpected edge, select the output compare function for the relevant pin in the timer I/O control register before doing so.

NFCS1[1:0] Bits (Digital Noise Filter Clock Select 1)

These bits select the sampling clock for the digital noise filter of CMTW1.
The input-capture signal is sampled on rising edges of the selected clock signal. If the sampled levels match three times in a row, the given level is regarded as the input-capture signal. If the levels do not match, the existing value is retained.
After setting these bits, wait for two selected sampling periods before selecting the input capture function.

19.2.10 ECM Dynamic Mode Error Output Select Register (ECDMESLR)

The ECDMESLR register is used to select the error output signal in error control module (ECM) dynamic mode.

Address(es): CMTW.ECDMESLR A008 0410h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DMERSL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	DMERSL [2:0]	ECM Dynamic Mode Error Output Select	b2 b1 b0 0 0 0: Fixed to 0 (an output compare signal not used) 0 0 1: CMTW0.CMWCOR0 output compare signal 0 1 0: CMTW0.CMWCOR1 output compare signal 0 1 1: CMTW1.CMWCOR0 output compare signal 1 0 0: CMTW1.CMWCOR1 output compare signal Settings other than above are prohibited.	R/W
b31 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

DMERSL[2:0] Bit (ECM Dynamic Mode Error Output Select)

These bits are used to select the error output signal in ECM dynamic mode from among the CMTW output compare signals.

For details of ECM dynamic mode, see section 32, Error Control Module (ECM).

19.3 Operation

When the CMWCR register is set and then the STR bit in CMWSTR is set to 1, the CMTW starts counter operation. Setting the CMWSTR.STR bit to 0 enables the CMWCNT counter to retain the value immediately before a stop of counting and stop counting. Setting CMWIOR register enables using the compare match function, input capture input function, and output compare output function.

19.3.1 Period Counting Operation

When the internal clock is selected by using the CMWCNT.CKS[1:0] bits and the CMWSTR.STR bit is set to 1, the CMWCNT counter starts counting up cycles of the selected clock. When the CCLR[2:0] bits in CMWCR are set so that CMWCNT should be cleared by a specific counter clearing source and the counter clearing source is generated, the CMWCNT counter is cleared to 0000 0000h and continues incrementing. When the CCLR[2:0] bits are set so that CMWCNT should not be cleared by any specific counter clearing source, the CMWCNT counter is cleared to 0000 0000h only when an overflow is generated (FFFF FFFFh → 0000 0000h (when the counter size is 32 bits) or 0000 FFFFh → 0000 0000h (when the counter size is 16 bits)) and continues incrementing.

19.3.2 Compare Match Function

When the values of the CMWCNT counter and CMWCOR register match, a compare match interrupt (CMWI) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[1:0].

1. When CMWCR.CCLR[2:0] = 000b
   When the values of the CMWCNT counter and CMWCOR register match, the CMWCNT counter is cleared to 0000 0000h. The CMWCNT counter then restarts counting up from 0000 0000h.
2. When CMWCR.CCLR[2:0] ≠ 000b
   Even when the values of the CMWCNT counter and CMWCOR register match, the CMWCNT counter is not cleared to 0000 0000h but continues counting up until the clearing condition set in CMWCR.CCLR[1:0] is satisfied or the value of the counter reaches FFFF FFFFh (when the size of the counter is 32 bits) or ****FFFFh (when the size of the counter is 16 bits).
   The CMWCNT counter is then cleared to 0000 0000h and restarts counting up from 0000 0000h.

Figure 19.2 shows an example of procedure for setting compare match operation.

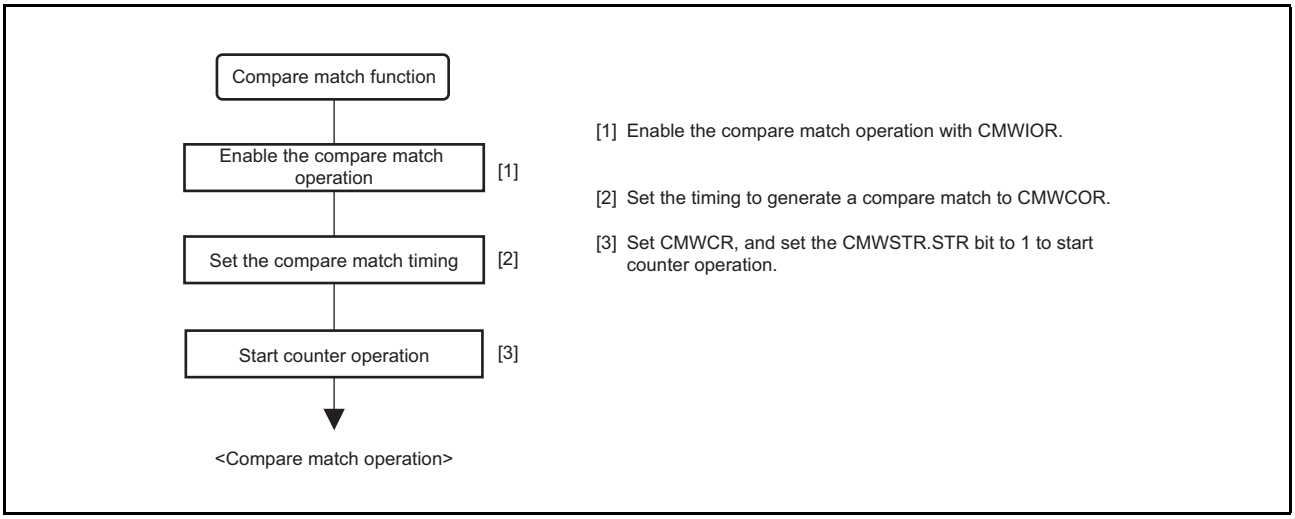


Figure 19.2 Procedure for Setting Compare Match Operation

Figure 19.3 shows an example when compare match with CMWCOR is set as a counter clearing source.

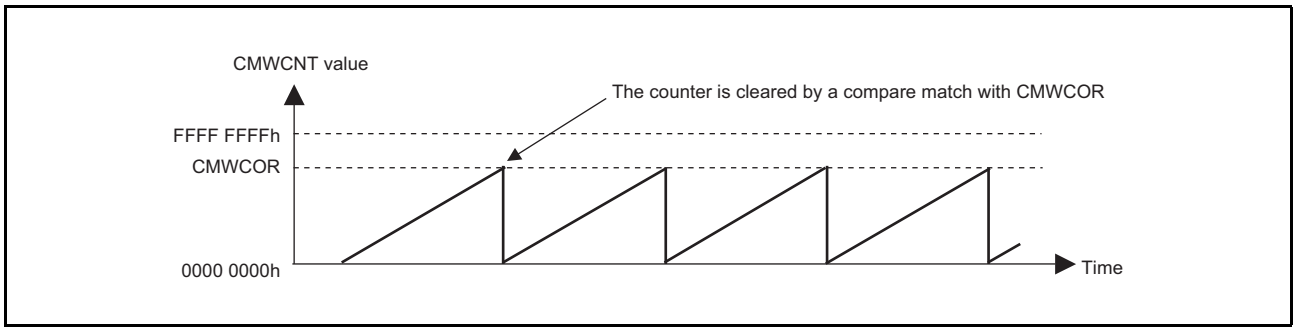


Figure 19.3 Example of Compare Match Operation

Figure 19.4 shows an example when CMWCOR is set to FFFF FFFFh and an overflow is detected.

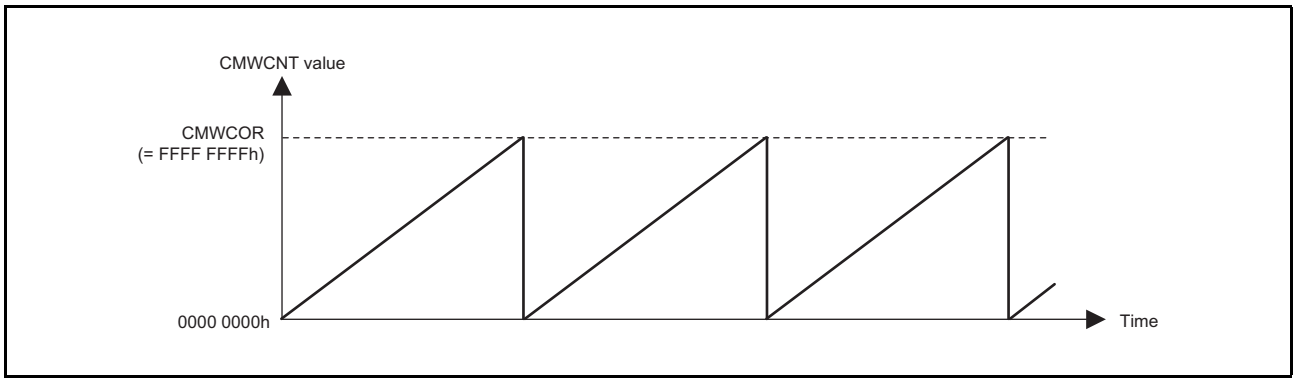


Figure 19.4 Example of Compare Match Operation (Overflow Detected)

19.3.3 Output Compare Function

Using the output compare function, toggle output from the relevant output pins can be provided. When the CMWCNT counter value matches either of the values of CMWOCR0 or CMWOCR1 register, the output compare interrupt (OC0I or OC1I) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[2:0].

1. When CMWCR.CCLR[2:0] = 110b
   When the values of the CMWCNT counter and CMWOCR0 register match, the CMWCNT counter is cleared to 0000 0000h. The CMWCNT counter then restarts counting up from 0000 0000h.
2. When CMWCR.CCLR[2:0] = 111b
   When the values of the CMWCNT counter and CMWOCR1 register match, the CMWCNT counter is cleared to 0000 0000h. The CMWCNT counter then restarts counting up from 0000 0000h.
3. When CMWCR.CCLR[2:0] ≠ 110b or 111b
   Even when the values of the CMWCNT counter and CMWOCR0 or CMWOCR1 register match, the CMWCNT counter is not cleared to 0000 0000h but continues counting up until the clearing condition set in CMWCR.CCLR[2:0] is satisfied or the value of the counter reaches FFFF FFFFh (when the size of the counter is 32 bits) or ****FFFFh (when the size of the counter is 16 bits). The CMWCNT counter is then cleared to 0000 0000h and restarts counting up from 0000 0000h.

Figure 19.5 shows an example of procedure for setting output compare operation.

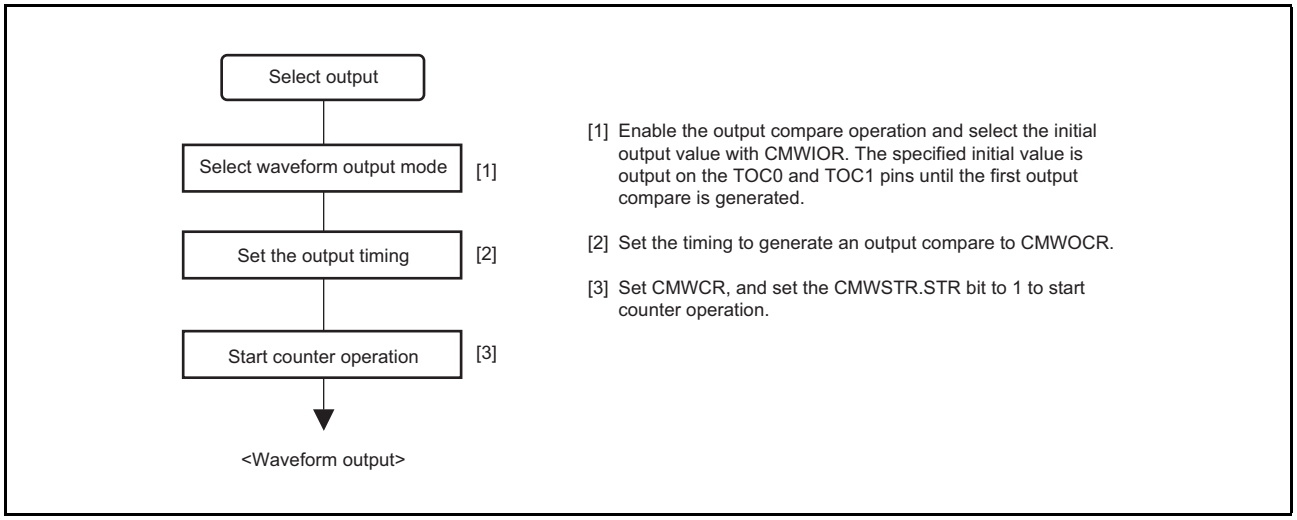


Figure 19.5 Procedure for Setting Output Compare Operation

Figure 19.6 shows an example when the counter is cleared upon compare match with CMWOCR1 register and toggle outputs are provided from the TOC0 and TOC1 pins.

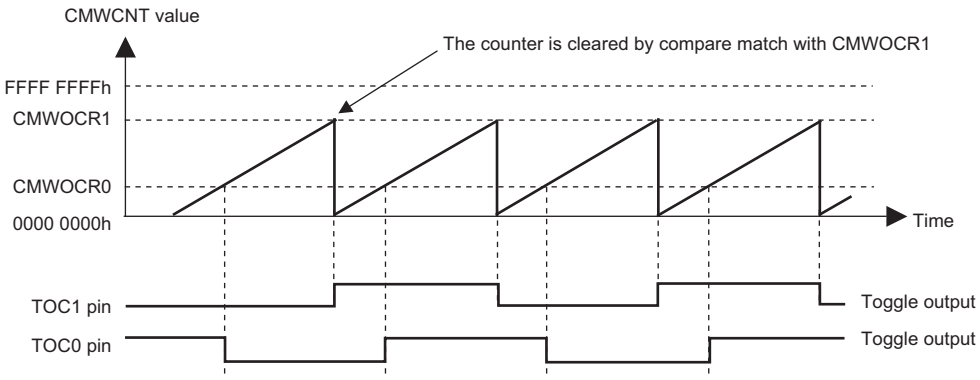


Figure 19.6 Example of Output Compare Operation

19.3.4 Input Capture Function

Through detecting the edge on the TIC0 and TIC1 pin input, the CMWCNT counter value can be transferred to CMWICR0 and CMWICR1 registers, respectively. The edges to be detected can be selected from among the rising edge alone, falling edge alone, and both the rising and falling edges. When the CMWCNT counter value is transferred to CMWICR0 or CMWICR1 register using the input capture function, an input compare interrupt (IC0I or IC1I) is generated. The CMWCNT counter operates as follows according to the setting of CMWCR.CCLR[2:0].

1. When CMWCR.CCLR[2:0] = 100b
   When the CMWCNT counter value is transferred to CMWICR0 using the input capture operation, the CMWCNT counter is cleared to 0000 0000h.
   The CMWCNT counter then restarts counting up from 0000 0000h.
2. When CMWCR.CCLR[2:0] = 101b
   When the CMWCNT counter value is transferred to CMWICR1 using the input capture operation, the CMWCNT counter is cleared to 0000 0000h.
   The CMWCNT counter then restarts counting up from 0000 0000h.
3. When CMWCR.CCLR[2:0] ≠ 100b or 101b
   Even when the CMWCNT counter value is transferred to CMWICR0 or CMWICR1 using the input capture operation, the CMWCNT counter is not cleared to 0000 0000h but continues counting up until the clearing condition set in CMWCR.CCLR[2:0] is satisfied or the value of the counter reaches FFFF FFFFh (when the size of the counter is 32 bits) or ****FFFFh (when the size of the counter is 16 bits). The CMWCNT counter is then cleared to 0000 0000h and restarts counting up from 0000 0000h.

Figure 19.7 shows an example of procedure for setting input capture operation.

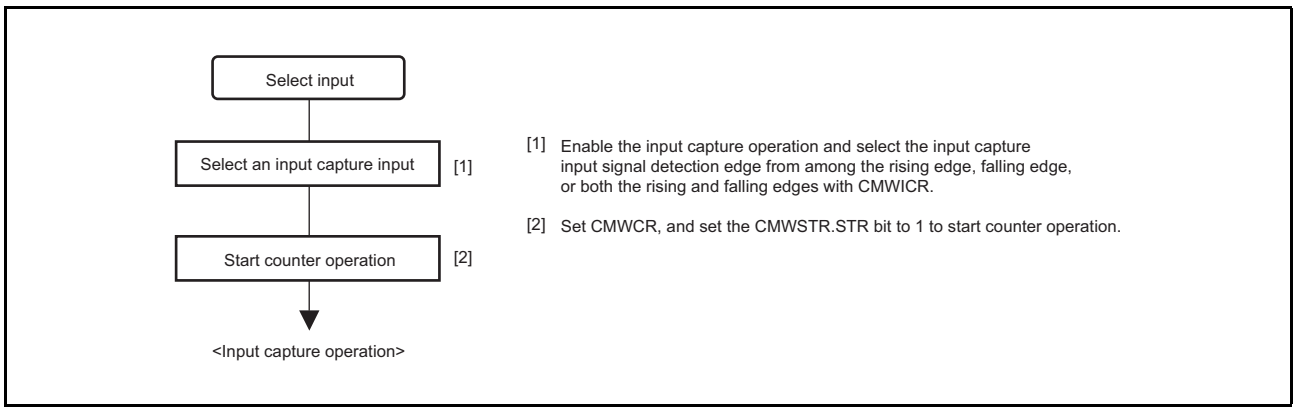


Figure 19.7 Procedure for Setting Input Capture Operation

Figure 19.8 shows an example in which both the rising and falling edges are selected for the TIC0 pin input capture input edge and the falling edge for the TIC1 pin, and the CMWCNT counter is cleared by a CMWICR1 register input capture.

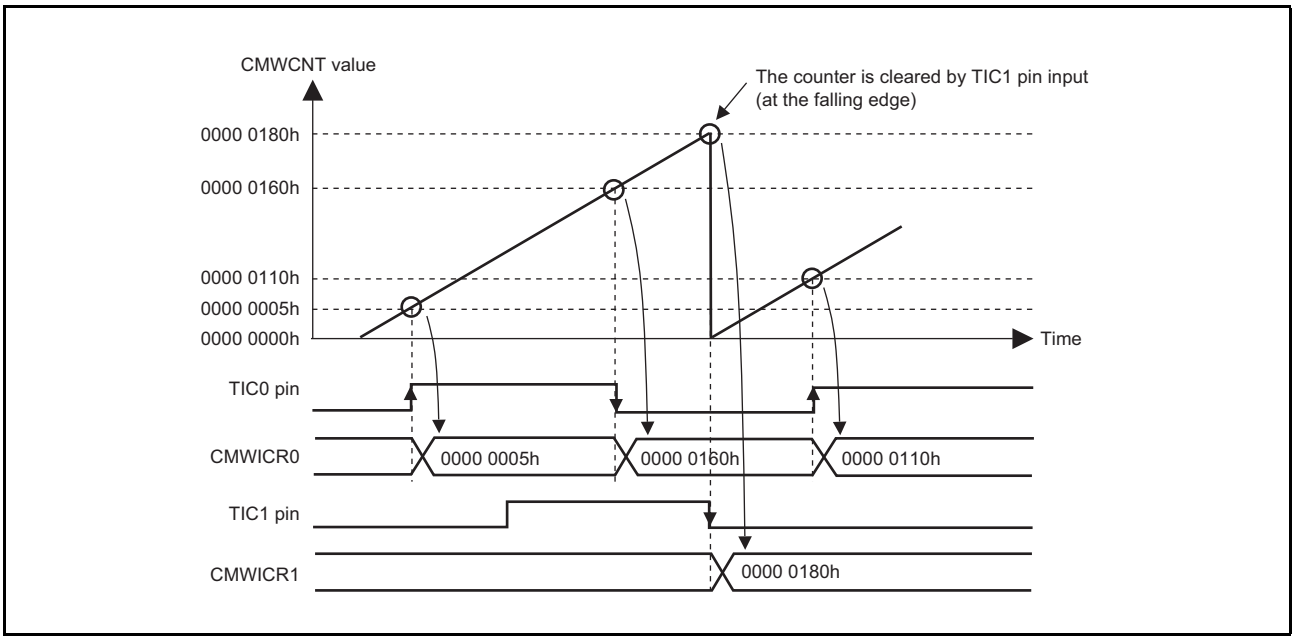


Figure 19.8 Example of Input Capture Operation

19.3.5 Counter Size

With the CMTW, either 16 or 32 bits can be selected as the counter size by using the CMWSTR.CMS bit. When the counter is used as a 32-bit counter, set the CMWCOR, CMWOCR0, or CMWOCR1 register to the desired values in 32-bit units. In reading, all 32 bits of CMWICR0 and CMWICR1 are valid. When the counter is used as a 16-bit counter, a 32-bit value should be set to the CMWCOR register with 0000h in the higher-order bits. Similarly, a 32-bit value should be set to CMWOCR0 and CMWOCR1 registers with 0000h in the higher-order bits.
A 32-bit value with 0000h in the higher-order bits is read from CMWICR0 and CMWICR1 registers.

19.3.6 Count Timing based on CMWCNT

One of four clocks (PCLKD/8, PCLKD/32, PCLKD/128, and PCLKD/512) obtained by dividing the peripheral clock (PCLKD) can be selected with the CMWCR.CKS[1:0] bits. Figure 19.9 shows the timing.

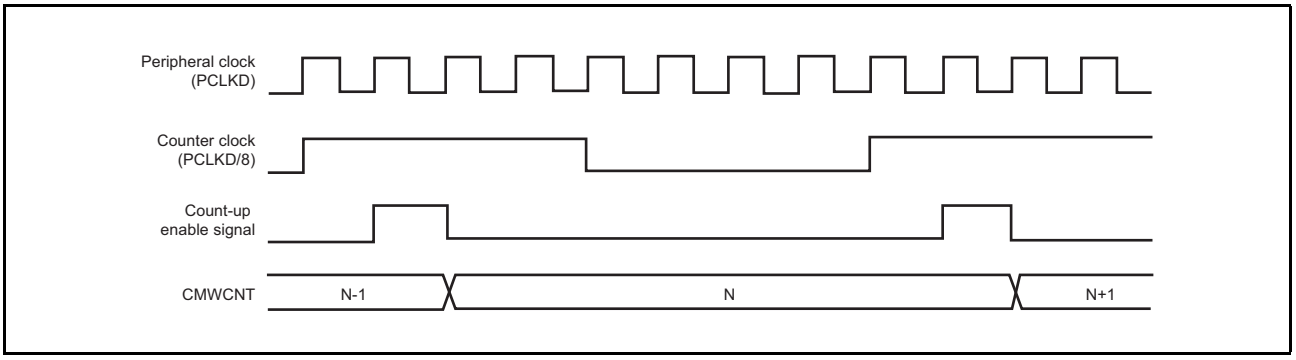


Figure 19.9 Count Timing (PCLKD/8)

19.3.7 Output Compare Output Timing

A compare match signal is generated in the last state in which the CMWOCR register and CMWCNT counter values match (the CMWCNT counter value is updated immediately after the state). That is, the compare match signal is not generated if the CMWCNT counter clock is not input after a match between the CMWOCR register and CMWCNT counter values. When a compare match signal is generated, the output compare output pin (TOC) changes in accord with the setting of the OC0 or OC1 bit in the CMWIOR register. Figure 19.10 shows output compare output timing.

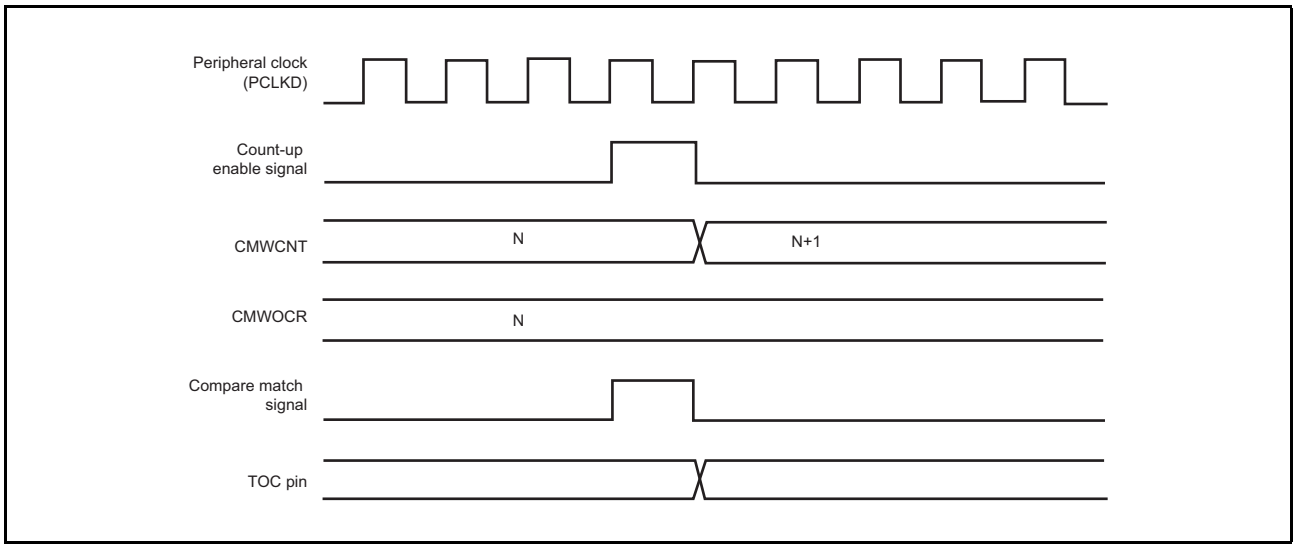


Figure 19.10 Output Compare Output Timing

19.3.8 Input Capture Signal Timing

Figure 19.11 shows the input capture timing.

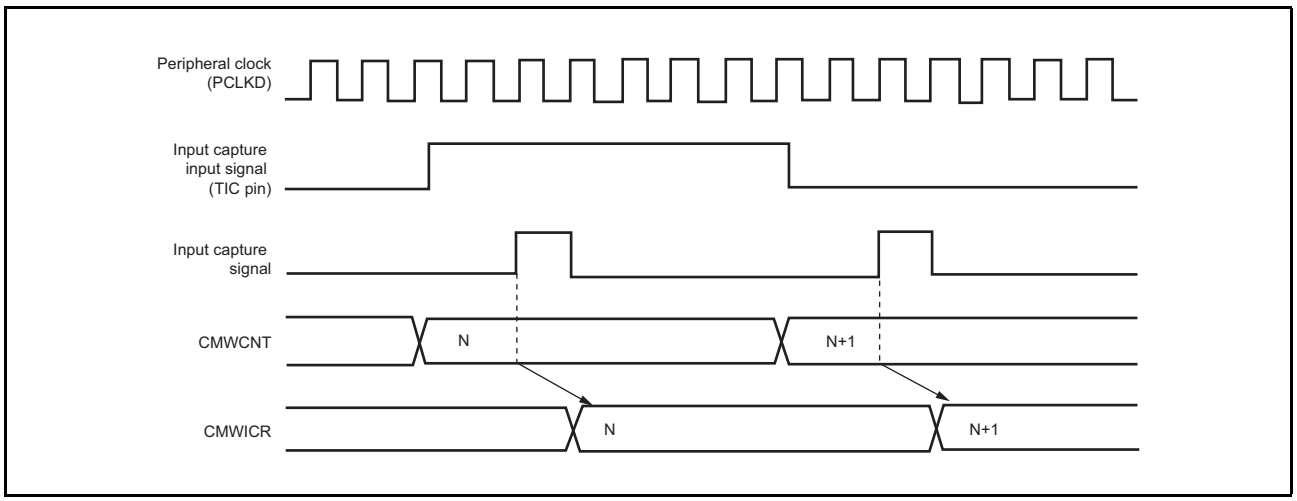


Figure 19.11 Input Capture Input Signal Timing

19.3.9 Digital Noise Filtering

The noise filter samples CMTW input-capture signals at the frequency of the sampling clock and the pulses with levels that only match once or twice are removed.
The digital noise filtering functionality includes enabling and disabling of the noise filtering for each pin. Figure 19.12 shows the timing of digital noise filtering.

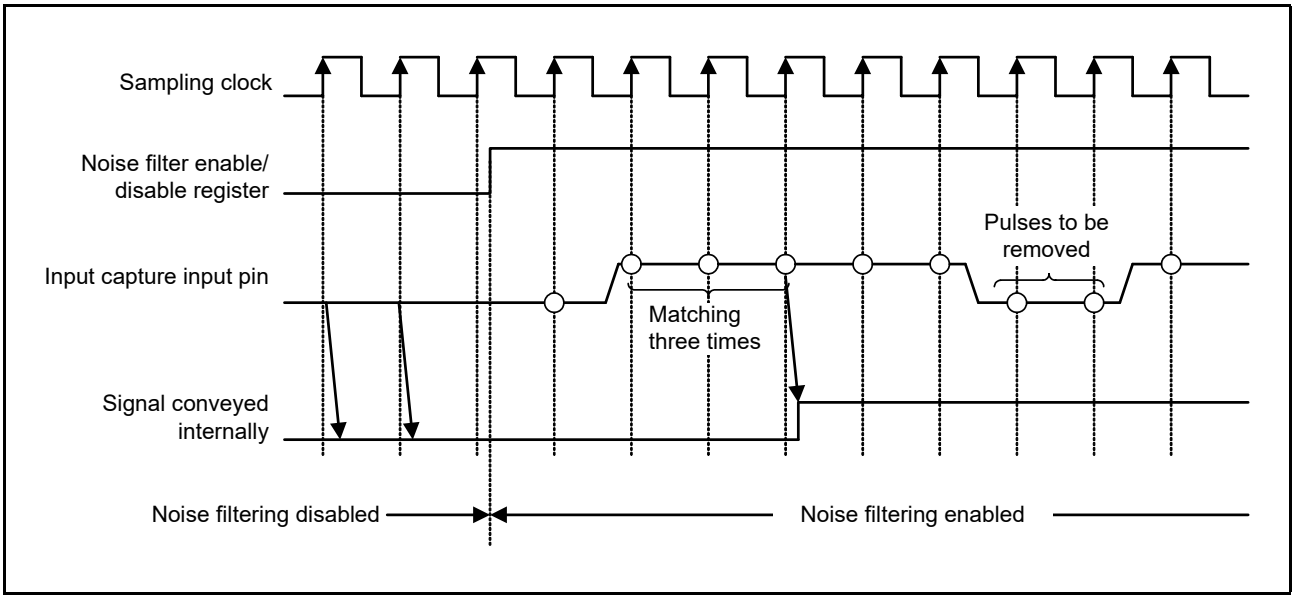


Figure 19.12 Timing of Digital Noise Filtering

If digital noise filtering is enabled, input capture proceeds on edges of the noise filtered signal after a delay of (minimum sampling interval × 2 + PCLKD) due to noise filtering of the input capture input.

19.3.10 ECM Dynamic Mode Error Output Selection

This LSI, uses an output compare signal of CMTW as an error output signal in ECM dynamic mode. That is, one of the CMTW output compare signals is selected as the error output in ECM dynamic mode.
When an actual output compare signal of CMTW is to be output to the ECM, set up the output comparison for the CMTW channels to be used in this way before setting the ECDMESLR register.
For the procedure for setting up output comparison, see section 19.3.3, Output Compare Function.
For details of ECM dynamic mode, see section 32, Error Control Module (ECM).

19.4 Interrupts

19.4.1 CMTW Interrupt Sources and DMAC Transfer Requests

The CMTW has five interrupt sources: two input capture interrupt requests (IC0In and IC1In), two output compare interrupt requests (OC0In and OC1In), and a compare match interrupt request (CMWIn) (n = 0, 1).
Table 19.3 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the IC0IE, IC1IE, OC0IE, OC1IE, and CMWIE bits in CMWCR and are separately issued to the interrupt controller.

Table 19.3 CMTW Interrupt Sources

Interrupt Source	Interrupt	Interrupt Enable Bit	DMAC Activation	Priority
CMWIn	Interrupt caused by compare match	CMWIE	Possible	High
IC0In	Interrupt caused by input capture 0	IC0IE	Possible	↑
IC1In	Interrupt caused by input capture 1	IC1IE	Possible	↕
OC0In	Interrupt caused by output compare 0	OC0IE	Possible	↓
OC1In	Interrupt caused by output compare 1	OC1IE	Possible	Low

19.4.2 Timing of Compare Match Interrupt Generation

When the values of the CMWCNT counter and CMWCOR register match, a compare match interrupt (CMWI) is generated. The compare match signal is generated at the end of the cycle where the values matched (i.e. when the CMWCNT counter is updated from the matching counter value). The compare match signal, therefore, is not generated until a further cycle of the input clock (PCLKD/8, PCLKD/32, PCLKD/128, or PCLKD/512) for the CMWCNT counter arrives after the values of the CMWCNT counter and CMWCOR register have matched. Figure 19.13 shows the timing of compare match interrupt generation.

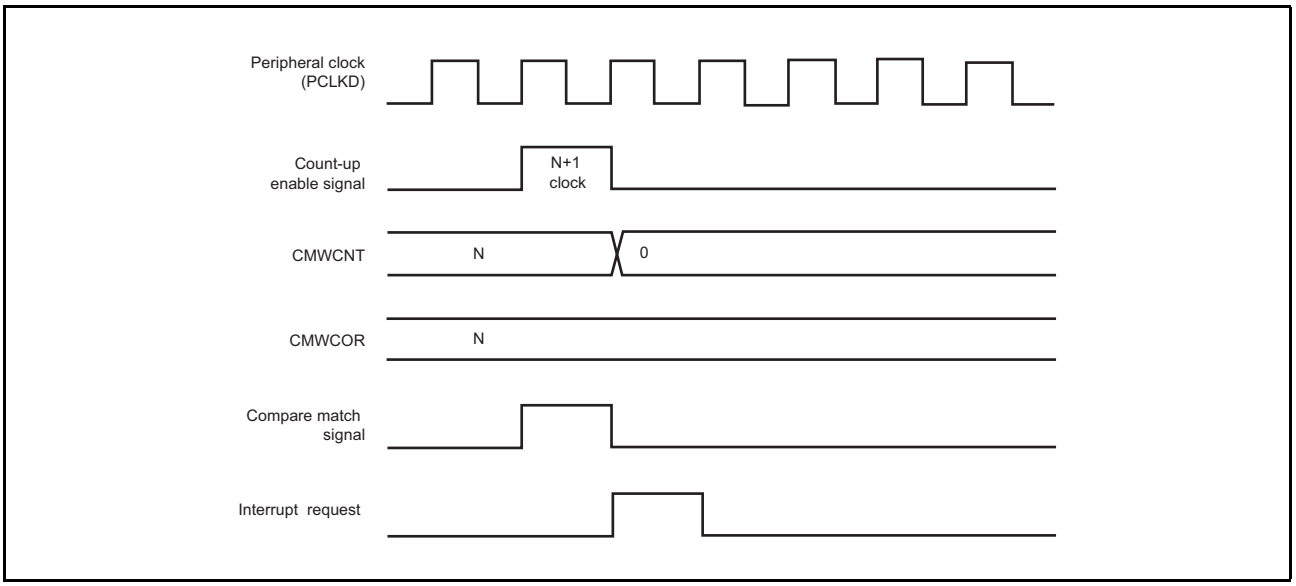


Figure 19.13 Timing of Compare Match Interrupt Generation

19.4.3 Timing of Output Compare Interrupt Generation

Figure 19.14 shows the timing of output compare interrupt generation.

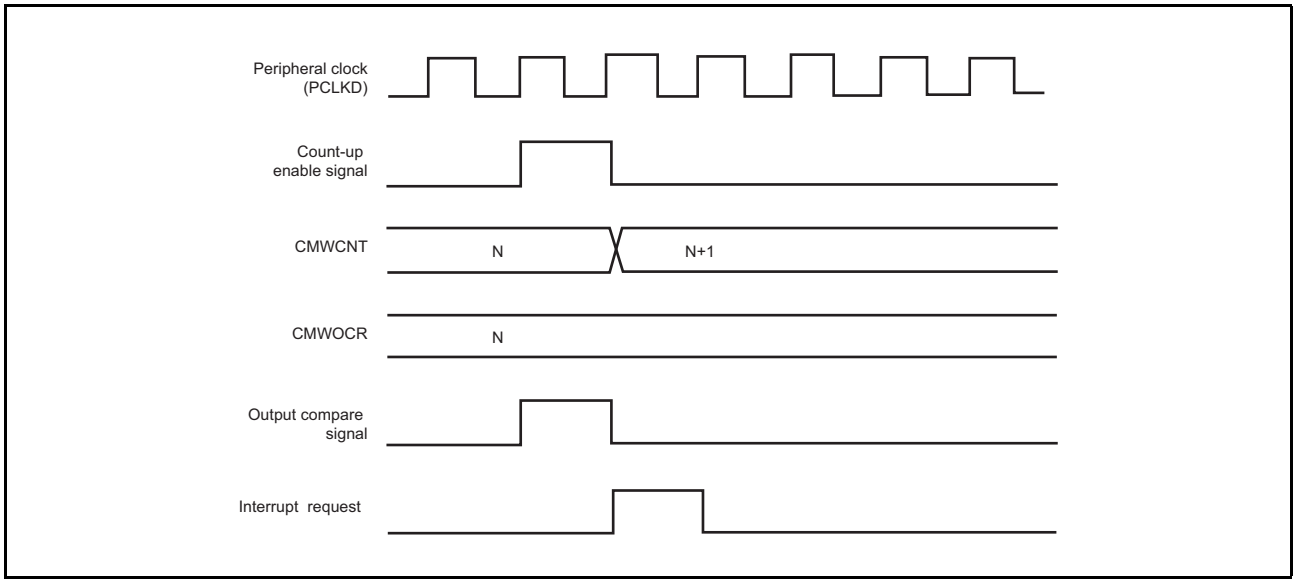


Figure 19.14 Timing of Output Compare Interrupt Generation

19.4.4 Timing of Input Capture Interrupt Generation

Figure 19.15 shows the timing of input capture interrupt generation.

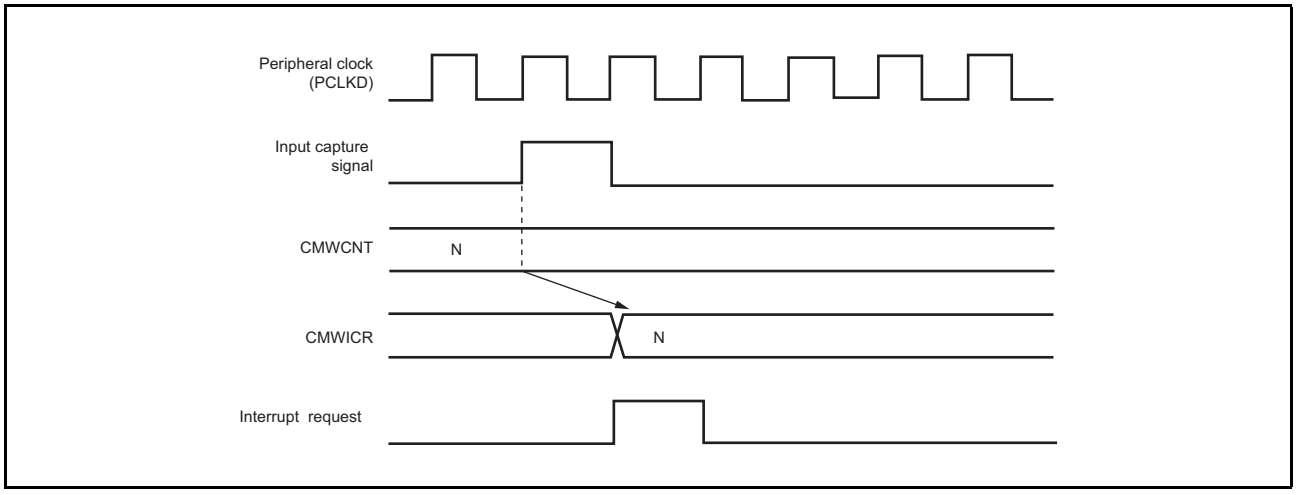


Figure 19.15 Timing of Input Capture Interrupt Generation

19.5 Event Link Operation

19.5.1 Issuing Events to the ELC

The CMTW can issue event signals to the event link controller (ELC) in response to the following events.

Compare Match Event

In response to a compare match, the CMTW simultaneously issues an interrupt request and a compare match event signal to the ELC. The event signal is issued regardless of the settings of the corresponding interrupt request enable bit (CMWCR.CMWIE bit).

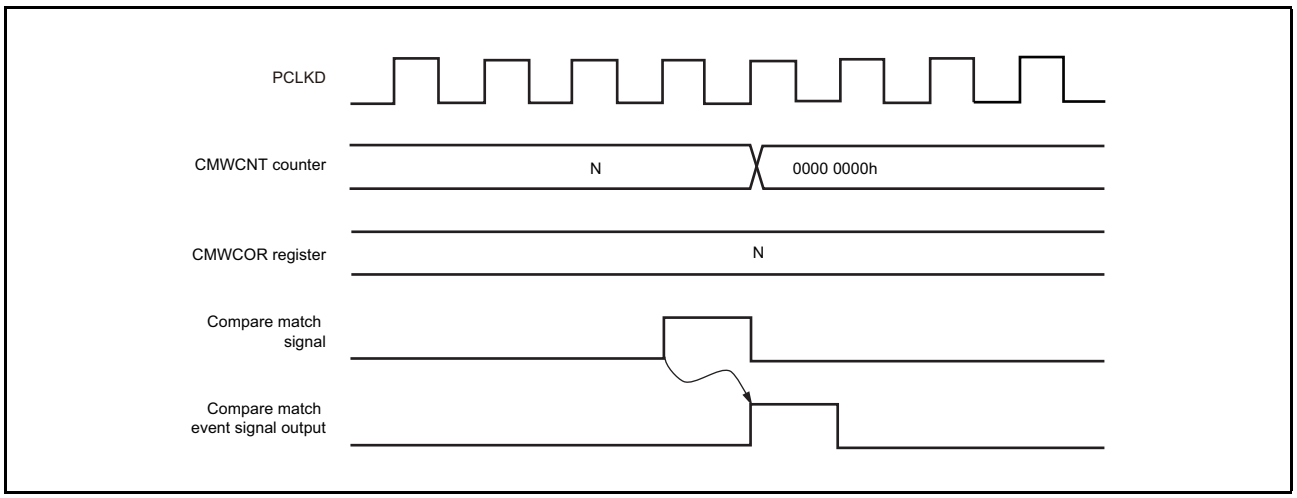


Figure 19.16 Timing of Issuing a Compare Match Event Signal

19.5.2 Actions on Acceptance of Event Signals from ELC

CMTW can respond with any of the following three actions when the action is set in the event link controller (ELC) and CMTW accepts the event signal.

(1) Start Counting

When an event signal is received while starting to count is the selected action, the STR bit in CMWSTR (the timer start register) is set to 1 and counting starts.

However, the event signal is ignored if it is accepted when the CMWSTR.STR bit has already been set to 1.

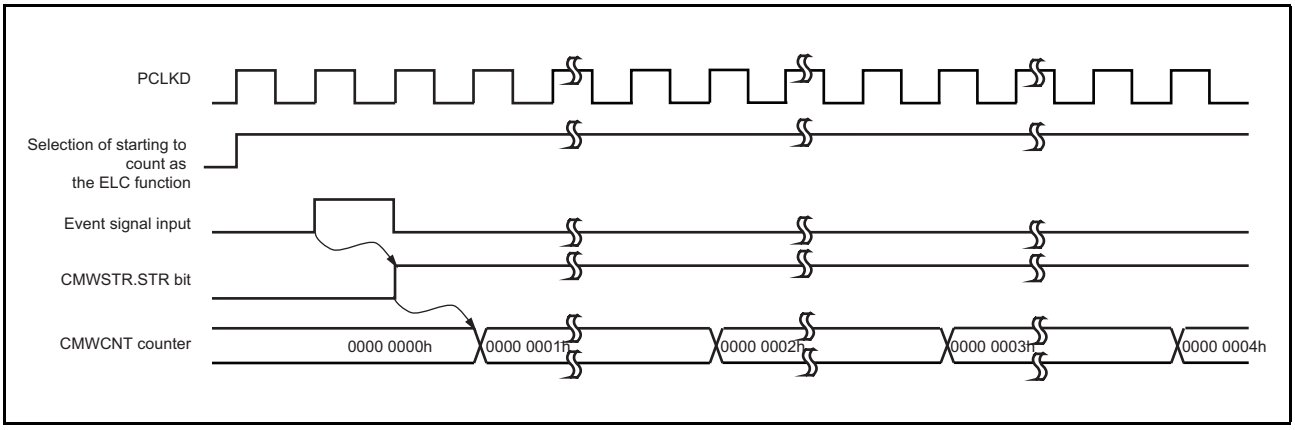


Figure 19.17 Starting to Count on Acceptance of the Event Signal

(2) Event Counting

When an event signal is received while counting events is the selected action, CMWCNT (the timer counter) is incremented. In this case, however, the STR bit in CMWSTR (the timer start register) must be set to 1 before the event signal is received.

In event counting, the setting of the CKS[1:0] bit in CMWCR (the timer control register) is not effective.

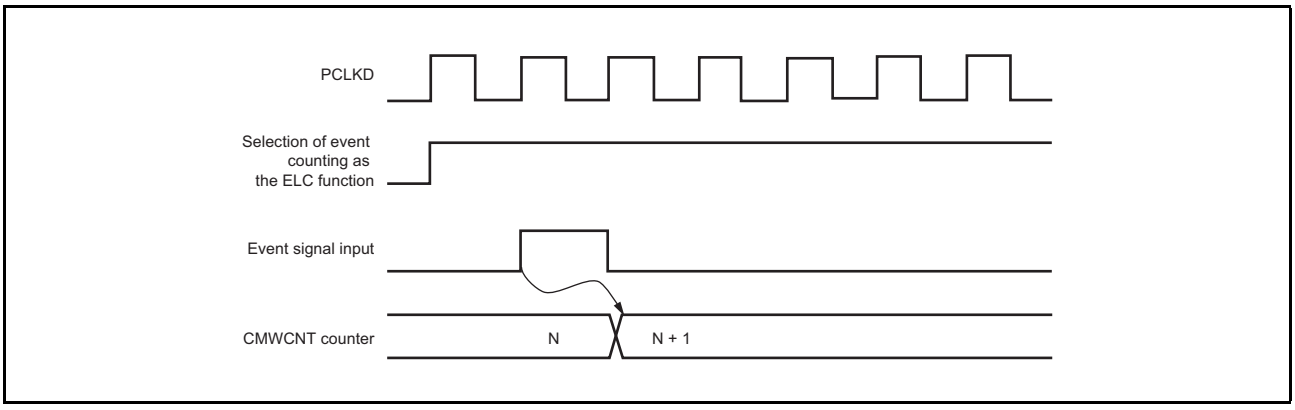


Figure 19.18 Counting an Event on Acceptance of the Event Signal

(3) Clear a Counter

Clearing of the CMTW counter is selected in the ELOPH register of the ELC. When the event specified in the corresponding ELSRn register is generated, the timer counter (CMWCNT) is returned to its initial value (0000h). Counting continues, however, if the setting of the STR bit of the timer start register (CMWSTR) is 1 at this time, so counting can be automatically restarted in this way. A timing chart of restarting the counter is shown in Figure 19.19

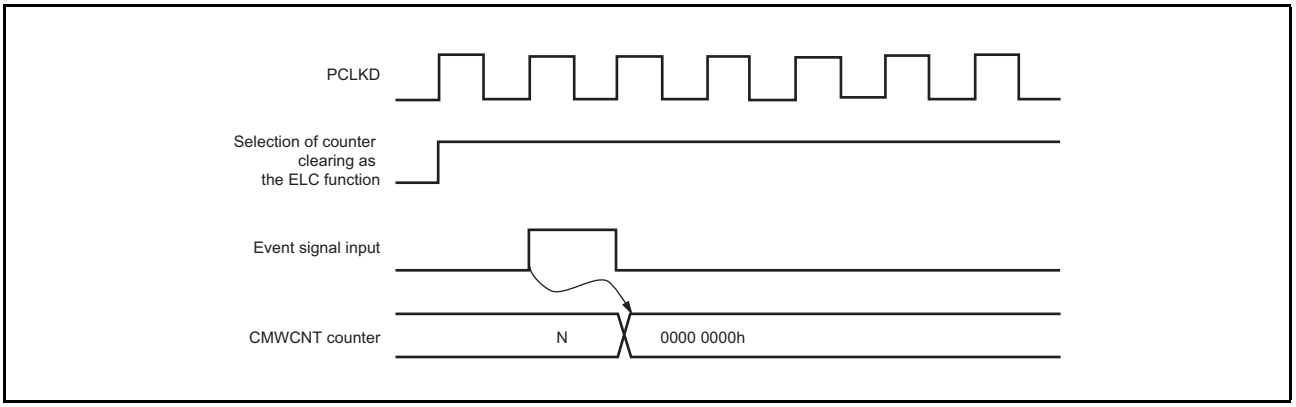


Figure 19.19 Restarting Counting on Acceptance of the Event Signal

19.6 Usage Notes

19.6.1 Module-Stop Function

The CMTW operation can be enabled or disabled by using the module-stop control register (MSTPCRA). In the initial setting, the CMTW is in the module-stop state. Register access is enabled by releasing the CMTW from the module-stop state. For details, refer to section 9, Low-Power Consumption Function.

19.6.2 Contention between CMWCNT Counter Writing and Compare Match

If the compare match signal is generated during CPU writing to the CMWCNT counter, the compare match request is output but the counter is not cleared since the CPU writing to the counter is given priority.
Figure 19.20 shows the timing of contention between CMWCNT counter writing and compare match.

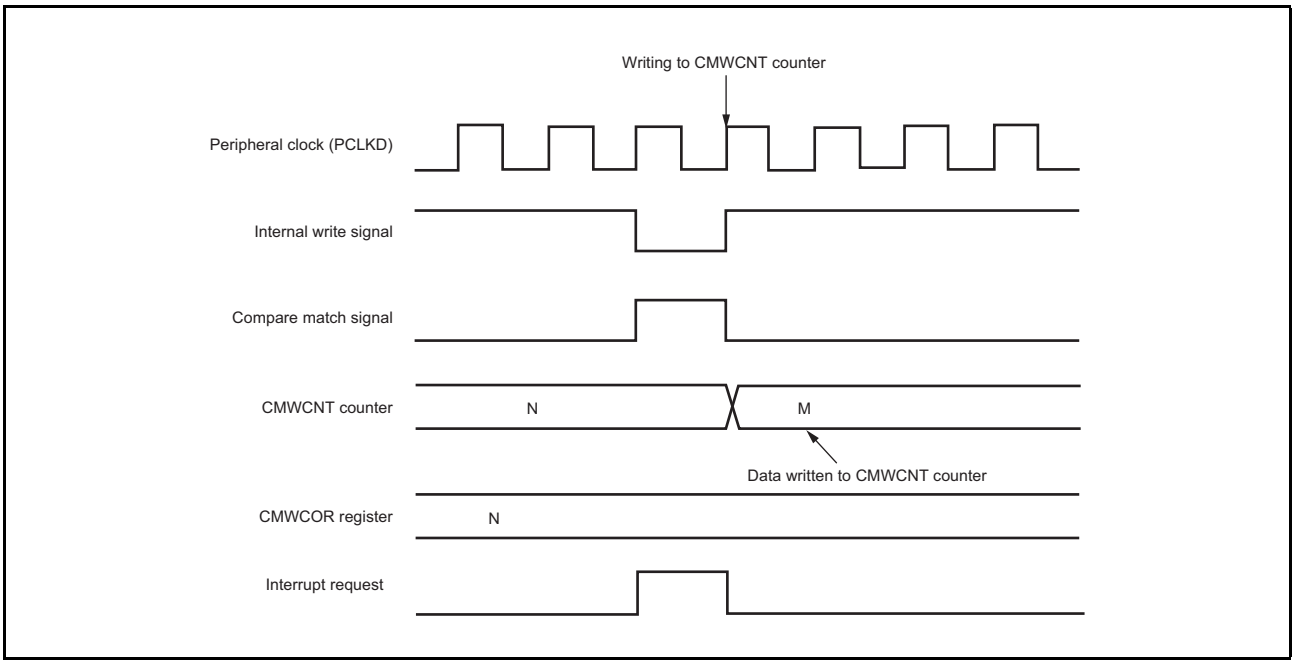


Figure 19.20 Contention between CMWCNT Counter Writing and Compare Match

19.6.3 Contention between CMWCNT Counter Writing and Incrementing or Clearing

In case of contention between incrementation or clearing of the CMWCNT counter and CPU writing to the counter, the counter is not actually incremented or cleared since the CPU writing to the CMWCNT counter is given priority.
Figure 19.21 shows the timing in the case of contention between writing to the CMWCNT counter and incrementation or clearing.

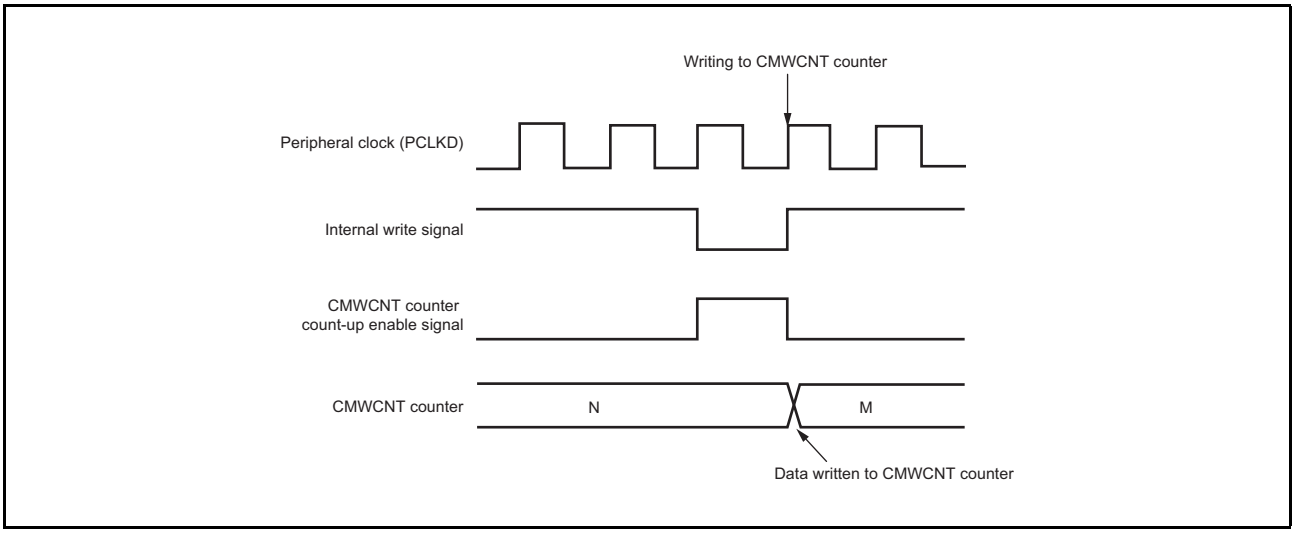


Figure 19.21 Contention between CMWCNT Counter Writing and Incrementing

19.6.4 Contention between CMWCOR Register Writing and Compare Match

If the compare match is generated during CPU writing to the CMWCOR register, the CPU writing to the CMWCOR register proceeds and also the compare match signal is output.
Figure 19.22 shows the timing of contention between CMWCOR register writing and compare match.

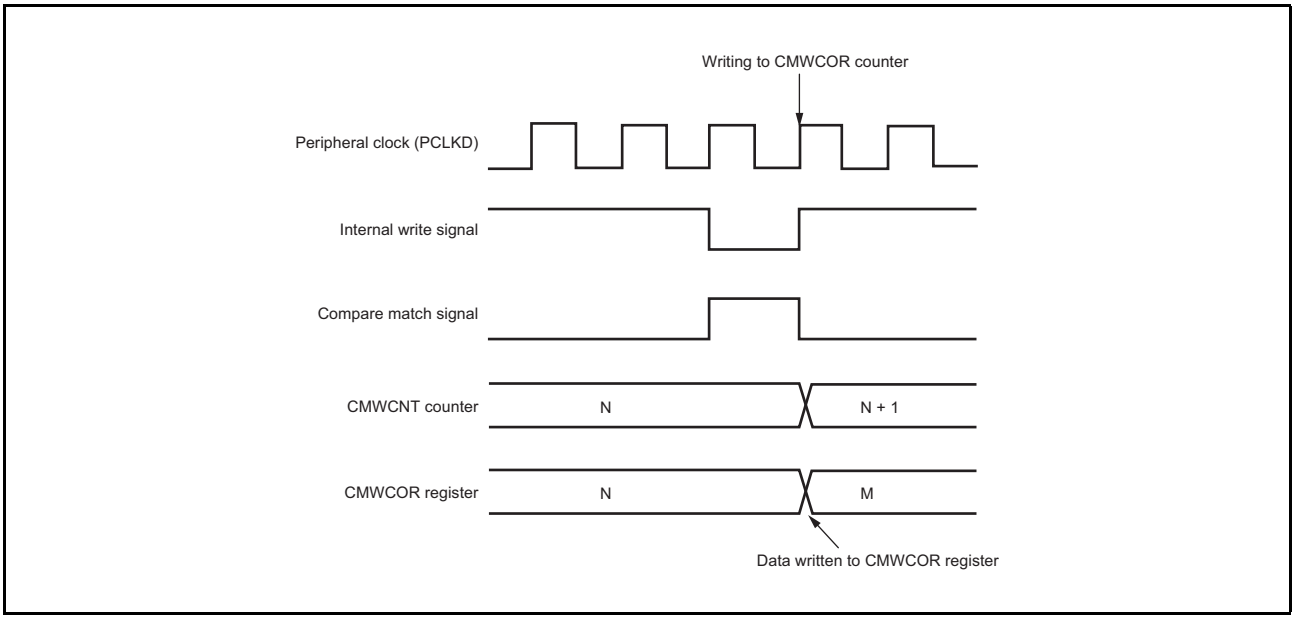


Figure 19.22 Contention between CMWCOR Register Writing and Compare Match

19.6.5 Contention between CMWOCR Register Writing and Compare Match

If the compare match is generated during CPU writing to the CMWOCR register, the CPU writing to the CMWOCR register proceeds and also the compare match signal is output.
Figure 19.23 shows the timing of contention between CMWOCR register writing and compare match.

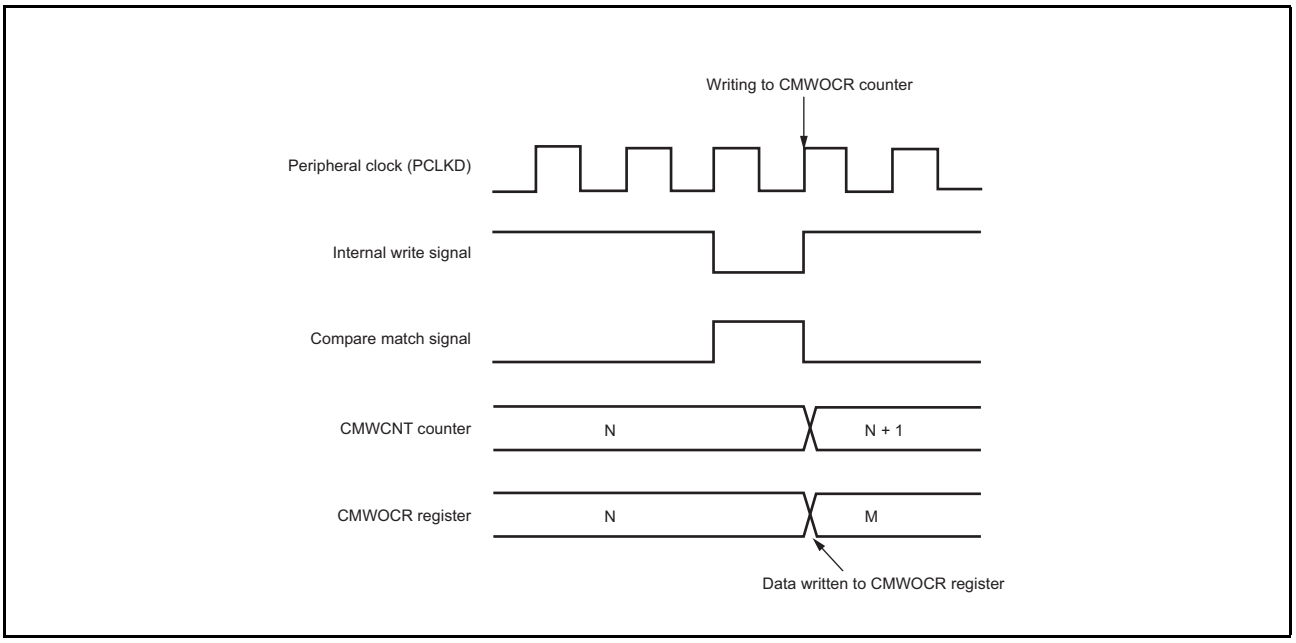


Figure 19.23 Contention between CMWOCR Register Writing and Compare Match

19.6.6 Contention between CMWCNT Counter Reading and Incrementing or Clearing

If the CMWCNT counter incrementing or clearing process occurs at the same time that the data of the CMWCNT counter is read, the value having been in the CMWCNT counter before incremented or cleared is read.
Figure 19.24 shows the timing of contention between the CMWCNT counter reading and incrementing.

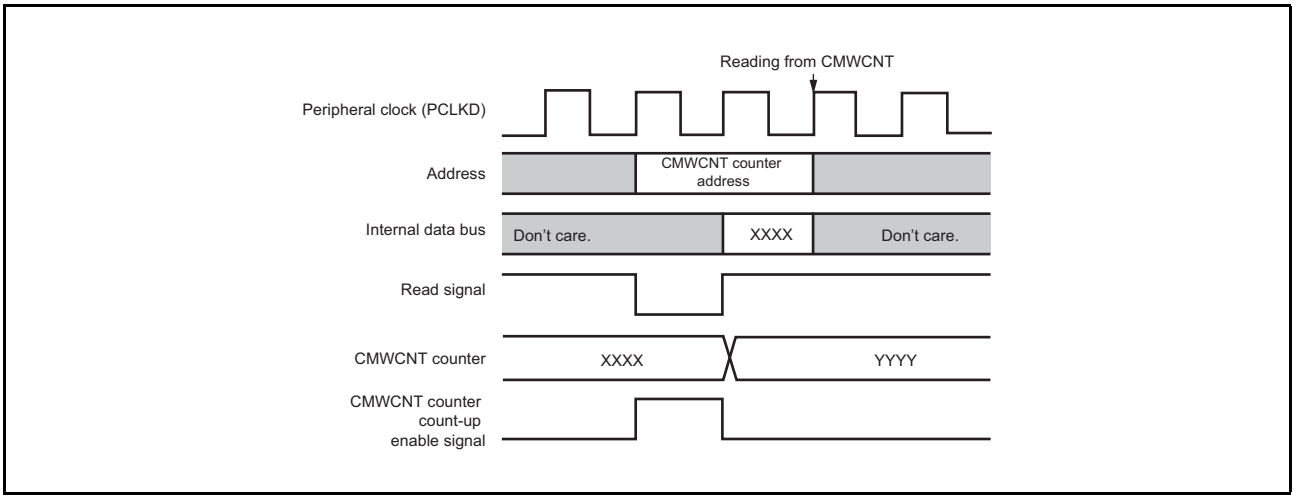


Figure 19.24 Contention between CMWCNT Counter Reading and Incrementing (When the Data Reading and Incrementing Process Occur Simultaneously)

19.6.7 Contention between CMWICR Register Reading and Input Capture

If the input capture signal is generated at the same time that the data of CMWICR register is read, the value having been in CMWICR register before updated by input capture transfer is read.

Figure 19.25 shows the timing of contention between CMWICR register reading and input capture.

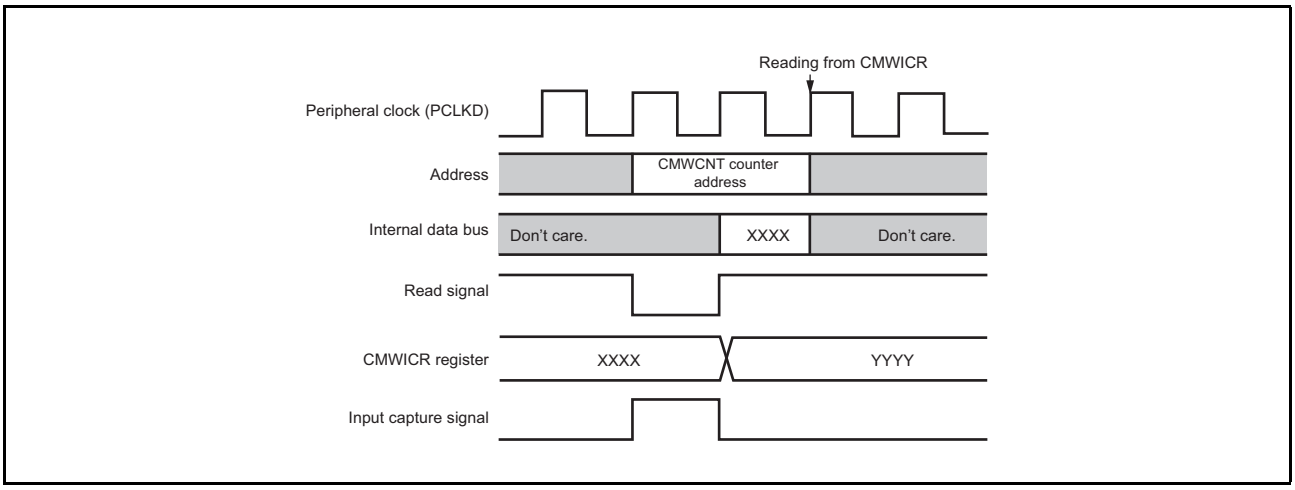


Figure 19.25 Contention between CMWICR Register Reading and Input Capture (When the Input Capture Signal and Read Signal are Generated Simultaneously)

19.6.8 Contention between Event Link Operation and Register Access

The followings are the notes on using CMTW for event link operations.
Table 19.4 summarizes contention between operations due to the event link, access to registers, and changes to the counter's state.

(1) Start Counting

When writing to the STR bit in the CMWSTR register and acceptance of the event signal are in contention, the CPU writing to the STR bit is ignored since setting of the STR bit to 1 in response to the event is given priority.

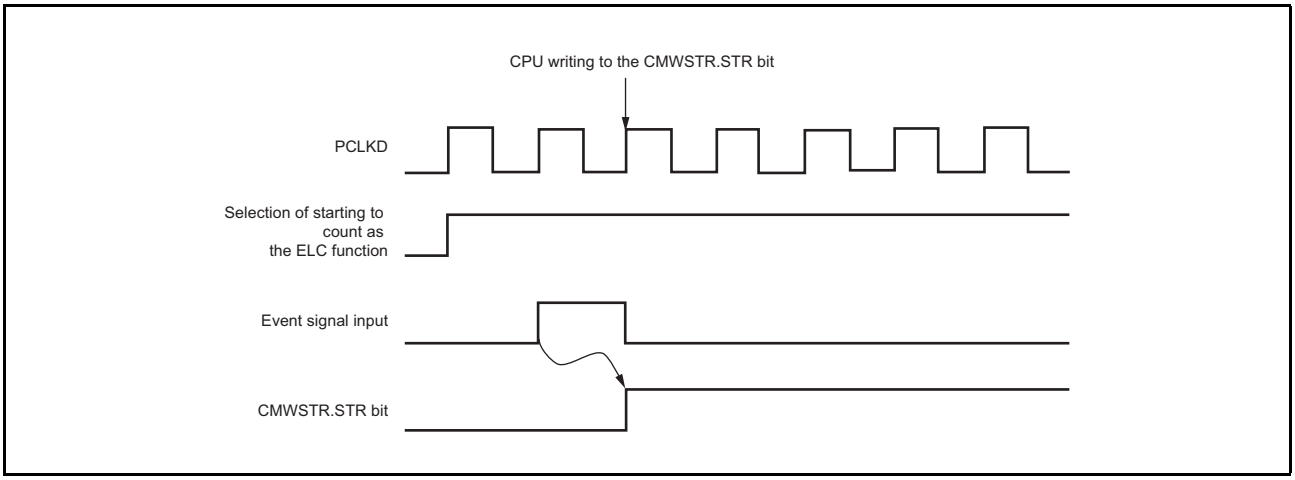


Figure 19.26 Contention between Event Acceptance and Register Access in Counting Start Operation

(2) Event Counting

When writing to CMWCNT (the timer counter) and acceptance of the event signal are in contention, the CPU writing to the CMWCNT counter is ignored since the counting operation in response to the event is given priority.

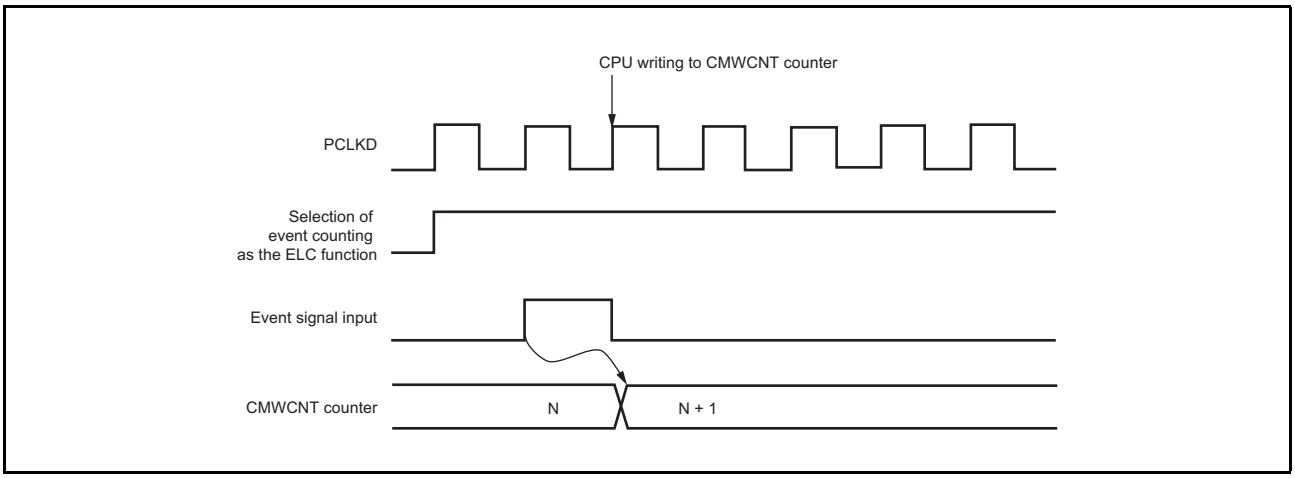


Figure 19.27 Contention between Event Acceptance and Register Access in Event Counting Operation

(3) Clear a Counter

When writing to CMWCNT (the timer counter) and acceptance of the event signal are in contention, the CPU writing to the CMWCNT counter is ignored since the counter value initialization in response to the event occurrence is given priority.

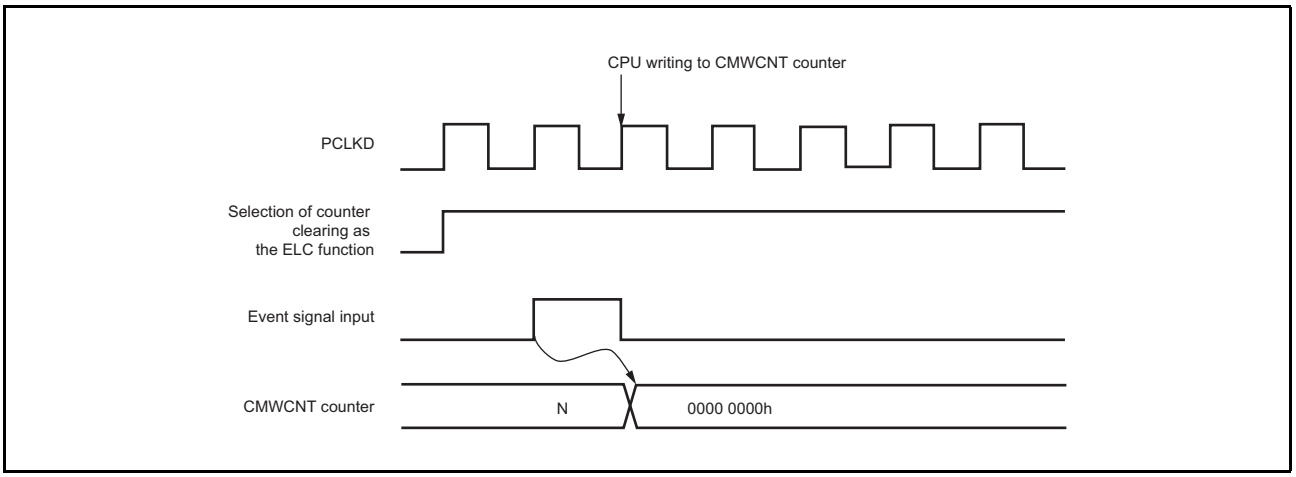


Figure 19.28 Contention between Event Acceptance and Register Access in Counting Clear Operation

Table 19.4 Summary of Contention between Operations Due to the Event Link, Access to Registers, and Changes to the Counter's State

Event Link Operation	Register Access	CMWCNT State	CMWICR0/1 State	Operation to be Performed
Counting start	Writing to CMWSTR.STR	Stopped state	—	Counting start
		Operating	—	Counting start
		Compare match	—	Counting start (CMWSTR.STR retains 1) and compare match
		Counting up	—	Counting start (CMWSTR.STR retains 1) and counting up
Event counting	Writing to CMWCNT	—	—	Event counting
	Writing to CMWCOR	Compare match	—	Compare match
Counting clear	Writing to CMWCNT	Other than compare match	—	Counting clear
	Writing to CMWCNT	Compare match	—	Compare match and counter clear
	(No access to registers)	Compare match	—	Compare match and counter clear
(No events)	Writing to CMWCNT	Compare match	—	Output of compare match interrupt request / Writing to CMWCNT
		Counting up	—	Writing to CMWCNT
	Writing to CMWCOR	Compare match	—	Compare match
	Writing to CMWOCR0	Output compare 0	—	Output compare 0
	Writing to CMWOCR1	Output compare 1	—	Output compare 1
	Reading from CMWCNT	Counting up	—	Counting up and reading of the previous value
	Reading from CMWICR0	—	Input capture 0	Input capture 0 and reading of the value before transfer
	Reading from CMWICR1	—	Input capture 1	Input capture 1 and reading of the value before transfer

20. Watchdog Timer (WDTA)

The watchdog timer (WDT) contains a 14-bit down-counter. If the counter underflows, an error notification to the error control module (ECM) is generated. The count value of the down-counter can be refreshed to the value after reset to enable counting to start again. Counter refreshing can be preformed during a period (the refresh-permitted period) you specify. If you perform refresh (register writing) outside of the permitted period, an error notification is generated and sent to the ECM. This enables detection of runaway of the program, taking the refresh interval into consideration. If underflow occurs or if refresh is performed outside of the refresh-permitted period, the WDT stops counting. Counting restarts after refresh is performed. (For details on refresh operation, see section 20.3.3, Refresh Operation.)
For details on the error control module (ECM), see section 32, Error Control Module (ECM).

20.1 Overview

WDT starts counting when refresh (register writing) is performed after reset is released.
Before starting the count, it is necessary to set the clock division ratio, the window start/end positions, and the timeout period in the WDT control register (WDTCR).
Table 20.1 lists the specifications of the WDT and Figure 20.1 shows a block diagram of the WDT.

Table 20.1 WDT Specifications

Item	Specifications
Number of internal channels	One channel
Count source	Peripheral clock (PCLKE)
Clock division ratio	Divide by 4, 64, 128, 512, 2048, or 8192
Counter operation	Counting down using a 14-bit down-counter
Conditions for starting the counter	Refresh (writing 00h and then FFh to the WDTRR register)
Conditions for stopping the counter	• Reset (the down-counter and other registers return to the values after reset) • A counter underflows or a refresh error occurs
Window function	Window start and end positions can be specified (refresh-permitted period)
Sources of sending an error notification to ECM	• Down-counter underflows • Refreshing outside the refresh-permitted period (refresh error)
Reading the counter value	The down-counter value can be read by the WDTSR register.
WDT register control	• Selection of clock division ratio after refresh operation (WDTCR.CKS[3:0] bits) • Selection of timeout period of the watchdog timer (WDTCR.TOPS[1:0] bits) • Selection of window start position of the watchdog timer (WDTCR.RPSS[1:0] bits) • Selection of window end position of the watchdog timer (WDTCR.RPES[1:0] bits)

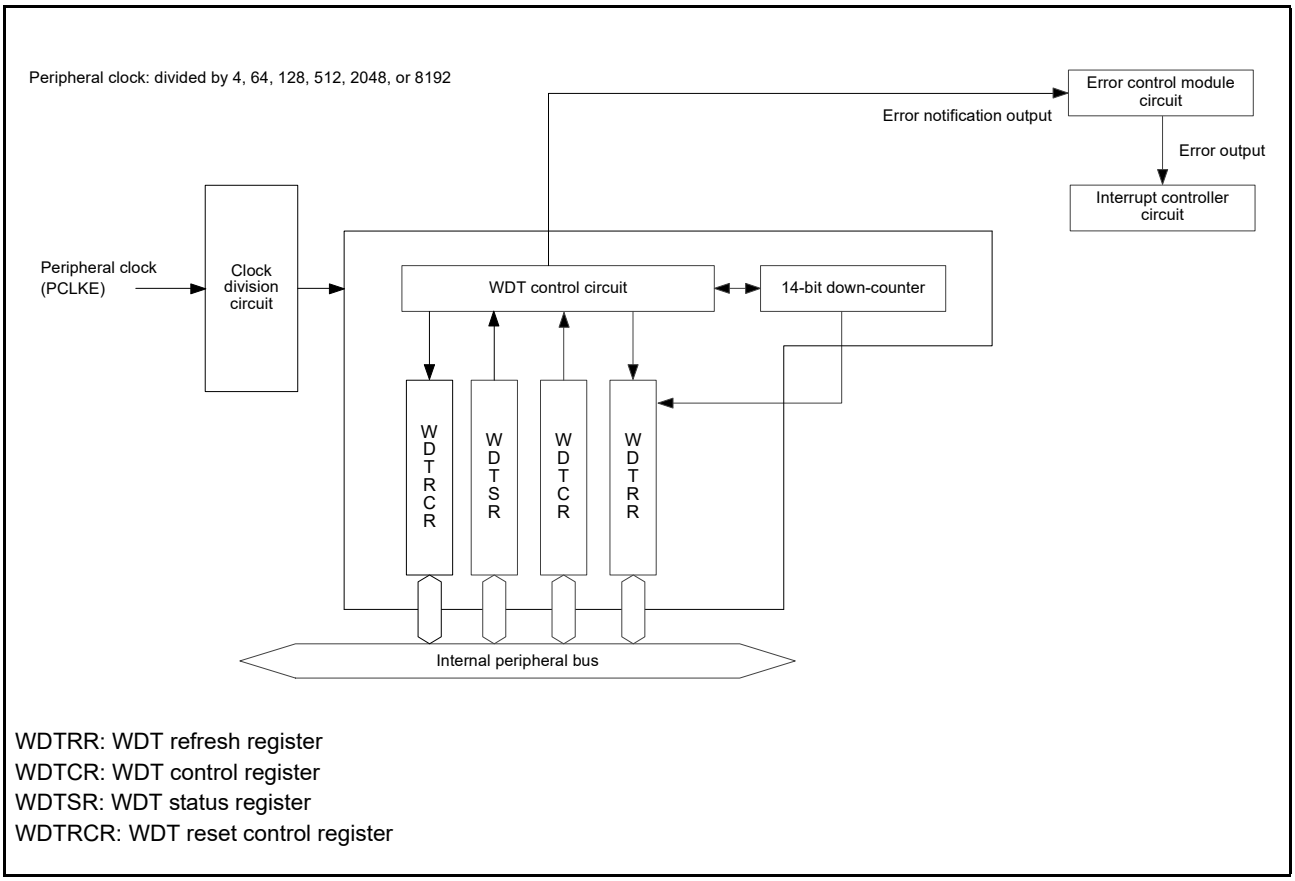


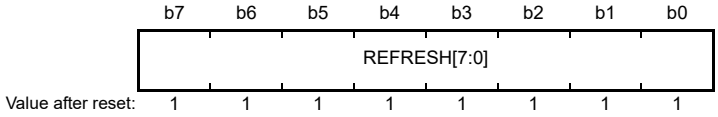
Figure 20.1 WDT Block Diagram

20.2 Register Descriptions

20.2.1 WDT Refresh Register (WDTRR)

The WDTRR register refreshes the down-counter of the WDT. To refresh the down-counter of the WDT, write 00h and then FFh (refresh operation) to the WDTRR register in the refresh-permitted period. After being refreshed, the down-counter starts counting down from the value specified with the WDTCR.TOPS[1:0] bits.
When 00h is written, the read value is 00h. When a value other than 00h is written, the read value is always FFh.
For details on the refresh operation, see section 20.3.3, Refresh Operation.

Address(es): WDT0.WDTRR A008 0600h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	REFRESH [7:0]	Refresh Register	The down-counter is refreshed by writing 00h and then writing FFh to this register.	R/W

20.2.2 WDT Control Register (WDTCR)

The WDTCR register allows you to select a timeout period before the down-counter underflows, a clock division ratio, and a window start/end position for refreshing. There are some restrictions on writing to this register. For details, see section 20.3.2, Control over Writing to the WDTCR and WDTRCR Registers.

Address(es): WDT0.WDTCR A008 0602h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	RPSS[1:0]		—	—	RPES[1:0]		CKS[3:0]				—	—	TOPS[1:0]	
Value after reset:	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TOPS[1:0]	Timeout Period Selection	b1 b0 0 0: 1,024 cycles (03FFh) 0 1: 4,096 cycles (0FFFh) 1 0: 8,192 cycles (1FFFh) 1 1: 16,384 cycles (3FFFh) Values within parentheses are down-count start values.	R/W
b3, b2	—	Reserved	These bits are read as 0 and cannot be modified.	R/W
b7 to b4	CKS[3:0]	Clock Division Ratio Selection	b7 b4 0 0 0 1: PCLKE/4 0 1 0 0: PCLKE/64 1 1 1 1: PCLKE/128 0 1 1 0: PCLKE/512 0 1 1 1: PCLKE/2048 1 0 0 0: PCLKE/8192 Other settings are prohibited.	R/W
b9, b8	RPES[1:0]	Window End Position Selection	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (window end position is not specified)	R/W
b11, b10	—	Reserved	These bits are read as 0 and cannot be modified.	R/W
b13, b12	RPSS[1:0]	Window Start Position Selection	These bits allow selection of a window start position of the down-counter from among 100%, 75%, 50%, and 25% of the counting period (assuming that the count start position as 100% and underflow occurrence as 0%). The period from the window start position to the window end position is the refresh-permitted period, and any other periods are the refresh-prohibited periods. Figure 20.2 shows the relation between the settings of the RPSS[1:0] and RPES[1:0] bits and the refresh-permitted and refresh-prohibited periods. b13 b12 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (window start position is not specified)	R/W
b15, b14	—	Reserved	These bits are read as 0 and cannot be modified.	R/W

TOPS[1:0] Bits (Timeout Period Selection)

These bits allow you to select a timeout period (period before the down-counter underflows) from among 1024, 4096, 8192, and 16384 cycles, taking the divided clock specified by the CKS[3:0] bits as one cycle. After the down-counter is refreshed, the combination of the CKS[3:0] and TOPS[1:0] bits determines the time (the number of peripheral clock (PCLKE) cycles) before the counter underflows.

Table 20.2 lists relations between the CKS[3:0] and TOPS[1:0] bit settings, the timeout period, and the number of PCLKE cycles.

CKS[3:0] Bits (Clock Division Ratio Selection)

These bits allow you to select a division ratio from among the peripheral clocks (PCLKE) divided by 4, 64, 128, 512, 2048, and 8192. Combined with the TOPS[1:0] bit setting, a counting period between 4,096 and 134,217,728 cycles of the peripheral clock (PCLKE) can be selected for the WDT.

RPES[1:0] Bits (Window End Position Selection)

These bits allow selection of a window end position from among 75%, 50%, 25%, and 0% of the counting period. The selected window end position should be a value smaller than the window start position (window start position > window end position). If the window end position is greater than the window start position, only the window start position setting is valid.

The counter values of the window start and end positions specified by the RPES[1:0] and RPSS[1:0] bit settings vary according to the TOPS[1:0] bit setting.

Table 20.3 lists the counter values of the window start and end positions corresponding to the TOPS[1:0] bit setting.

RPSS[1:0] Bits (Window Start Position Selection)

These bits allow selection of a window start position of the down-counter from among 100%, 75%, 50%, and 25% of the counting period (assuming that the count start position as 100% and underflow occurrence as 0%). The period from the window start position to the window end position is the refresh-permitted period, and any other periods are the refresh-prohibited periods.

Figure 20.2 shows the relation between the settings of the RPSS[1:0] and RPES[1:0] bits and the refresh-permitted and refresh-prohibited periods.

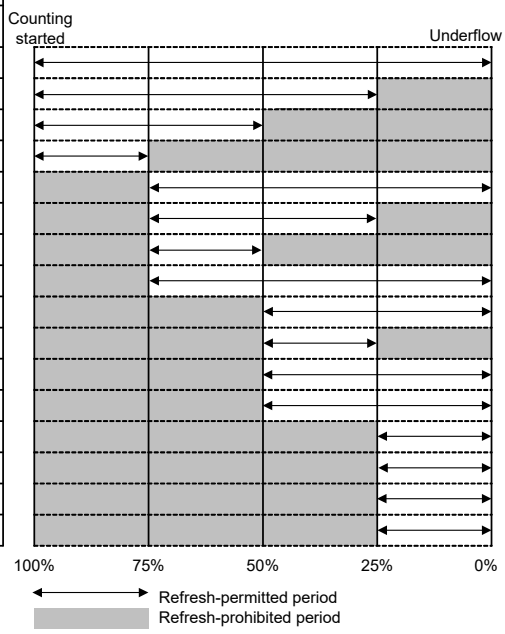
Table 20.2 Timeout Period Settings

CKS[3:0] Bits				TOPS[1:0] Bits		Clock Division Ratio	Timeout Period (Number of Cycles)	No. of Peripheral Clock (PCLKE) Cycles @Count clock = PCLKE
b7	b6	b5	b4	b1	b0			
0	0	0	1	0	0	Count clock/4	1024	4096
				0	1		4096	16384
				1	0		8192	32768
				1	1		16384	65536
0	1	0	0	0	0	Count clock/64	1024	65536
				0	1		4096	262144
				1	0		8192	524288
				1	1		16384	1048576
1	1	1	1	0	0	Count clock/128	1024	131072
				0	1		4096	524288
				1	0		8192	1048576
				1	1		16384	2097152
0	1	1	0	0	0	Count clock/512	1024	524288
				0	1		4096	2097152
				1	0		8192	4194304
				1	1		16384	8388608
0	1	1	1	0	0	Count clock/2048	1024	2097152
				0	1		4096	8388608
				1	0		8192	16777216
				1	1		16384	33554432
1	0	0	0	0	0	Count clock/8192	1024	8388608
				0	1		4096	33554432
				1	0		8192	67108864
				1	1		16384	134217728

Table 20.3 Relationship between Timeout Period and Window Start/End Counter Values

TOPS[1:0] Bits		Timeout Period		Window Start/End Counter Value			
b1	b0	Cycles	Counter Value	100%	75%	50%	25%
0	0	1024	03FFh	03FFh	02FFh	01FFh	00FFh
0	1	4096	0FFFh	0FFFh	0BFFh	07FFh	03FFh
1	0	8192	1FFFh	1FFFh	17FFh	0FFFh	07FFh
1	1	16384	3FFFh	3FFFh	2FFFh	1FFFh	0FFFh

RPSS[1:0] Bits		RPES[1:0] Bits		Window	
b13	b12	b9	b8	Start (%)	End (%)
1	1	1	1	100	0
		1	0		25
		0	1		50
		0	0		75
1	0	1	1	75	0
		1	0		25
		0	1		50
		0	0		75
0	1	1	1	50	0
		1	0		25
		0	1		50
		0	0		75
0	0	1	1	25	0
		1	0		25
		0	1		50
		0	0		75



Note: If window end setting ≥ window start setting, the window end setting is set to 0%.

Figure 20.2 RPSS[1:0] and RPES[1:0] Bit Settings and the Refresh-Permitted and Refresh-Prohibited Periods

20.2.3 WDT Status Register (WDTSR)

The WDTSR register shows the counter value of the down-counter and whether an underflow or refresh error has occurred.

Address(es): WDT0.WDTSR A008 0604h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	REFEF	UNDFF	CNTVAL[13:0]													
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b13 to b0	CNTVAL[13:0]	Down-Counter Value	Value counted by the down-counter	R
b14	UNDFF	Underflow Flag	0: No underflow occurred 1: Underflow occurred	R/(W)
b15	REFEF	Refresh Error Flag	0: No refresh error occurred 1: Refresh error occurred	R/(W)

CNTVAL[13:0] Bits (Down-Counter Value)

Read these bits to confirm the value of the down-counter, but note that the read value may differ from the actual count by a value of one count.

UNDFF Flag (Underflow Flag)

Read this bit to confirm whether an underflow has occurred in the down-counter.
The value 1 indicates that the down-counter has underflowed. The value 0 indicates that the down-counter has not underflowed.
Write 0 to the UNDFF flag to set the value to 0. Writing 1 has no effect.

REFEF Flag (Refresh Error Flag)

Read this bit to confirm whether a refresh error (performing a refresh operation during a refresh-prohibited period) has occurred.
The value 1 indicates that a refresh error has occurred. The value 0 indicates that no refresh error has occurred.
Write 0 to the REFEF flag to set the value to 0. Writing 1 has no effect.

20.2.4 WDT Reset Control Register (WDTRCR)

The WDTRCR register controls whether to send an error notification to the error control module (ECM) when underflow occurs in the down-counter of WDT.

There are some restrictions on writing to this register. For details, see section 20.3.2, Control over Writing to the WDTCR and WDTRCR Registers.

Address(es): WDT0.WDTRCR A008 0606h

	b7	b6	b5	b4	b3	b2	b1	b0
	RSTIRQS	—	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0 and cannot be modified.	R/W
b7	RSTIRQS	Reset Interrupt Request Selection	0: Error notification to ECM is permitted 1: Error notification to ECM is not performed	R/W

RSTIRQS Bit (Reset Interrupt Request Selection)

This bit specifies whether an error notification should be sent to the error control module (ECM) when an underflow or refresh error occurs in the down-counter.

20.3 Operation

20.3.1 Count Operation in Start Mode

Counting starts by a refresh operation of the WDT refresh register (WDTRR) when the WDT control register (WDTCR) and WDT reset control register (WDTRCR) are set.

20.3.1.1 Register Setting

After the reset state is released, set the clock division ratio, window start and end positions, and timeout period in the WDTCR register, and also set the error notification to the error control module (ECM) in the WDTRCR register. Then, the value specified by the WDTCR.TOPS[1:0] bits is set in the down-counter by a refresh operation to start counting down.
Thereafter, the value in the counter is re-set at each refresh operation and count-down continues if the program runs normally and the counter is refreshed in the refresh-permitted period. The WDT does not output an error notification to ECM as long as the count-down continues. However, if the down-counter underflows because the down-counter cannot be refreshed due to a program runaway, or if a refresh error occurs because the counter is refreshed outside of the refresh-permitted period, the WDT outputs an error notification to the ECM.

Figure 20.3 shows an example of operation under the following conditions.

- Reset interrupt request bit (RSTIRQS): 0b (Error notification to ECM is permitted.)
- Window start position selection bits (RPSS[1:0]): 10b (75%)
- Window end position selection bits (RPES[1:0]): 10b (25%)

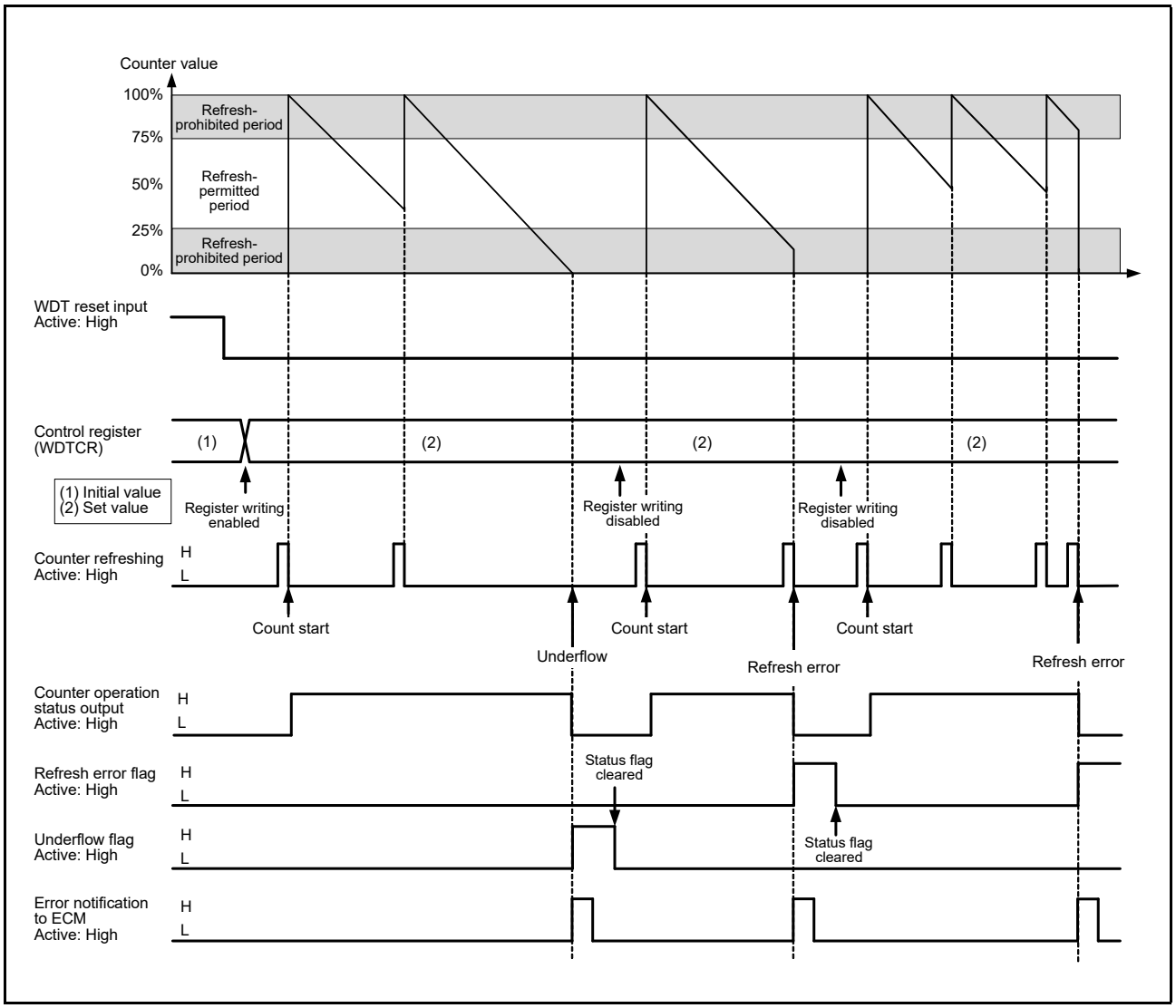


Figure 20.3 Operation Example in Register Start Mode

20.3.2 Control over Writing to the WDTCR and WDTRCR Registers

Writing to the WDT control register (WDTCR) is possible only once between the release from the reset state and the first refresh operation.

After a refresh operation (counting starts) or writing to the WDTCR register, the protection signal in the WDT becomes 1 and WDTCR is protected from subsequent attempts of writing.

Writing to the WDT reset control register (WDTRCR) is also controlled similarly.

This protection is released by the reset source for the WDT. Any other reset sources cannot release the protection.

Figure 20.4 shows control waveforms produced in response to writing to the WDTCR.

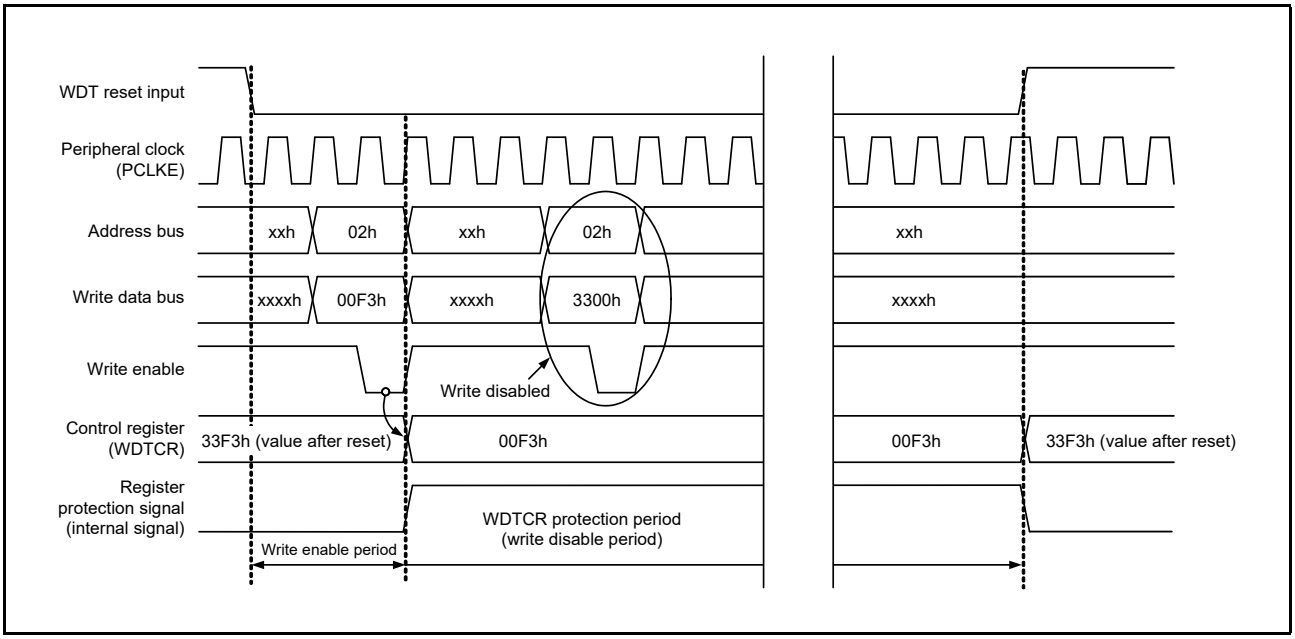


Figure 20.4 Control Waveforms Produced in Response to Writing to the WDTCR Register

20.3.3 Refresh Operation

To refresh the down-counter and to start the down-counter operation (count start due to refreshing), write the values 00h and then FFh to the WDT refresh register (WDTRR). If a value other than FFh is written after 00h, the down-counter is not refreshed. To perform refreshing after such invalid writing, write 00h and then FFh again to the WDTRR register.
If 00h is written twice in succession, writing FFh after the second 00h refreshes the down-counter because the 00h→FFh condition is satisfied. The writing sequence of 00h (n-1th)→00h (nth)→FFh also satisfies the refreshing condition and thus refreshes the down-counter.
Even if a register other than WDTRR is accessed or WDTRR is read between writing 00h and writing FFh to WDTRR, correct refreshing will be done.

[Sample sequences of writing that are valid for refreshing the counter]

- 00h→FFh
- 00h (n–1-th time)→00h (nth time)→FFh
- 00h→access to another register or read from WDTRR→FFh

[Sample sequences of writing that are not valid for refreshing the counter]

- 23h (a value other than 00h)→FFh
- 00h→54h (a value other than FFh)
- 00h→AAh (values other than 00h and FFh)→FFh

If FFh is written to the WDTRR register during the refresh-permitted period after 00h is written outside of the period, write operation is acknowledged and down-counter is refreshed. (Whether writing is made within the refresh-permitted period is determined by when FFh is written.)

After FFh is written to the WDT refresh register (WDTRR), refreshing the down-counter requires up to four counting cycles. (The number of peripheral clock (PCLKE) cycles in a single counting cycle differs depending on the setting of the clock division ratio selection bits (WDTCR.CKS[3:0]).) Therefore, writing FFh to the WDTRR register should be completed within four count-cycles before the refresh-permitted period end position or before the down-counter underflows. Confirm the down-counter value with the down-counter value bits (WDTSR.CNTVAL[13:0]).

[Sample of refresh operation timing]

- When the window start position is 1FFFh, if 00h is written to the WDTRR register before 1FFFh (for example, 2002h), the down-counter can be refreshed by writing FFh to the WDTRR register after the value of WDTSR.CNTVAL[13:0] becomes 1FFFh.
- When the window end position is 1FFFh, if the value of WDTSR.CNTVAL[13:0] is 2003h (four counts before 1FFFh) or more immediately after 00h and FFh are written to the WDTRR register, the down-counter will be refreshed.
- If the refresh-permitted period ranges down to 0000h, refreshing is possible until just before underflow. In this case, if the value of WDTSR.CNTVAL[13:0] is 0003h (four counts before underflow) or more immediately after 00h and FFh are written to the WDTRR register, underflow does not occur and the down-counter is refreshed.

Figure 20.5 shows the WDT refresh-operation waveforms when the clock division ratio = PCLKE/64.

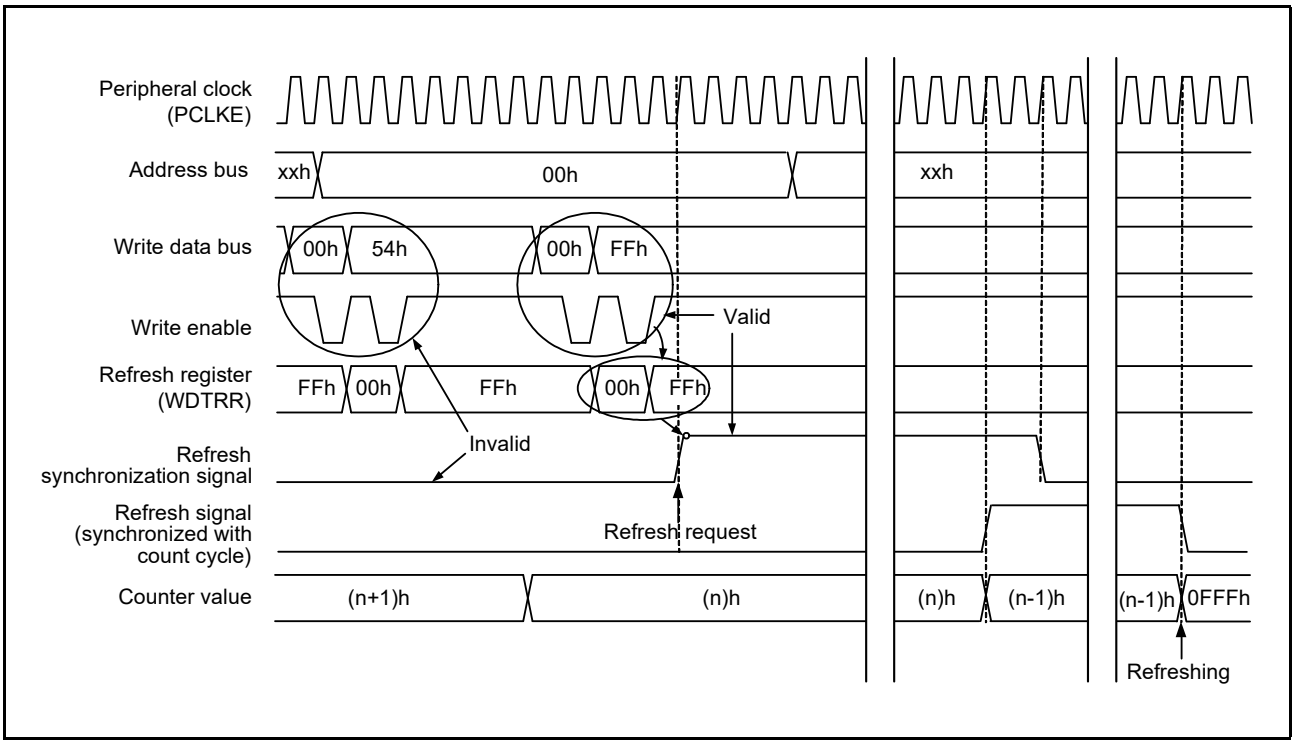


Figure 20.5 WDT Refresh Operation Waveforms (WDTCR.CKS[3:0] = 0100b, WDTCR.TOPS[1:0] = 01b)

20.3.4 Status Flag

The refresh error flag (WDTSR.REFEF) and the underflow flag (WDTSR.UNDFF) retain the error causes after error notifications were output to the error control module (ECM) of WDT.
Occurrence state of error notifications to ECM can be confirmed by reading the WDTSR.REFEF flag or the WDTSR.UNDFF flag after reset is released or upon occurrence of an error notification to ECM.
To clear these flags, write 0. Writing 1 is ignored.
If these flags are not cleared, it will give no effect to the operation. Upon occurrence of the next error notification to ECM, the previous error notification is automatically cleared and a new notification to ECM is written.

20.3.5 Error Notification to the Error Control Module (ECM)

When the reset interrupt selection bit (WDTRCR.RSTIRQS) is set to 0, underflow or refresh error of the down-counter causes an error notification to ECM to be generated during one count cycle.

20.3.6 Reading the Down-Counter Value

The WDT stores the counter value in the down-counter value bits (WDTSR.CNTVAL[13:0]) of the WDT status register. Thus, the counter value can be checked through the WDTSR.CNTVAL[13:0] bits.

Reading requires up to four peripheral clock (PCLKE) cycles. Therefore, the read value may differ from the actual down-counter value by one count.

Figure 20.6 shows the processing for reading the WDT down-counter value when the clock division ratio = PCLKE/64.

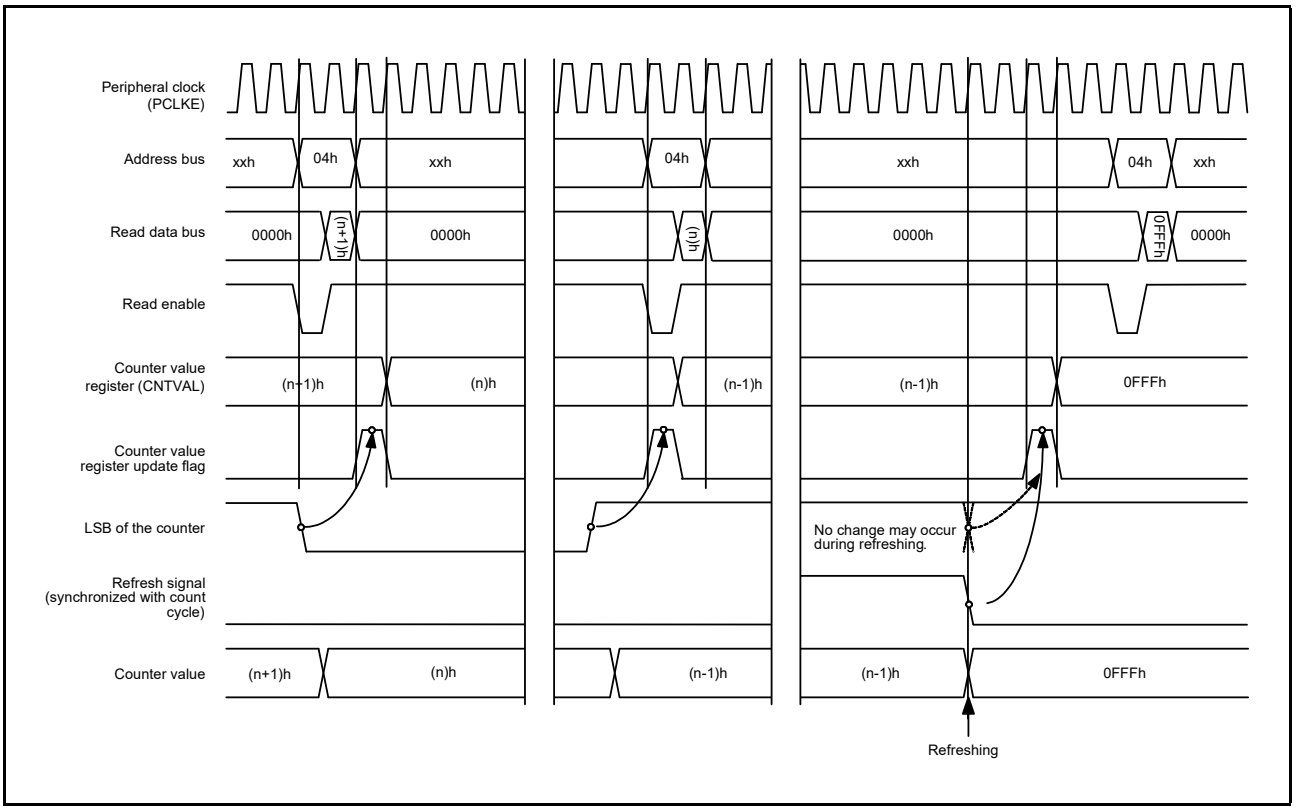


Figure 20.6 Processing for Reading WDT Down-Counter Value (WDTCR.CKS[3:0] = 0100b, WDTCR.TOPS[1:0] = 01b)

20.4 Low-Power Consumption Control

20.4.1 Watchdog Timer Operations in Low-Power Consumption Mode Transition

Clock supply to the WDT can be controlled during transition to the standby mode of Cortex-R4 while the down-counter of WDT is operating.

Table 20.4 lists the WDT operations during transition to the low-power consumption mode.

Table 20.4 WDT Operations during Transition to Low-Power Consumption Mode <in Low-Power Consumption Mode Transition>

Low-Power Consumption Mode	WDT0 Clock Supply	WDT0 Operation
Cortex-R4 standby	√	√

√: Operating

21. Independent Watchdog Timer (IWDTa)

The independent watchdog timer (IWDT) contains a 14-bit down-counter. If the counter underflows, an error notification to the error control module (ECM) is generated. The count value of the down-counter can be refreshed to the value after reset to enable counting to start again. Counter refreshing can be preformed during a period (the refresh-permitted period) you specify. If you perform refresh (register writing) outside of the permitted period, an error notification is generated and sent to the ECM. This enables detection of runaway of the program, taking the refresh interval into consideration. If underflow occurs or if refresh is performed outside of the refresh-permitted period, the IWDT stops counting. Counting restarts after refresh is performed. (For details on refresh operation, see section 21.3.3, Refresh Operation.)

For details on the error control module (ECM), see section 32, Error Control Module (ECM).

21.1 Overview

The IWDT starts counting when refresh (register writing) is performed after reset is released.

Before starting the count, it is necessary to set the clock division ratio, the window start/end positions, and the timeout period in the IWDT control register (IWDTCR).

Table 21.1 lists the specifications of the IWDT.

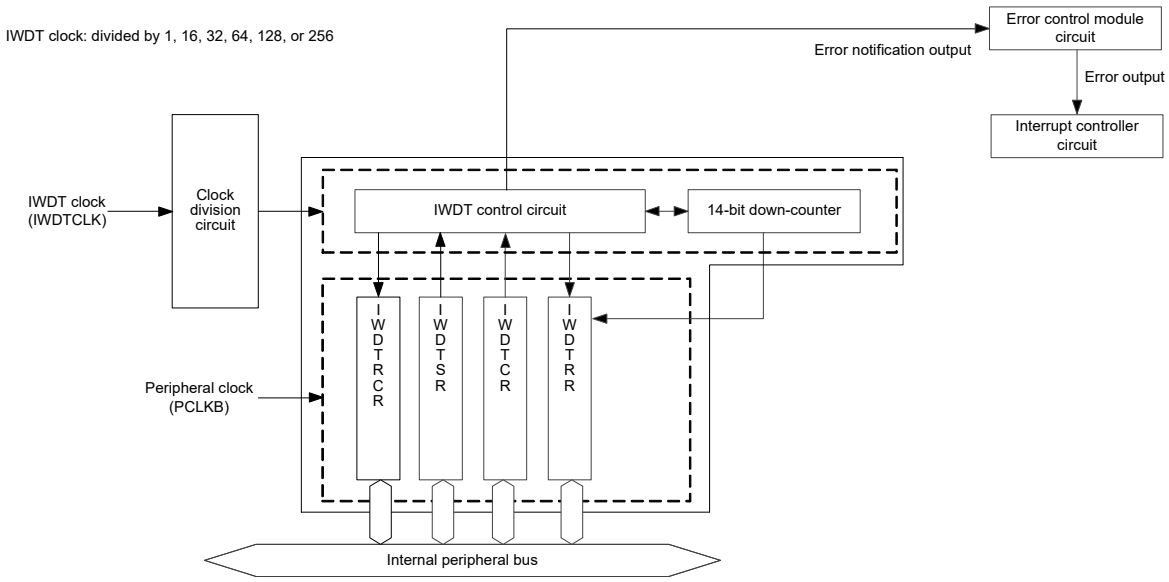
Table 21.1 IWDT Specifications

Item	Description
Count source	IWDT clock (IWDTCLK)
Clock division ratio	Division by 1, 16, 32, 64, 128, or 256
IWDT clock (IWDTCLK) oscillation enable	IWDT clock oscillation starts by a refresh operation.
Counter operation	Counting down using a 14-bit down-counter
Conditions for starting the counter	Counting starts by refreshing the counter (writing 00h and then FFh to the IWDTRR register).
Conditions for stopping the counter	• Reset (the down-counter and other registers return to the values after reset) • A counter underflows or a refresh error is generated
Window function	Window start and end positions can be specified (refresh-permitted period)
Sources for the output of error notification to ECM	• When the down-counter underflows • When refreshing is done outside the refresh-permitted period (refresh error)
Reading the counter value	The down-counter value can be read by reading the IWDTSR register.
IWDT register control	• Selecting the clock frequency division ratio after refreshing (IWDTCR.CKS[3:0] bits) • Selecting the timeout period of the independent watchdog timer (IWDTCR.TOPS[1:0] bits) • Selecting the window start position in the independent watchdog timer (IWDTCR.RPSS[1:0] bits) • Selecting the window end position in the independent watchdog timer (IWDTCR.RPES[1:0] bits)

For operation to continue even if the peripheral clock (PCLKB) stops unexpectedly, counting by the IWDT is driven by two clock signals, the peripheral clock (PCLKB) and the IWDT clock (IWDTCLK). The peripheral clock (PCLKB) provides the timing for the bus interface and registers, and the IWDT clock (IWDTCLK) provides the timing for the 14-bit down-counter and control circuits.

Signals between the peripheral clock (PCLKB) operation blocks and the IWDT clock (IWDTCLK) operation blocks are connected via a synchronization circuit.

Figure 21.1 is a block diagram of the IWDT.



IWDTRR: IWDT refresh register
IWDTCR: IWDT control register
IWDTSR: IWDT status register
IWDTRCR: IWDT reset control register

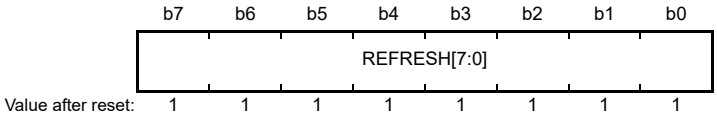
Figure 21.1 IWDT Block Diagram

21.2 Register Descriptions

21.2.1 IWDT Refresh Register (IWDTRR)

The IWDTRR register refreshes the down-counter of the IWDT.
To refresh the down-counter of the IWDT, write 00h and then FFh (refresh operation) to the IWDTRR register in the refresh-permitted period. After being refreshed, the down-counter starts counting down from the value specified with the IWDTCR.TOPS[1:0] bits.
When 00h is written, the read value is 00h. When a value other than 00h is written, the read value is always FFh. For details on the refresh operation, see section 21.3.3, Refresh Operation.

Address(es): A008 0700h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	REFRESH [7:0]	Refresh Register	The down-counter is refreshed by writing 00h and then writing FFh to this register.	R/W

21.2.2 IWDT Control Register (IWDTCR)

The IWDTCR register allows you to select a timeout period before the down-counter underflows, a clock division ratio, and a window start/end position for refreshing.

There are some restrictions on writing to this register. For details, see section 21.3.2, Control Over Writing to the IWDTCR and IWDTRCR Registers.

Address(es): A008 0702h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	RPSS[1:0]		—	—	RPES[1:0]		CKS[3:0]				—	—	TOPS[1:0]	
Value after reset:	0	0	1	1	0	0	1	1	1	1	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TOPS[1:0]	Timeout Period Selection	b1 b0 0 0: 1,024 cycles (03FFh) 0 1: 4,096 cycles (0FFFh) 1 0: 8,192 cycles (1FFFh) 1 1: 16,384 cycles (3FFFh) Values within parentheses are down-count start values.	R/W
b3, b2	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R/W
b7 to b4	CKS[3:0]	Clock Division Ratio Selection	b7 b4 0 0 0 0: IWDTCLK 0 0 1 0: IWDTCLK/16 0 0 1 1: IWDTCLK/32 0 1 0 0: IWDTCLK/64 1 1 1 1: IWDTCLK/128 0 1 0 1: IWDTCLK/256 Other settings are prohibited.	R/W
b9, b8	RPES[1:0]	Window End Position Selection	b9 b8 0 0: 75% 0 1: 50% 1 0: 25% 1 1: 0% (window end position is not specified.)	R/W
b11, b10	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R/W
b13, b12	RPSS[1:0]	Window Start Position Selection	b13 b12 0 0: 25% 0 1: 50% 1 0: 75% 1 1: 100% (window start position is not specified.)	R/W
b15, b14	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R/W

TOPS[1:0] Bits (Timeout Period Selection)

These bits allow you to select a timeout period (a period before the down-counter underflows) from among 1024, 4096, 8192, and 16384 cycles, taking the divided clock specified by the CKS[3:0] bits as one cycle. After the down-counter is refreshed, the combination of the CKS[3:0] and TOPS[1:0] bits determines the time (the number of IWDT clock (IWDTCLK) cycles) before the counter underflows.

Table 21.2 lists relations between the CKS[3:0] and TOPS[1:0] bit settings, the timeout period, and the number of IWDT clock (IWDTCLK) cycles.

CKS[3:0] Bits (Clock Division Ratio Selection)

These bits allow you to select the IWDT clock (IWDTCLK) division ratio from among division by 1, 16, 32, 64, 128, and 256. Combined with the TOPS[1:0] bit setting, a counting period between 1024 and 4194304 cycles of the IWDT clock (IWDTCLK) can be selected for the IWDT. The down-counter value may not be read correctly depending on the relation between the lowest peripheral clock (PCLKB) frequency and the highest IWDT clock (IWDTCLK) frequency.

RPES[1:0] Bits (Window End Position Selection)

These bits allow selection of a window end position from among 75%, 50%, 25%, and 0% of the counting period. The selected window end position should be a value smaller than the window start position (window start position > window end position). If the window end position is greater than the window start position, only the window start position setting is valid.

The counter values of the window start and end positions specified by the RPES[1:0] and RPSS[1:0] bit settings vary according to the TOPS[1:0] bit setting.

Table 21.3 lists the counter values of the window start and end positions corresponding to the TOPS[1:0] bit setting.

RPSS[1:0] Bits (Window Start Position Selection)

These bits allow selection of a window start position of the down-counter from among 100%, 75%, 50%, and 25% of the counting period (assuming that the count start position as 100% and underflow occurrence as 0%). The period from the window start position to the window end position is the refresh-permitted period, and any other periods are the refresh-prohibited periods.

Figure 21.2 shows the relation between the settings of the RPSS[1:0] and RPES[1:0] bits and the refresh-permitted and refresh-prohibited periods.

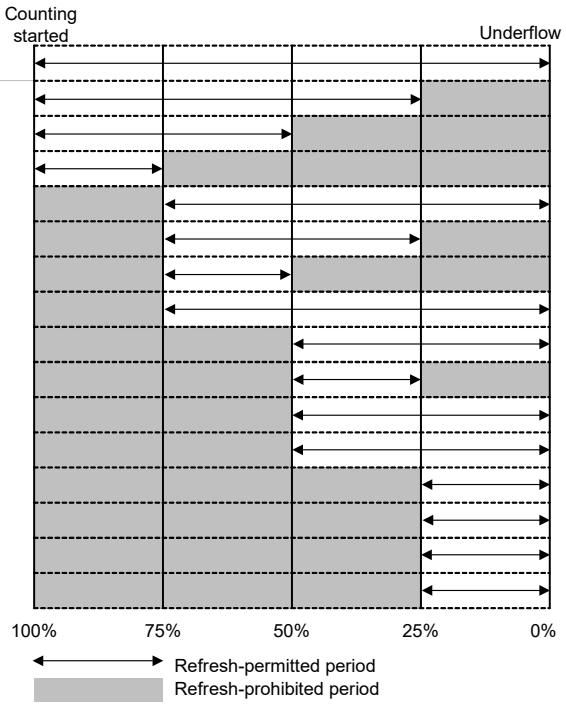
Table 21.2 Settings and Timeout Periods

CKS[3:0] Bits				TOPS[1:0] Bits		Clock Division Ratio	Timeout Period (Number of Cycles)	No. of IWDT Clock (IWDTCLK) Cycles
b7	b6	b5	b4	b1	b0			
0	0	0	0	0	0	IWDTCLK	1024	1024
				0	1		4096	4096
				1	0		8192	8192
				1	1		16384	16384
0	0	1	0	0	0	IWDTCLK/16	1024	16384
				0	1		4096	65536
				1	0		8192	131072
				1	1		16384	262144
0	0	1	1	0	0	IWDTCLK/32	1024	32768
				0	1		4096	131072
				1	0		8192	262144
				1	1		16384	524288
0	1	0	0	0	0	IWDTCLK/64	1024	65536
				0	1		4096	262144
				1	0		8192	524288
				1	1		16384	1048576
1	1	1	1	0	0	IWDTCLK/128	1024	131072
				0	1		4096	524288
				1	0		8192	1048576
				1	1		16384	2097152
0	1	0	1	0	0	IWDTCLK/256	1024	262144
				0	1		4096	1048576
				1	0		8192	2097152
				1	1		16384	4194304

Table 21.3 Relationship between Timeout Period and Window Start and End Counter Values

TOPS[1:0] Bits		Timeout Period		Window Start and End Counter Value			
b1	b0	Cycles	Counter Value	100%	75%	50%	25%
0	0	1024	03FFh	03FFh	02FFh	01FFh	00FFh
0	1	4096	0FFFh	0FFFh	0BFFh	07FFh	03FFh
1	0	8192	1FFFh	1FFFh	17FFh	0FFFh	07FFh
1	1	16384	3FFFh	3FFFh	2FFFh	1FFFh	0FFFh

RPSS[1:0] Bits		RPES[1:0] Bits		Window	
b13	b12	b9	b8	Start (%)	End (%)
1	1	1	1	100	0
		1	0		25
		0	1		50
		0	0		75
1	0	1	1	75	0
		1	0		25
		0	1		50
		0	0		75
0	1	1	1	50	0
		1	0		25
		0	1		50
		0	0		75
0	0	1	1	25	0
		1	0		25
		0	1		50
		0	0		75



Note: If window end setting ≥ window start setting, the window end setting is set to 0%.

Figure 21.2 RPSS[1:0] and RPES[1:0] Bit Settings and the Refresh-Permitted Period

21.2.3 IWDT Status Register (IWDTSR)

The IWDTSR register shows the counter value of the down-counter and whether an underflow or refresh error has occurred.

Address(es): A008 0704h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	REFEF	UNDFF	CNTVAL[13:0]													
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b13 to b0	CNTVAL[13:0]	Counter Value	Value counted by the counter	R
b14	UNDFF	Underflow Flag	0: No underflow occurred 1: Underflow occurred	R/(W)
b15	REFEF	Refresh Error Flag	0: No refresh error occurred 1: Refresh error occurred	R/(W)

CNTVAL[13:0] Bits (Counter Value)

Read these bits to confirm the value of the down-counter, but note that the read value may differ from the actual count by a value of one count.

UNDFF Flag (Underflow Flag)

Read this bit to confirm whether an underflow has occurred in the down-counter.
The value 1 indicates that the down-counter has underflowed. The value 0 indicates that the down-counter has not underflowed.
Write 0 to the UNDFF flag to set the value to 0. Writing 1 has no effect.

REFEF Flag (Refresh Error Flag)

Read this bit to confirm whether a refresh error (performing a refresh operation during a refresh-prohibited period) has occurred.
The value 1 indicates that a refresh error has occurred. The value 0 indicates that no refresh error has occurred.
Write 0 to the REFEF flag to set the value to 0. Writing 1 has no effect.

21.2.4 IWDT Reset Control Register (IWDTRCR)

The IWDTRCR register controls whether to send an error notification to the error control module (ECM) when underflow occurs in the down-counter of IWDT.

There are some restrictions on writing to this register. For details, see section 21.3.2, Control Over Writing to the IWDTCR and IWDTRCR Registers.

Address(es): A008 0706h

	b7	b6	b5	b4	b3	b2	b1	b0
	RSTIRQS	—	—	—	—	—	—	—
Value after reset:	1	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b6 to b0	—	Reserved	These bits are read as 0. Writing to these bits has no effect.	R/W
b7	RSTIRQS	Reset Interrupt Request Selection	0: Error notification to ECM is permitted. 1: Error notification to ECM is not performed.	R/W

RSTIRQS Bit (Reset Interrupt Request Selection)

This bit specifies whether an error notification should be sent to the error control module (ECM) when an underflow or refresh error occurs in the down-counter.

21.3 Operation

21.3.1 Count Operation in Start Mode

Counting starts by a refresh operation of the IWDT refresh register (IWDTRR) if the IWDT control register (IWDTCR) and IWDT reset control register (IWDTRCR) are set.

21.3.1.1 Register Setting

After the reset state is released, set the clock division ratio, window start and end positions, and timeout period in the IWDTCR register, and also set whether to send an error notification to the error control module (ECM) in the IWDTRCR register. Then, the value specified by the timeout period selection bits (IWDTCR.TOPS[1:0]) is set in the down-counter by a refresh operation to start counting down.
Thereafter, the value in the counter is re-set at each refresh operation and count-down continues if the program runs normally and the counter is refreshed in the refresh-permitted period. The IWDT does not output an error notification to ECM as long as the count-down continues.
However, if the down-counter underflows because the down-counter cannot be refreshed due to a program runaway, or if a refresh error occurs because the counter is refreshed outside of the refresh-permitted period, the IWDT outputs an error notification to the ECM.

Figure 21.3 shows an example of operation under the following conditions.

- Reset interrupt request bit (IWDTCR.RSTIRQS): 0b (Error notification to ECM is permitted.)
- Window start position selection bits (IWDTCR.RPSS[1:0]): 10b (75%)
- Window end position selection bits (IWDTCR.RPES[1:0]): 10b (25%)

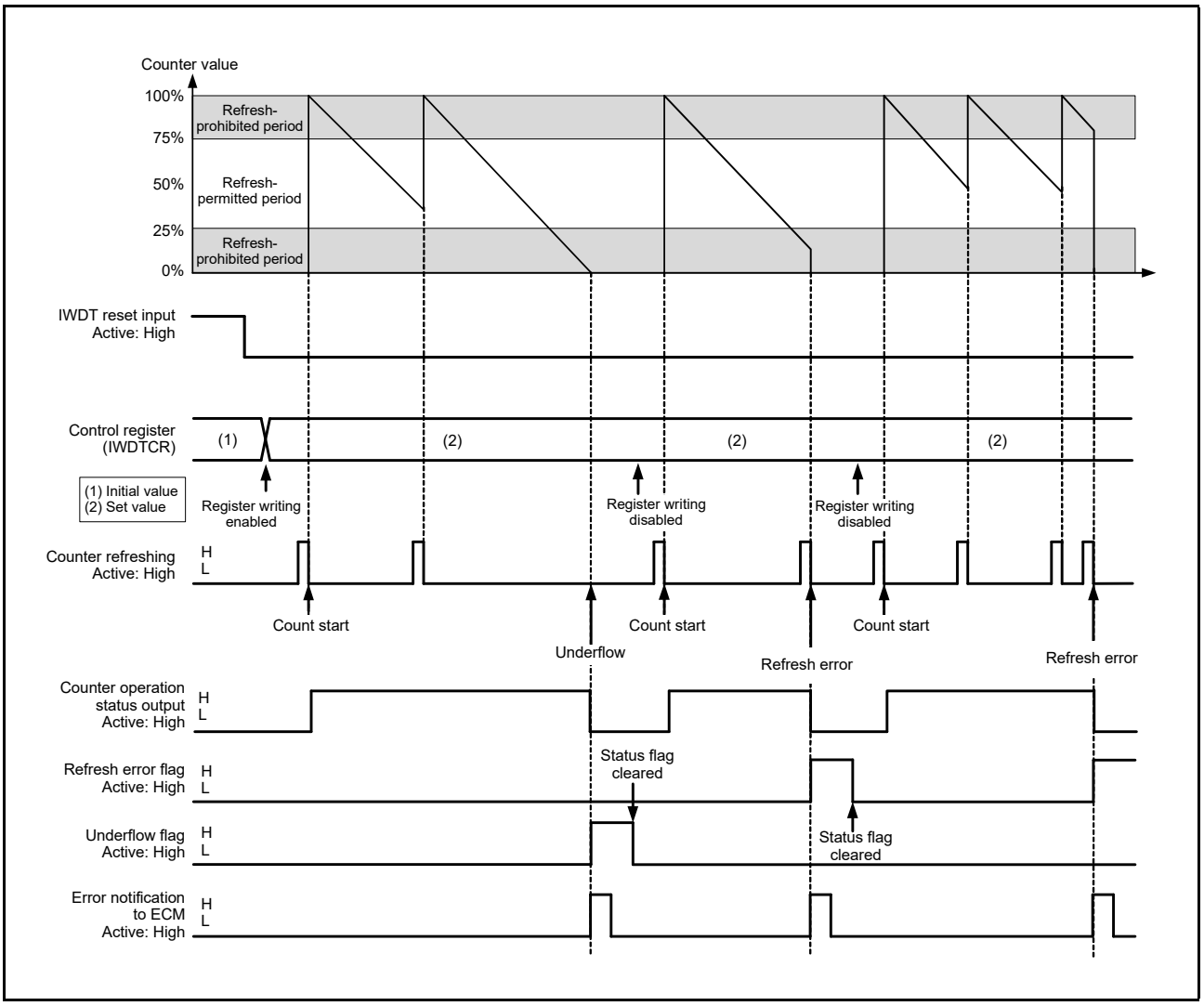


Figure 21.3 Operation Example in Register Start Mode

21.3.2 Control Over Writing to the IWDTCR and IWDTRCR Registers

Writing to the IWDT control register (IWDTCR) is possible only once between the release from the reset state and the first refresh operation.

After a refresh operation (counting starts) or writing to the IWDTCR register, the protection signal in the IWDT becomes 1 to protect IWDTCR from subsequent attempts of writing.

Writing to the IWDT reset control register (IWDTRCR) is also controlled similarly.

This protection is released by the reset source for the IWDT. With other reset sources, the protection is not released.

Figure 21.4 shows control waveforms produced in response to writing to the IWDTCR register.

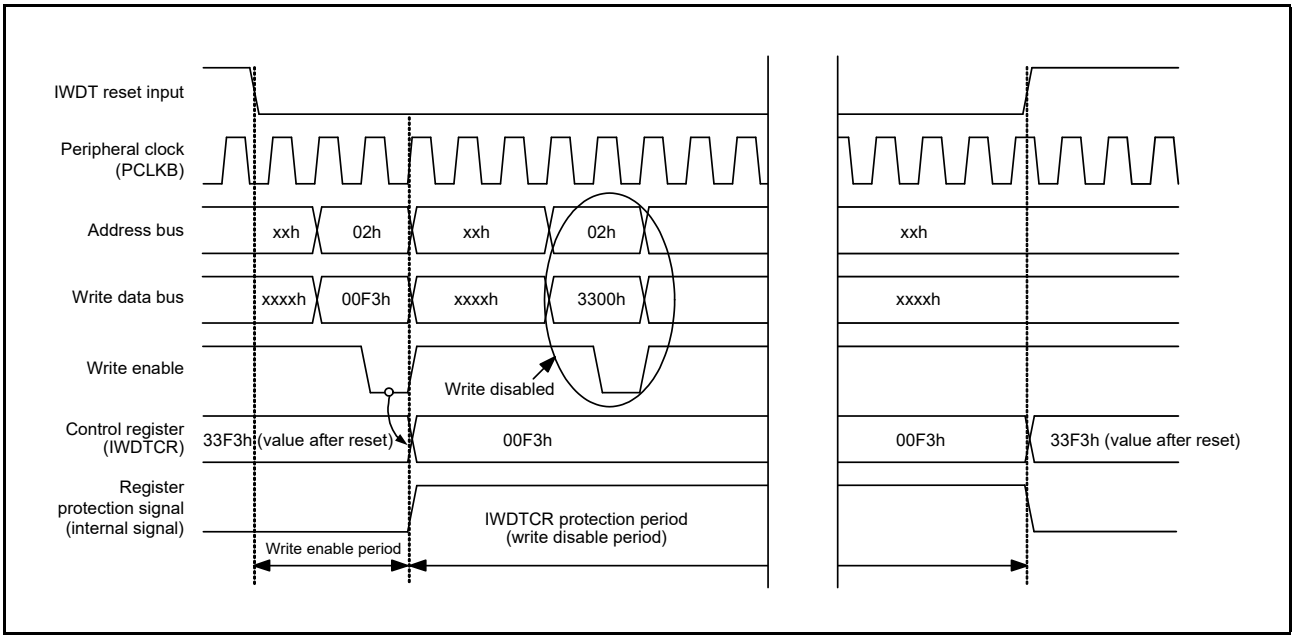


Figure 21.4 Control Waveforms Produced in Response to Writing to the IWDTCR Register

21.3.3 Refresh Operation

To refresh the counter and to start the counter operation (counting is started by refreshing), write the values 00h and then FFh to the IWDT refresh register (IWDTRR). If a value other than FFh is written after 00h, the counter is not refreshed. To perform refreshing after such invalid writing, write 00h and FFh again to the IWDT refresh register (IWDTRR).
When writing is done in the order of 00h (first time) → 00h (second time), and if FFh is written after that, the writing order 00h → FFh is satisfied; writing 00h (n–1-th time) → 00h (nth time) → FFh is valid and correct refreshing will be done. Moreover, even if a register other than IWDTRR is accessed or IWDTRR is read between writing 00h and writing FFh to IWDTRR, correct refreshing will be done.

[Sample sequences of writing that are valid for refreshing the counter]

- 00h → FFh
- 00h (n–1-th time) → 00h (nth time) → FFh
- 00h → access to another register or read from IWDTRR → FFh

[Sample sequences of writing that are not valid for refreshing the counter]

- 23h (a value other than 00h) → FFh
- 00h → 54h (a value other than FFh)
- 00h → AAh (values other than 00h and FFh) → FFh

Even when 00h is written to IWDTRR outside the refresh-permitted period, if FFh is written to IWDTRR in the refresh-permitted period, the writing sequence is valid and refreshing will be done. (Whether writing is made within the refresh-permitted period is determined by when FFh is written.)
After FFh is written to the IWDTRR register, refreshing the counter requires up to four cycles of the signal for counting (the clock division ratio selection bits (IWDTCR.CKS[3:0]) determine how many cycles of the IWDT clock (IWDTCLK) make up one cycle for counting). Therefore, writing FFh to the IWDTRR should be completed four-count cycles before the end position of the refresh-permitted period or a counter underflow. The value of the down-counter can be checked by the down-counter bits (IWDTSR.CNTVAL[13:0]).

[Sample refreshing timings]

- When the window start position is set to 1FFFh, even if 00h is written to IWDTRR before 1FFFh is reached (2002h, for example), refreshing is done if FFh is written to IWDTRR after the value of the IWDTSR.CNTVAL[13:0] bits has reached 1FFFh.
- When the window end position is set to 1FFFh, refreshing is done if 2003h (four-count cycles before 1FFFh) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 00h → FFh to IWDTRR.
- When the refresh-permitted period continues until count 0000h, refreshing can be done immediately before an underflow. In this case, if 0003h (four-count cycles before an underflow) or a greater value is read from the IWDTSR.CNTVAL[13:0] bits immediately after writing 00h → FFh to IWDTRR, no underflow occurs and refreshing is done.

Figure 21.5 shows the IWDT refresh-operation waveforms when PCLKB > IWDTCLK and clock division ratio = IWDTCLK. Figure 21.6 shows the IWDT refresh-operation waveforms when PCLKB < IWDTCLK and clock division ratio = IWDTCLK/16.

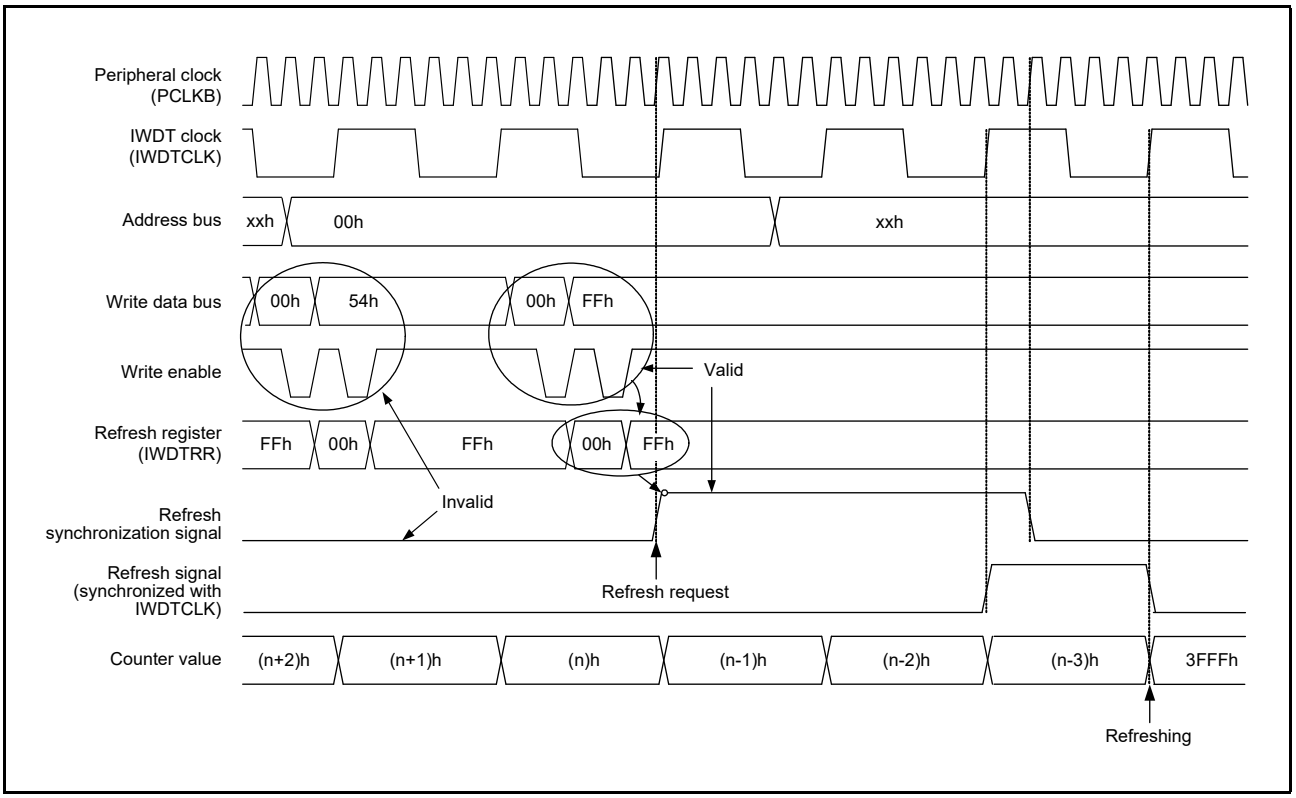


Figure 21.5 IWDT Refresh Operation Waveforms (IWDTCR.CKS[3:0] = 0000b, IWDTCR.TOPS[1:0] = 11b)

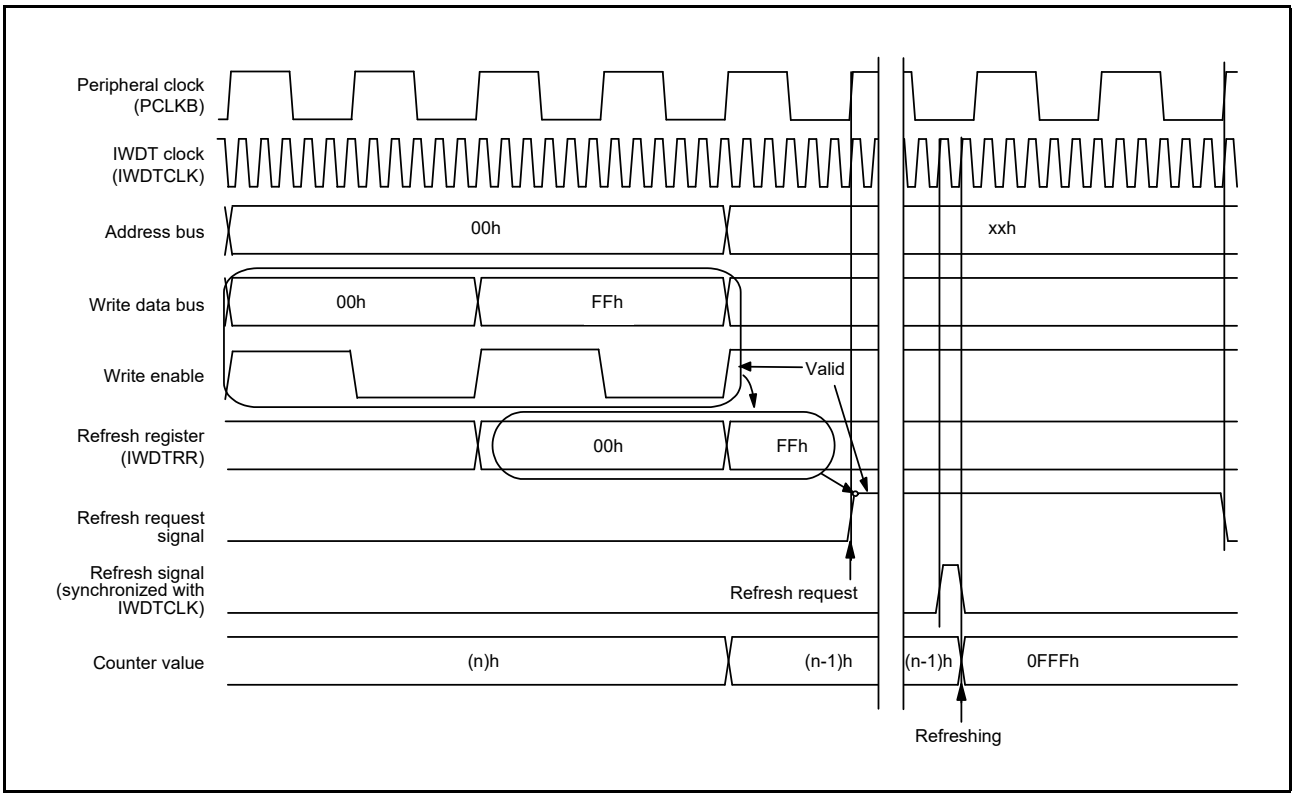


Figure 21.6 IWDT Refresh Operation Waveforms (IWDTCR.CKS[3:0] = 0010b, IWDTCR.TOPS[1:0] = 01b)

21.3.4 Status Flags

The refresh error flag (IWDTSR.REFEF) and the underflow flag (IWDTSR.UNDFF) retain the error causes when error notifications are output to the error control module (ECM) of IWDT.
Occurrence state of error notifications to ECM can be confirmed by reading the IWDTSR.REFEF flag or the IWDTSR.UNDFF flag after reset is released or upon occurrence of an error notification to ECM.
To clear these flags to 0, write 0. Writing 1 is ignored.
If these flags are not cleared, it will give no effect to the operation. Upon occurrence of the next error notification to ECM, the previous error notification is automatically cleared and a new notification to ECM is written.

21.3.5 Error Notification to the Error Control Module (ECM)

When the reset interrupt selection bit (IWDTRCR.RSTIRQS) is set to 0, underflow or refresh error of the down-counter causes an error notification to ECM to be generated during one count cycle.

21.3.6 Reading the Down-Counter Value

As the counter in the IWDT operates with the IWDT clock (IWDTCLK), the IWDT cannot read the counter value directly. Therefore, the IWDT synchronizes the counter value with the peripheral clock (PCLKB) and stores it in the counter value bits (IWDTSR.CNTVAL[13:0]) of the IWDT status register. Thus, the counter value can be checked indirectly through the IWDTSR.CNTVAL[13:0] bits.

Reading the counter value requires multiple PCLKB clock cycles (up to four clock cycles), and the read counter value may differ from the actual counter value by a value of one count.

Figure 21.7 shows the processing for reading the IWDT down-counter value.

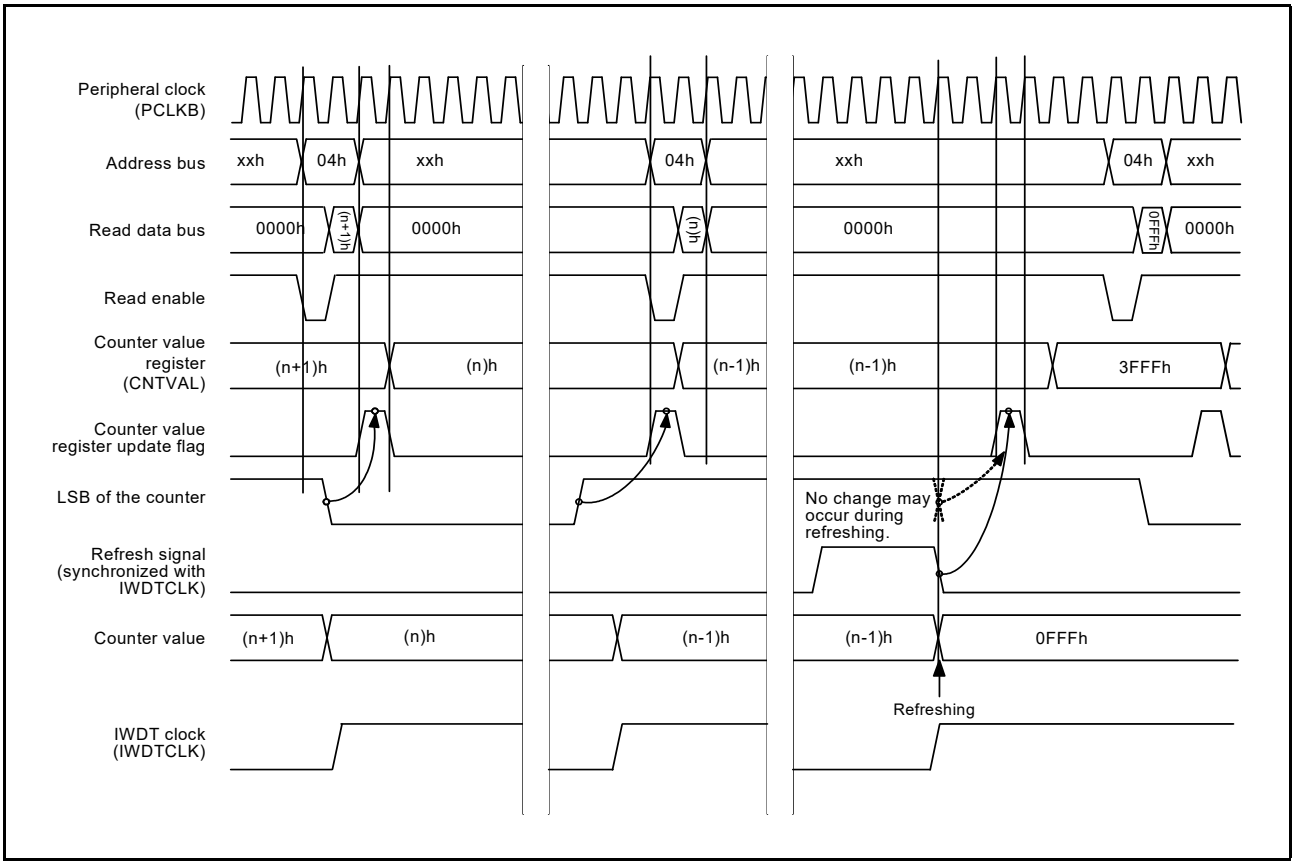


Figure 21.7 Processing for Reading IWDT Counter Value (IWDTCR.CKS[3:0] = 0000b, IWDTCR.TOPS[1:0] = 11b)

21.4 Low-Power Consumption Control

21.4.1 Watchdog Timer Operations in Low-Power Consumption Mode Transition

Clock supply to the IWDT can be controlled during transition to the standby mode of Cortex-R4 while the down-counter of IWDT is operating.

Table 21.4 lists the IWDT operations during transition to the low-power consumption mode.

Table 21.4 IWDT Operations during Transition to Low-Power Consumption Mode <in Low-Power Consumption Mode Transition>

Low-Power Consumption Mode	IWDT Clock Supply	IWDT Operation
Cortex-R4 standby	√	√

√: Operating

22. EtherCAT Slave Controller

22.1 Overview

The EtherCAT slave controller (ESC) uses an EtherCAT Slave Controller IP Core made by Beckhoff Automation GmbH, Germany.
The ESC handles EtherCAT communications as an interface between the EtherCAT field bus and slave applications.

Table 22.1 Specification of the EtherCAT Slave Controller

Item	Description
Number of ports	2
FMMU	8
SyncManager	8
Process data RAM [Kbyte]	8
Distributed clocks	64 bits
EBus	No
Process data interfaces (PDI)	
Digital I/O	No
SPI slave	No
Host MPU interface	On-chip bus

Figure 22.1 is a block diagram of the EtherCAT slave controller.
The management interfaces MDC and MDIO for use in communications with the ESC become available only after the communications with the master device has been established. If configuration of the PHY module through the management interface is required while the ESC has not been activated, use the MDIO controller. However, this controller is not used for the initial settings of the PHY module or the configuration is possible by Strap Option.

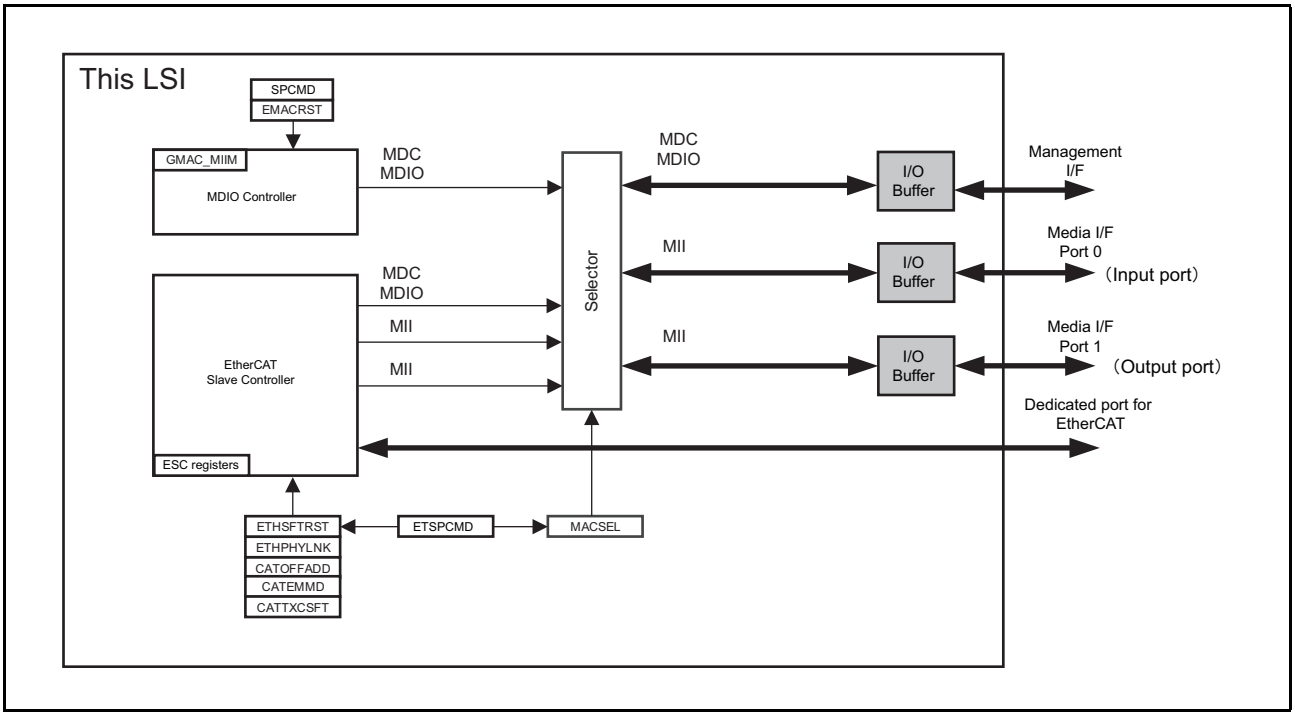


Figure 22.1 Block Diagram of the EtherCAT Slave Controller

Table 22.2 lists the input/output pins of the EtherCAT controller.

Table 22.2 Input/Output Pins of the EtherCAT Slave Controller (excluding PHY MII pins)

Pin Name	I/O	Function	Active
ETH0_TXC, ETH1_TXC	Input	Inputs for the 100M transmission clock signals (25 MHz).	—
ETH0_TXEN, ETH1_TXEN	Output	Outputs for the transmission enable signal.	—
ETH0_TXD0 to 3, ETH1_TXD0 to 3	Output	Outputs for the transmission data signal.	—
ETH0_RXC, ETH1_RXC	Input	Inputs for the reception clock signal.	—
ETH0_RXDV, ETH1_RXDV	Input	Inputs for the reception data enable signal.	—
ETH0_RXER, ETH1_RXER	Input	Inputs for the reception data error signal.	—
ETH0_RXD0 to 3, ETH1_RXD0 to 3	Input	Inputs for the reception data signal.	—
ETH_MDC	Output	Output for the management interface clock signal.	—
ETH_MDIO	Input/Output	Input and output for the management data signal.	—
PHYLINK0, PHYLINK1	Input	Inputs for the PHY link signal.	—
PHYRESETOUT#	Output	Output for the PHY reset signal.	Low
CATLEDRUN	Output	Output for the EtherCAT RUN LED signal.	High
CATIRQ	Output	Output for the EtherCAT IRQ signal.	High
CATLEDSTER	Output	Output for the EtherCAT Dual-color State LED signal.	High
CATLEDERR	Output	Output for the EtherCAT Error LED signal.	High
CATLINKACT0	Output	Outoput for the EtherCAT link / Activity LED signal (port 0).	High
CATLINKACT1	Output	Output for the EtherCAT link / Activity LED signal (port 1).	High
CATSYNC0	Output	Output for the EtherCAT SYNC0 signal.	High
CATSYNC1	Output	Output for the EtherCAT SYNC1 signal.	High
CATLATCH0	Input	Input for the EtherCAT LATCH0 signal.	High
CATLATCH1	Input	Input for the EtherCAT LATCH1 signal.	High
CATI2CCLK	Output	Output for the EtherCAT EEPROM I^2C clock signal.	—
CATI2CDATA	Input/Output	Input and output for the EtherCAT EEPROM I^2C data signal.	—

22.2 Functional Overview

Typical functions of the EtherCAT slave controller and whether they are supported by this LSI or not are shown below. Regarding the detailed specification of the EtherCAT and ESC, refer to the documentation (ETG.1000 EtherCAT Specification, etc.) provided by EtherCAT Technology Group (ETG) and the EtherCAT Slave Controller IP Core (v2.04) data sheet provided by Beckhoff Automation.

Table 22.3 Typical Functions of EtherCAT Slave Controller and Supported Functions by this LSI (1 / 3)

Features	Functions	Support
EtherCAT protocol	Handling the following frames: • Ethernet frames with EtherType 88A4h • EtherCAT frames encapsulated in UDP/IP • EtherCAT frames with VLAN Tag • Normal Ethernet frames	√
Addressing modes	Device addressing • Auto increment address • Configured station address • Broadcast address	√
	Logical addressing	√
Working counter	Counting the number of read/write from/to the device	√
EtherCAT command type	Processing the command that master requests slaves to address each addressing mode	√
Loop control	Loop control and loop state in ESC	√
Shadow buffer	Shadow buffers function when register is read/written	√
Circulating frames	Processing of circulating frames during the failure	√
Link detection	Link MII signal (PHY link signal)	√
	MI Link detection and configuration (monitoring the PHY register via the management interface)	—
	Enhanced link detection (monitoring the state of transfer by MII RX error monitor)	√
FIFO size reduction	RX FIFO size reduction because of reduction of propagation delay	√
Ethernet physical layer	MII	√
	EBUS	—
	Back-to-Back MII connection	√
	MII management interface	√
	Read/write of the PHY register via MII management interface	√
	PHY address offset	√
	Manual TX clock shift compensation	√
	Automatic TX clock shift compensation	√
FMMU	Mapping between logical address and physical address	√
SyncManager	Buffer mode	√
	Mailbox mode	√
	Interrupt and latch event generation when a buffer was completely and successfully written or read.	√
	Repeating mailbox communication	√
	SyncManager deactivation by the PDI	√

Table 22.3 Typical Functions of EtherCAT Slave Controller and Supported Functions by this LSI (2 / 3)

Features	Functions	Support
Distributed clocks	Clock Synchronization considering propagation delay and drift compensation	√
	Generation of synchronous output signals (SYNC0 and 1 signals) • Cyclic mode • Single shot mode • Cyclic acknowledge mode • Single shot acknowledge mode	√
	Precise time stamping of input events (LATCH0 and 1 signals) • Single event mode • Continuous mode • SyncManager event mode (for debugging)	√
	Generation of synchronous interrupts	√
	Synchronous digital output updates / Synchronous digital input sampling	—
	Exclusive control for the SYNC and LATCH signals of the ECAT and PDI	√
	System time control by the PDI	—
	Communication Timing • Free run • Synchronized to output event • Synchronized to SYNC signal	√
EtherCAT state machine	Control of state machine / Indication of the status and error code	√
	Device emulation	—
SII EEPROM	SII EEPROM commands	√
	SII EEPROM error indication	√
	SII EEPROM access interface	√
	EEPROM size selection	√
	EEPROM emulation	—
Interrupt	AL event request (PDI interrupt)	√
	ECAT event request (ECAT interrupt)	√
Watchdog	Process data watchdog	√
	PDI watchdog	√
Error counters	Port error counters	√
	Forwarded RX error counter	√
	ECAT processing unit error counter	√
	PDI error counter	√
	Lost link counter	√
	Watchdog counter process data	√
	Watchdog counter PDI	√
LED signals	RUN LED signal	√
	ERR LED signal	√
	STATE LED and STATE_RUN LED signals	√
	LINK/ACT LED signals	√
	Port error LED signal	—
	RUN/ERR LED override	√

Table 22.3 Typical Functions of EtherCAT Slave Controller and Supported Functions by this LSI (3 / 3)

Features	Functions	Support
Process data interface (PDI)	Digital I/O	—
	SPI	—
	8-bit/16-bit synchronous/asynchronous microcontroller interface	—
	On-chip bus	√
	General purpose I/O	—
Write protection	Write protection for the register area (0000h to 0FFFh)	√
	Write protection for the whole area including the user RAM and process data RAM (0000h to 2FFFh)	√
ESC reset	ESC reset from the master or PDI	√

22.3 Description of Registers

22.3.1 Pheripheral Control Registers

22.3.1.1 System Protect Command Register (SPCMD)

The SPCMD register is used to control writing to write-protected registers. For details, see section 22.4.1, Protect Command Register.

The SPCMD register allows read/write access in 32-bit units.

Address(es): A00F 2100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PROTREL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PROTREL	Protection Unlock Enable	Permits write access to a write-protected register. Writing is allowed in a specific instruction sequence only. 1: Enables write access. 0: Disables write access (write-protected).	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: After data is written to the write-protected register, be sure to clear the SPCMD.PROTREL bit (to 0) to enable protection again.

22.3.1.2 MDIO Controller Reset Register (EMACRST)

The EMACRST register is used to control the reset status of the MDIO controller by software. After the module is released from the reset state, the EMACRST bit is initialized to 0. This means that the MDIO controller is still in the reset state. After the setting of the MAC selection register has been completed, use this register to release the reset state. To reset the MDIO controller during operation, after 0s are written to this register, use software to read the EMACRST bit. Then, after confirming that the bit is set to 0, write 1 to release the reset state.
This register is write-protected by the system protect command register (SPCMD). To perform a write to this register, use the SPCMD register to release write protection.

Address(es): A00F 2110h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EMAC RST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	EMACRST	MDIO Controller Reset Control	Controls the reset state of MDIO controller. 1: Reset-released state 0: Reset state (initial value)	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

22.3.1.3 Ethernet System Protect Command Register (ETSPCMD)

The ETSPCMD register is used to control writing to write-protected registers. For details, see section 22.4.1, Protect Command Register.

The ETSPCMD register allows read/write access in 32-bit units.

Address(es): A00B F000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PROTREL
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PROTREL	Protection Unlock Enable	Permits write access to a write-protected register. Writing is allowed in a specific instruction sequence only. 1: Enables write access. 0: Disables write access (write-protected).	R/W
b31 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note: After data is written to the write-protected register, be sure to clear the ETSPCMD.PROTREL bit (to 0) to enable protection again.

22.3.1.4 MAC Select Register (MACSEL)

The MACSEL register is used to select the function of the Ethernet interface.
Before changing the settings of this register, use the EMACRST register to place the MDIO controller in the reset state and use the ETHSFTRST registers to place the EtherCAT slave controller in the reset state. After changing the register settings, be sure to reset PHY, and then specify the MDIO controller or the Ethercat slave controller settings. This register allows read/write access in 32-bit units. For the configuration of the MAC function, see Figure 22.1 and Table 22.4.
This register is write-protected by the Ethernet system protect command register (ETSPCMD). To perform a write to this register, use the ETSPCMD register to release write protection.

Address(es): A00B F004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	MAC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	MAC[2:0]	Ethernet MAC Mode Select	Selects the function of the Media/Management interface of the MAC to be used. For details, see Table 22.4.	R/W
b31 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Table 22.4 MAC Function Selection Method

MAC[2:0]	Media I/F Port 0	Media I/F Port 1	Management I/F 0
001	EtherCAT Slave Port 0	EtherCAT Slave Port 1	EtherCAT
011	Not used	Not used	MDIO controller
Other than above	Setting prohibited		

22.3.1.5 Ethernet Peripheral Reset Register (ETHSFTRST)

The ETHSFTRST register is used to control the reset states of the EtherCAT slave controller and the PHYRESETOUT# pin by software.

After reset is released, all bits are initialized to 0. Therefore, both the ESC and the PHYRESETOUT# pin are placed in the reset state. After settings of the MAC selection register and EtherCAT peripheral registers (ETHPHYLNK, CATOFFADD, CATEMMD, CATTXCSFT) has been completed, use this register to release the reset state of each.

To reset the ESC or the PHYRESETOUT# pin during operation, after 0s are written to this register, use software to read the target bits. Then, after confirming that the bits are set to 0, write 1 to release the reset state.

This register is write-protected by the Ethernet system protect command register (ETSPCMD). To perform a write to this register, use the ETSPCMD register to release write protection.

Address(es): A00B F118h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PHYRST	—	CATRST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CATRST	EtherCAT Reset Control	Resets EtherCAT slave controller. 1: Reset-released state 0: Reset state (initial value)	R/W
b1	—	Reserved	This is read as 0. The write value should be 0.	R/W
b2	PHYRST	PHYRESETOUT# Pin Reset Control	Resets the PHYRESETOUT# output pin. For products with EtherCAT, the RESETOUT signals are output from the EtherCAT with their functions being logically ORed. 1: Reset-released state 0: Reset state (initial value)	R/W
b31 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

22.3.1.6 Ethernet PHY LINK Mode Register (ETHPHYLNK)

This register is used to specify the active level of the LINK signal of the ethernet interface.
This register is write-protected and can only be written after being protection-unlocked by using the Ethernet system protect command register (ETSPCMD).

Address A00B F014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	CATLINK1	CATLINK0	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	The read value is undefined. When writing, write 0.	R/W
b2	CATLINK0	PHYLINK0 Pin Active Level Switching in Use with EtherCAT	Specify the active level of the PHYLINK0 signal using the EtherCAT interface. 0: Active-High PHYLINK signal. 1: Active-Low PHYLINK signal (value after reset).	R/W
b3	CATLINK1	LINK1 Pin Active Level Switching in Use with EtherCAT	Specify the active level of the PHYLINK1 signal using EtherCAT interface. 0: Active-High PHYLINK signal. 1: Active-Low PHYLINK signal (value after reset).	R/W
b31 to b4	—	Reserved	The read value is undefined. When writing, write 0.	R/W

22.3.1.7 EtherCAT PHY Offset Address Setting Register (CATOFFADD)

This register sets the offset address of the PHY module when the EtherCAT is in use.
This register can be read/written in 32-bit units.
This register can only be written when protection is unlocked by the the Ethernet system protect command register (ETSPCMD).

Address(es): A00B F100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	OADD4	OADD3	OADD2	OADD1	OADD0
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	OADD4 to OADD0	PHY Offset Address Setting	Set the offset address of PHY of the EtherCAT.	R/W
b31 to b5	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.	R/W

22.3.1.8 EtherCAT Operation Mode Setting Register (CATEMMD)

This register sets the EEPROM memory size when the EtherCAT is in use.
This register can be read/written in 32-bit units.
This register can only be written when protection is unlocked by the specific sequence using the Ethernet system protect command register (ETSPCMD).

Address(es): A00B F104h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EEPROMSIZE
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	EEPROMSIZE	EEPROM Memory Size Specification	Sets the size of the EEPROM memory of the EtherCAT. 0: 16 Kbits or less 1: 32 Kbits to 4 Mbits	R/W
b31 to b1	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.	R/W

22.3.1.9 EtherCAT TXC Shift Setting Register (CATTXCSFT)

This register controls the delay time of TXC. This register can be read/written in 32-bit units. This register can only be written when protection is unlocked by the specific sequence using the Ethernet system protect command register (ETSPCMD).

Address(es): A00B F10Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	TXSFT11	TXSFT10	TXSFT01	TXSFT00
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b1, b0	TXSFT01, TXSFT00	ETH0_TXC Delay Time Setting	Set the delay time for ETH0_TXC of the EtherCAT. 00: 0 ns 01: 10 ns 10: 20 ns 11: 30 ns	R/W
b3, b2	TXSFT11, TXSFT10	ETH1_TXC Delay Time Setting	Set the delay time for ETH1_TXC of the EtherCAT. 00: 0 ns 01: 10 ns 10: 20 ns 11: 30 ns	R/W
b31 to b4	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.	R/W

22.3.2 MDIO Controller Registers

22.3.2.1 MIIM Register (GMAC_MIIM)

The GMAC_MIIM register is used to control register access from the MDIO controller to each Ethernet PHY register. For access to this register, follow the procedure below.

For write access:

1. Starting a write operation: Set 1 to the RWDV bit; set the PHY address to the PHYADDR[4:0] bits; set the PHY register address to the REGADDR[4:0] bits; set the write data to the DATA[15:0] bits.
2. Waiting for the operation to end: Wait until 1 is read from the RWDV bit.
3. Confirming the end of the operation: The write operation finishes when 1 is read from the RWDV bit.

For read access:

1. Starting a read operation: Set 0 to the RWDV bit; set the PHY address to the PHYADDR[4:0] bits; set the PHY register address to the REGADDR[4:0] bits.
2. Waiting for the operation to end: Wait until 1 is read from the RWDV bit.
3. Confirming the end of the operation: When 1 is read from the RWDV bit and valid data is read from the DATA[15:0] bits, the read operation finishes.

Address(es): A00F 00A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	RWDV	PHYADDR[4:0]					REGADDR[4:0]				
Value after reset:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DATA[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	DATA[15:0]	Data	Indicates write data or read data.	R/W
b20 to b16	REGADDR [4:0]	PHY Register Address	Specifies the register address of the PHY to be accessed. Because this is a write-only bit, the value read from the bit is undefined.	W
b25 to b21	PHYADDR [4:0]	PHY Address	Specifies the address of the PHY to be accessed. Because this is a write-only bit, the value read from the bit is undefined.	W
b26	RWDV	Read/Write Operation	Writing this bit starts read or write operation, either applicable. Setting of other bits of this register is required when using this function. 1: Starts a write operation. 0: Starts a read operation.	W
			After a read/write operation has started, the status of the operation can be checked by reading the value of this bit.*1 1: Operation is completed. 0: Waiting for the operation to start.	R
b31 to b27	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. After reset is released, the RWDV bit is set to 1. However, the values held by the DATA[15:0] bits at this time are not valid. To use the RWDV bit to check the status correctly, make sure that the value is read from the bit after the operation has started.

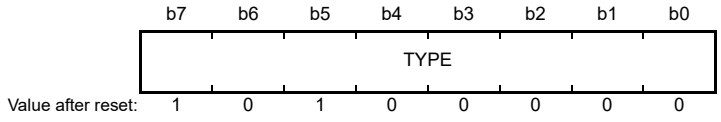
22.3.3 ESC Information Register

Throughout this section, ECAT and PDI indicate access by the EtherCAT master controller and by the local CPU, respectively.

22.3.3.1 Type Register (TYPE)

This register indicates the type of the EtherCAT slave controller.

Address(es): A00D 0000h

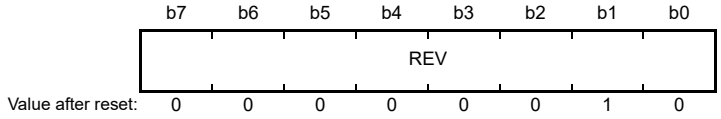


Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	TYPE	Slave Controller Type Indication	Type of the EtherCAT slave controller	R	R

22.3.3.2 Revision Register (REVISION)

This register indicates the revision of the EtherCAT slave controller.

Address(es): A00D 0001h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	REV	Slave Controller Type Indication	Revision of the EtherCAT slave controller	R	R

22.3.3.3 Build Register (BUILD)

This register indicates the build number of the EtherCAT slave controller.

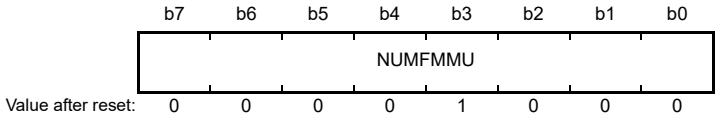
Address(es): A00D 0002h

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	BUILD	Slave Controller Build Number Indication	Build number of the EtherCAT slave controller	R	R

22.3.3.4 FMMU Supported Register (FMMU_NUM)

This register indicates the number of FMMU channels supported in the EtherCAT slave controller.

Address(es): A00D 0004h

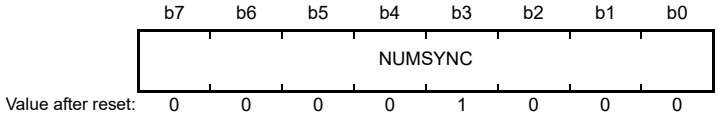


Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	NUMFMMU	FMMU Channel Number Indication	Number of FMMU channels supported in the EtherCAT slave controller.	R	R

22.3.3.5 SyncManager Supported Register (SYNC_MANAGER)

This register indicates the number of SyncManager channels supported in the EtherCAT slave controller.

Address(es): A00D 0005h

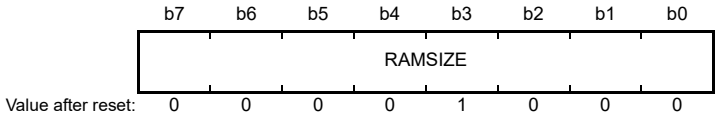


Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	NUMSYNC	SyncManager Channel Number Indication	Number of SyncManager channels supported in the EtherCAT slave controller	R	R

22.3.3.6 RAM Size Register (RAM_SIZE)

This register indicates the process data RAM size supported in the EtherCAT slave controller in Kbyte.

Address(es): A00D 0006h

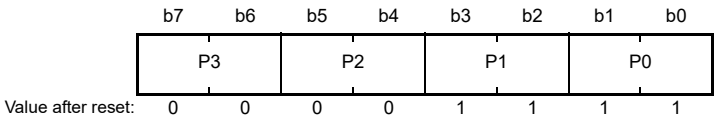


Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	RAMSIZE	Process Data RAM Size Indication	Process data RAM size supported in the EtherCAT slave controller (unit: Kbyte)	R	R

22.3.3.7 Port Descriptor Register (PORT_DESC)

This register indicates the port configuration.

Address(es): A00D 0007h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b1, b0	P0	Port 0 Configuration	Port 0 configuration: Fixed to the setting for MII connection (11) in this LSI. 00: Not implemented 01: Not configured (SII EEPROM) 10: EBUS 11: MII	R	R
b3, b2	P1	Port 1 Configuration	Port 1 configuration: Fixed to the setting for MII connection (11) in this LSI. 00: Not implemented 01: Not configured (SII EEPROM) 10: EBUS 11: MII	R	R
b5, b4	P2	Port 2 Configuration	Port 2 configuration: This LSI does not implement port 2. Fixed to (00). 00: Not implemented 01: Not configured (SII EEPROM) 10: EBUS 11: MII	R	R
b7, b6	P3	Port 3 Configuration	Port 3 configuration: This LSI does not implement port 3. Fixed to (00). 00: Not implemented 01: Not configured (SII EEPROM) 10: EBUS 11: MII	R	R

22.3.3.8 ESC Features Supported Register (FEATURE)

This register indicates the features supported in the EtherCAT slave controller.

Address(es): A00D 0008h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	FSCONFIG	RWSUPP	LRW	DCSYNC	FCS	LINKDECMII	—	—	DCWID	DC	—	FMMU
Value after reset:	x	x	x	x	0	0	0	1	1	1	x	x	1	1	x	0

x: Undefined

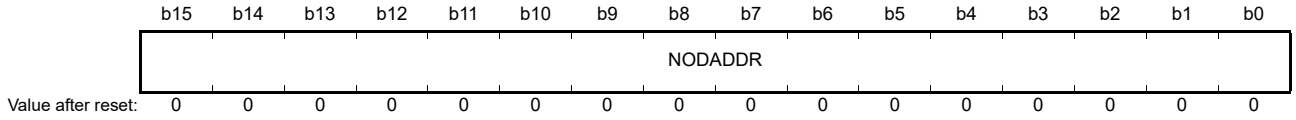
Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	FMMU	FMMU Operation Specification	FMMU operation 0: Bit oriented 1: Byte oriented	R	R
b1	—	Reserved	When read, the value returned is undefined.	R	R
b2	DC	Distributed Clock Specification	Distributed clocks 0: Not available 1: Available	R	R
b3	DCWID	Distributed Clock Width Specification	Distributed clocks (width) 0: 32 bits 1: 64 bits	R	R
b5, b4	—	Reserved	When read, the value returned is undefined.	R	R
b6	LINKDECMII	Enhanced Link Detection Specification	Enhanced link detection in MII 0: Not available 1: Available	R	R
b7	FCS	FCS Error Specification	Separate handling of FCS errors 0: Not supported 1: Supported. Frames with wrong FCS and additional nibble will be counted separately in forwarded RX error counter.	R	R
b8	DCSYNC	DC SYNC Specification	Enhanced DC SYNC activation 0: Not available 1: Available	R	R
b9	LRW	LRW Command Support Specification	EtherCAT LRW command support 0: Supported 1: Not supported	R	R
b10	RWSUPP	Command Support Specification	EtherCAT read/write command support (BRW, APRW, FPRW) 0: Supported 1: Not supported	R	R
b11	FSCONFIG	FMMU/SyncManager Specification	Fixed FMMU/SyncManager configuration 0: Variable configuration 1: Fixed configuration	R	R
b15 to b12	—	Reserved	When read, the value returned is undefined.	R	R

22.3.4 Station Address Registers

22.3.4.1 Configured Station Address Register (STATION_ADR)

This register indicates the address used for node addressing.

Address(es): A00D 0010h

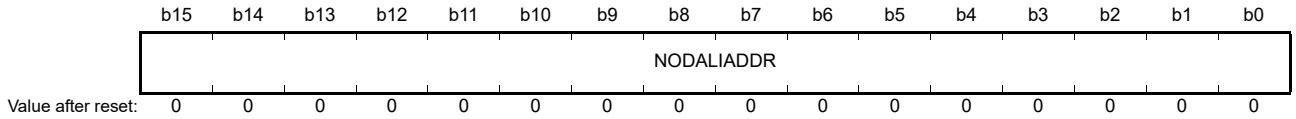


Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	NODADDR	Node Addressing Address Indication	Address used for node addressing (FPxx commands)	R	R/W

22.3.4.2 Configured Station Alias Register (STATION_ALIAS)

This register indicates the alias address used for node addressing (FPxx commands).

Address(es): A00D 0012h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	NODALIADDR	Alias Address Indication	Alias address used for node addressing (FPxx commands). The use of this alias is activated by setting bit 8 of the extended ESC DL control register (ESC_EX_DL_CONTROL at 0102h) to 1.	R/W	R

Note: The initial value, 0, is retained until the EEPROM is loaded. After that, the value becomes the value at address 0004h in the EEPROM. This value is only taken over from the EEPROM the first time the EEPROM is loaded after a power-on or reset.

22.3.5 Write Protection Registers

22.3.5.1 Write Register Enable Register (WR_REG_ENABLE)

This register is used to unlock the write protection temporarily while registers are write-protected.

Address(es): A00D 0020h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ENABLE
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	ENABLE	Register Write Protection Unlock	When registers are currently being protected against writing (bit 0 is 1 in the write register protection register, WR_REG_PROTECT, at 0021h) and freely writing to registers of the given node is to be permitted, the operation to do so by writing to this register has to proceed in the same Ethernet frame and preceding the other desired writing to registers. Write protection will be reactivated once the frame period elapses (unless the value in the write register protection register is changed).	R	R/W
b7 to b1	—	Reserved	When read, the value returned is undefined.	R	R

22.3.5.2 Write Register Protection Register (WR_REG_PROTECT)

This register is used to protect registers against writing. The registers in the area A00D 0000h to A00D 0FFFh are write-protected (except for the WR_REG_ENABLE register (0020h) and ESC_WR_ENABLE register (0030h)).

Address(es): A00D 0021h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	PROTECT
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	PROTECT	Register Write Protection Specification	Protection of registers against writing 0: Protection disabled 1: Protection enabled	R	R/W
b7 to b1	—	Reserved	When read, the value returned is undefined.	R	R

22.3.5.3 ESC Write Enable Register (ESC_WR_ENABLE)

This register is used to unlock the write protection temporarily while registers and memories are write-protected by ESC write protection.

Address(es): A00D 0030h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ENABLE
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	ENABLE	Register/Memory Write Protection Unlock	When registers are currently being protected against writing by ESC write protection (bit 0 is 1 in the ESC write protection register, ESC_WR_PROTECT, at 0031h) and freely writing to registers of the given node is to be permitted, the operation to do so by writing to this register has to proceed in the same Ethernet frame and preceding the other desired writing to registers. Write protection will be reactivated once the frame period elapses (unless the value in the ESC write protection register is changed).	R	R/W
b7 to b1	—	Reserved	When read, the value returned is undefined.	R	R

22.3.5.4 ESC Write Protection Register (ESC_WR_PROTECT)

This register is used to protect registers against writing. Registers and memories in the area A00D 0000h to A00D 2FFFh including the process data RAM are write-protected (except for the WR_REG_ENABLE register (0020h) and ESC_WR_ENABLE register (0030h)).

Address(es): A00D 0031h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	PROTECT
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	PROTECT	Register/Memory Write Protection Specification	Protection of registers and process memories against writing 0: Protection disabled 1: Protection enabled	R	R/W
b7 to b1	—	Reserved	When read, the value returned is undefined.	R	R

22.3.6 Data Link Layer Registers

22.3.6.1 ESC Reset ECAT Register (ESC_RESET_ECAT)

This register is used to reset the EtherCAT slave controller from the ECAT (master) by software.

When written: ESC_RESET_ECAT_W

Address(es): A00D 0040h

	b7	b6	b5	b4	b3	b2	b1	b0
	RESET_ECAT							
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	RESET_ECAT	Software Reset Setting	A reset is enabled after writing 52h (“R”), 45h (“E”) and 53h (“S”) consecutively to this register.	R	R/W

When read: ESC_RESET_ECAT_R

Address(es): A00D 0040h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	RESET_ECAT	
Value after reset:	x	x	x	x	x	x	0	0

x: Undefined

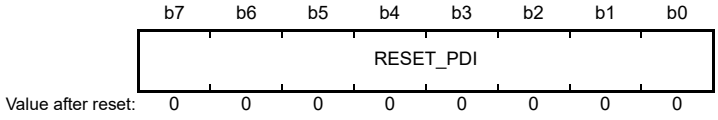
Bit	Symbol	Bit Name	Description	PDI	ECAT
b1, b0	RESET_ECAT	Reset Progress Status	Progress of the reset procedure 01: After writing 52h 10: After writing 45h (if 52h was written before) 00: Others	R	R/W
b7 to b2	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.	R	R/W

22.3.6.2 ESC Reset PDI Register (ESC_RESET_PDI)

This register is used to reset the EtherCAT slave controller from the PDI (slave) by software.

When written: ESC_RESET_PDI_W

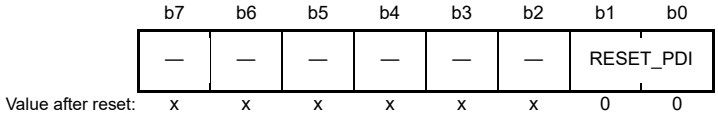
Address(es): A00D 0041h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	RESET_PDI	Software Reset Setting	A reset is enabled after writing 52h (“R”), 45h (“E”) and 53h (“S”) consecutively to this register.	R/W	R

When read: ESC_RESET_PDI_R

Address(es): A00D 0041h



x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b1, b0	RESET_PDI	Reset Progress Status	Progress of the reset procedure 01: After writing 52h 10: After writing 45h (following the writing of 52h) 00: Others	R/W	R
b7 to b2	—	Reserved	When read, the value returned is undefined.	R	R

22.3.6.3 ESC DL Control Register (ESC_DL_CONTROL)

This register is used to control loop in the EtherCAT slave controller and configure the RX FIFO size and station alias.

Address(es): A00D 0100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	STAALIAS	—	—	—	—	—	RXFIFO		
Value after reset:	x	x	x	x	x	x	x	0	x	x	x	x	x	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LP3		LP2		LP1		LP0		—	—	—	—	—	—	TEMPUSE	FWDRULE
Value after reset:	1	1	0	0	0	0	0	0	x	x	x	x	x	x	0	1

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	FWDRULE	Forwarding Rule Specification	Forwarding rule 0: EtherCAT frames are processed. Non-EtherCAT frames are forwarded without processing. 1: EtherCAT frames are processed. Non-EtherCAT frames are destroyed. The source MAC address is changed for every frame (SOURCE_MAC[1] is set to 1 (locally administered address)) regardless of the forwarding rule.	R	R/W
b1	TEMPUSE	Temporary Use of Bits 15 to 8	Temporary use of bits 15 to 8 settings 0: Permanent use 1: Use for about 1 second, then revert to previous settings	R	R/W
b7 to b2	—	Reserved	When read, the value returned is undefined.	R	R
b9, b8	LP0	Loop Port 0 Configuration	Loop port 0 configuration 00: Auto 01: Auto close 10: Open 11: Closed	R	R/W
b11, b10	LP1	Loop Port 1 Configuration	Loop port 1 configuration 00: Auto 01: Auto close 10: Open 11: Closed	R	R/W
b13, b12	LP2	Loop Port 2 Configuration	Loop port 2 configuration (port 2 is not available on this LSI.) 00: Auto 01: Auto close 10: Open 11: Closed	R	R/W
b15, b14	LP3	Loop Port 3 Configuration	Loop port 3 configuration (port 3 is not available on this LSI.) 00: Auto 01: Auto close 10: Open 11: Closed	R	R/W
b18 to b16	RXFIFO	RX FIFO Size Specification	Set the RX FIFO size. The transfer time can be reduced by reducing the FIFO size. 0 to 3: Reduction of 40 ns 4 to 6: No change 7: Default	R	R/W
b23 to b19	—	Reserved	When read, the value returned is undefined.	R	R

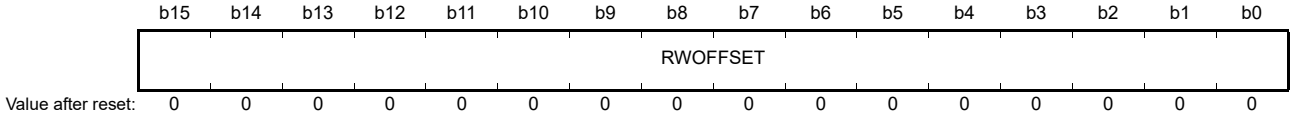
Bit	Symbol	Bit Name	Description	PDI	ECAT
b24	STAALIAS	Station Alias Status Specification	Station alias 0: Ignore station alias 1: Alias can be used for all configured address command types (FPRD, FPWR, ...).	R	R/W
b31 to b25	—	Reserved	When read, the value returned is undefined.	R	R

Note: Changes to loop configurations are delayed until any current reception or transmission of a frame through the port is completed.
Note: Reducing the size of the Rx FIFO depends on all masters and slaves connected to the same network as the EtherCAT having very precise clock sources. An Rx FIFO size of 7 (default) is sufficient if the precision of all clocks is 100 ppm or better and Rx FIFO sizes of 0 to 3 are possible if the precision is 25 ppm or better.

22.3.6.4 Physical Read/Write Offset Register (PHYSICAL_RW_OFFSET)

This register is used to set the offset between read address and write address in the R/W commands.

Address(es): A00D 0108h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	RWOFFSET	Offset between Read and Write Addresses	Offset of R/W commands (FPRW, APRW) between read address and write address. That is, in the case of reading, RD_ADR = ADR (the given address is read) In the case of writing, WR_ADR = ADR + R/W offset (writing is to the address obtained by adding the offset set in this register to the given address)	R	R/W

22.3.6.5 ESC DL Status Register (ESC_DL_STATUS)

This register indicates the state of the EtherCAT slave controller.

Address(es): A00D 0110h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	COMP3	LP3	COMP2	LP2	COMP1	LP1	COMP0	LP0	PHYP3	PHYP2	PHYP1	PHYP0	—	ENHLINKD	PDIWDST	PDIOPE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	x	1	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	PDIOPE	PDI/EEPROM Load State Indication	PDI operation/EEPROM load state 0: EEPROM not loaded, the PDI not operational (process data RAM is not accessible) 1: EEPROM loaded correctly, the PDI operational (process data RAM is accessible)	R	R (ack)
b1	PDIWDST	Watchdog Status	PDI watchdog timer status 0: Timeout of the watchdog timer 1: Watchdog timer reloaded	R	R (ack)
b2	ENHLINKD	Enhanced Link Detection Indication	Enhanced link detection 0: Deactivated for all ports 1: Activated for at least one port Note: This bit is set to the value of bit 9 at address 0000h in the EEPROM the first time the EEPROM is loaded after power is initially supplied or after a reset.	R	R (ack)
b3	—	Reserved	When read, the value returned is undefined.	R	R (ack)
b4	PHYP0	Port 0 Link State Indication	Physical link on port 0 0: No link 1: Link detected	R	R (ack)
b5	PHYP1	Port 1 Link State Indication	Physical link on port 1 0: No link 1: Link detected	R	R (ack)
b6	PHYP2	Port 2 Link State Indication	Physical link on port 2 (port 2 is not available on this LSI.) 0: No link 1: Link detected	R	R (ack)
b7	PHYP3	Port 3 Link State Indication	Physical link on port 3 (port 3 is not available on this LSI.) 0: No link 1: Link detected	R	R (ack)
b8	LP0	Loop Port 0 State Indication	Loop port 0 0: Open 1: Closed	R	R (ack)
b9	COMP0	Port 0 Communication State Indication	Communication on port 0 0: No stable communication 1: Communication established	R	R (ack)
b10	LP1	Loop Port 1 State Indication	Loop port 1 0: Open 1: Closed	R	R (ack)
b11	COMP1	Port 1 Communication State Indication	Communication on port 1 0: No stable communication 1: Communication established	R	R (ack)
b12	LP2	Loop Port 2 State Indication	Loop port 2 (port 2 is not available on this LSI.) 0: Open 1: Closed	R	R (ack)
b13	COMP2	Port 2 Communication State Indication	Communication on port 2 (port 2 is not available on this LSI.) 0: No stable communication 1: Communication established	R	R (ack)

Bit	Symbol	Bit Name	Description	PDI	ECAT
b14	LP3	Loop Port 3 State Indication	Loop port 3 (port 3 is not available on this LSI.) 0: Open 1: Closed	R	R (ack)
b15	COMP3	Port 3 Communication State Indication	Communication on port 3 (port 3 is not available on this LSI.) 0: No stable communication 1: Communication established	R	R (ack)

Note: Reading this register from the ECAT clears bit 2 of the ECAT event request register (ECAT_EVENT_REQ at 0210h).

22.3.7 Application Layer Registers

22.3.7.1 AL Control Register (AL_CONTROL)

This register is used to change the state transition of the device state machine and to acknowledge error indication.

Address(es): A00D 0120h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	DEVICEID	ERRINDACK	INISTATE			
Value after reset:	x	x	x	x	x	x	x	x	x	x	0	0	0	0	0	1

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b3 to b0	INISTATE	State Machine State Change Specification	Change the state transition of the device state machine. 1: Initial state request 3: Bootstrap state request 2: Pre-operational state request 4: Safe-operational state request 8: Operational state request	R (clear)	R/(W)
b4	ERRINDACK	Error Indication Acknowledge	Error indication acknowledge (response) 0: Error Indication in AL status register is not acknowledged. 1: Error Indication in AL status register is acknowledged.	R (clear)	R/(W)
b5	DEVICEID	Device ID Request	Device ID request 0: No request is present. 1: A request is present.	R (clear)	R/(W)
b15 to b6	—	Reserved	When read, the value returned is undefined.	R (clear)	R/(W)

Note: The PDI has to read the AL control register after the ECAT has written it. Otherwise the ECAT cannot write again to the AL control register. Reading the AL control register from the PDI clears bit 0 of the AL event request register (AL_EVENT_REQ at 0220h).

22.3.7.2 AL Status Register (AL_STATUS)

This register indicates the state of slave application.

Address(es): A00D 0130h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	DEVICEID	ERR	ACTSTATE			
Value after reset:	x	x	x	x	x	x	x	x	x	x	0	0	0	0	0	1

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b3 to b0	ACTSTATE	State Machine State Indication	Actual state of the device state machine 1: Initial state 3: Request bootstrap state 2: Pre-operational state 4: Safe-operational state 8: Operational state	R/(W)	R (ack)
b4	ERR	Error State Indication	Error indicator 0: The device is in the state as requested or flag was cleared by command. 1: The device has not entered the requested state or the state was changed as a result of local action.	R/(W)	R (ack)
b5	DEVICEID	Device ID Load State Indication	Device ID load state 0: Loading the device ID failed. 1: The device ID was loaded.	R/(W)	R (ack)
b15 to b6	—	Reserved	When read, the value returned is undefined.	R/(W)	R (ack)

Note: Reading this register from the ECAT clears bit 3 of the ECAT event request register (ECAT_EVENT_REQ at 0210h).

22.3.7.3 AL Status Code Register (AL_STATUS_CODE)

This register indicates an error code from slave application.

Address(es): A00D 0134h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	STATUSCODE															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	STATUSCODE	Error Code Indication	AL status code	R/W	R

22.3.7.4 RUN LED Override Register (RUN_LED_OVERRIDE)

This register is used to override control of the RUN LED pin.

Address(es): A00D 0138h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	OVERRIDEEN	LEDCODE			
Value after reset:	x	x	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W	PDI	ECAT
b3 to b0	LEDCODE	LED Code Indication	LED code 0h: Off 1h to Ch: Flash 1x to 12x Dh: Blinking Eh: Flickering Fh: On	(FSM state) (1-Init) (4-SafeOp 1x) (2-PreOp) (3-Bootstrap) (8-Op) Note: Bits 3 to 0 of the AL status register (AL_STATUS)	R/W	R/W
b4	OVERRIDEEN	Override Setting	Override enable 0: Override disabled 1: Override enabled		R/W	R/W
b7 to b5	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.		R/W	R/W

Note: Changing the value of the AL status register to an appropriate value will clear bit 4 (override enable).
Normally RUN LED is controlled by the AL status register (AL_STATUS at 0130h) automatically. It is not necessary to override RUN LED in order to indicate the state of a general state machine.
For example, this register can be used to run special lighting patterns that indicate the positions of specific slaves.

22.3.7.5 ERR LED Override Register (ERR_LED_OVERRIDE)

This register is used to override control of the error LED pin.

Address(es): A00D 0139h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	OVERRIDEEN	LEDCODE			
Value after reset:	x	x	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b3 to b0	LEDCODE	LED Code Indication	LED code 0h: Off 1h to Ch: Flash 1x to 12x Dh: Blinking Eh: Flickering Fh: On	R/W	R/W
b4	OVERRIDEEN	Override Setting	Override enable 0: Override disabled 1: Override enabled	R/W	R/W
b7 to b5	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.	R/W	R/W

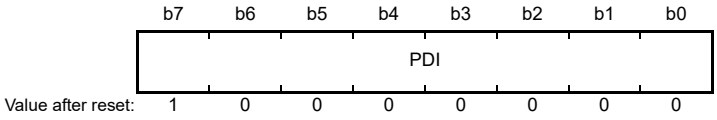
Note: Bit 4 (override enable) will be cleared if a new error occurs.
The ESC automatically controls an error LED under the conditions below. Regarding other errors, the error LED should be controlled by application using this register.
- SII EEPROM load error
- PDI watchdog timeout

22.3.8 PDI Registers

22.3.8.1 PDI Control Register (PDI_CONTROL)

This register indicates the type of PDI.

Address(es): A00D 0140h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	PDI	PDI Type Indication	Process data interface. In this chip, the below value is indicated. 80h: On-chip bus	R	R

22.3.8.2 ESC Configuration Register (ESC_CONFIG)

This register indicates configuration of the EtherCAT slave controller.

Address(es): A00D 0141h

	b7	b6	b5	b4	b3	b2	b1	b0
	ENLP3	ENLP2	ENLP1	ENLP0	DCLATCH	DCSYNC	ENLALLP	DEVEMU
Value after reset:	0	0	0	0	1	1	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	DEVEMU	Device Emulation Setting	Device emulation (control of AL status) 0: The AL status register has to be set by the PDI. 1: The AL status register will be set to a value written to the AL control register.	R	R
b1	ENLALLP	All Ports Enhanced Link Detection Setting	Sets enhanced link detection for all ports. 0: Disabled (if bits 15 to 12 of address 0 in the EEPROM = 0) 1: Enabled at all ports	R	R
b2	DCSYNC	SYNC Output Unit State Indication	Sets the SYNC output unit for distributed clocks (fixed to 1 in this LSI). 0: Disabled (power saving) 1: Enabled	R	R
b3	DCLATCH	Latch Input Unit Setting	Sets the latch input unit for distributed clocks (fixed to 1 in this LSI). 0: Disabled (power saving) 1: Enabled	R	R
b4	ENLP0	Port 0 Enhanced Link Detection Setting	Sets enhanced link detection for port 0. 0: Disabled (if bit 9 of address 0 in the EEPROM = 0) 1: Enabled	R	R
b5	ENLP1	Port 1 Enhanced Link Detection Setting	Sets enhanced link detection for port 1. 0: Disabled (if bit 9 of address 0 in the EEPROM = 0) 1: Enabled	R	R
b6	ENLP2	Port 2 Enhanced Link Detection Setting	Sets enhanced link detection for port 2 (port 2 is not available on this LSI). 0: Disabled (if bit 9 of address 0 in the EEPROM = 0) 1: Enabled	R	R
b7	ENLP3	Port 3 Enhanced Link Detection Setting	Sets enhanced link detection for port 3 (port 3 is not available on this LSI). 0: Disabled (if bit 9 of address 0 in the EEPROM = 0) 1: Enabled	R	R

22.3.8.3 PDI Configuration Register (PDI_CONFIG)

This register indicates configuration of the PDI.

Address(es): A00D 0150h

	b7	b6	b5	b4	b3	b2	b1	b0
	ONCHIPBUS			ONCHIPBUSCLK				
Value after reset:	0	1	0	0	0	1	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b4 to b0	ONCHIPBUSCLK	On-Chip Bus Clock Indication	Indicate the frequency of the on-chip bus clock. In this chip, the value is always 4 (corresponding to 100 MHz).	R	R
b7 to b5	ONCHIPBUS	On-Chip Bus Type Indication	Indicate the type of on-chip bus. In this chip, the value is always 010.	R	R

22.3.8.4 SYNC/LATCH PDI Configuration Register (SYNC_LATCH_CONFIG)

This register indicates the configuration of SYNC output and LATCH input.

Address(es): A00D 0151h

	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC1 MAP	SYNCL AT1	SYNC1OUT		SYNC0 MAP	SYNCL AT0	SYNC0OUT	
Value after reset:	1	1	1	0	1	1	1	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b1, b0	SYNC0OUT	SYNC0 Polarity Indication	Indicate the SYNC0 output driver/polarity. In this chip, the value is always 10 (push-pull active high).	R	R
b2	SYNCLAT0	SYNC0/LATCH0 Indication	Indicates the SYNC0/LATCH0 configuration. In this chip, the value is always 1*1. 0: LATCH0 input 1: SYNC0 output	R	R
b3	SYNC0MAP	SYNC0 State Mapping Indication	Indicates enabling or disabling of mapping of the SYNC0 state to bit 2 of the AL event request register (AL_EVENT_REQ at 0220h). This is always enabled in this chip, so the value indicated is always 1 (enabled). 0: Disabled 1: Enabled	R	R
b5, b4	SYNC1OUT	SYNC1 Polarity Indication	Indicate the SYNC1 output driver/polarity. In this chip, the value is always 10 (push-pull active high).	R	R
b6	SYNCLAT1	SYNC1/LATCH1 Indication	Indicates the SYNC1/LATCH1 configuration. In this chip, the value is always 1*1. 0: LATCH1 input 1: SYNC1 output	R	R
b7	SYNC1MAP	SYNC1 State Mapping Indication	Indicates enabling or disabling of mapping of the SYNC1 state to bit 3 of the AL event request register (AL_EVENT_REQ at 0220h). This is always enabled in this chip, so the value indicated is always 1 (enabled). 0: Disabled 1: Enabled	R	R

Note 1. Latch input is available though the value indicates SYNC output. Use the MPC function in order to switch SYNC output to LATCH input and vice versa.

22.3.8.5 Extended PDI Configuration Register (EXT_PDI_CONFIG)

This register indicates configuration of the PDI.

Address(es): A00D 0152h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DATABUSWID	
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b1, b0	DATABUSWID	PDI Data Bus Width Indication	Indicate the data bus width of the PDI. In this chip, the value is always 0 (4 bytes). 00: 4 bytes 01: 1 byte 10: 2 bytes 11: Reserved	R	R
b15 to b2	—	Reserved	When read, the value returned is undefined.	R	R

22.3.9 Interrupt Registers

22.3.9.1 ECAT Event Mask Register (ECAT_EVENT_MASK)

The ECAT event request (ECAT interrupt) is used to transmit the slave event to the EtherCAT master. This register is used to set mask to each event of the ECAT event request register (ECAT_EVENT_REQ at 0210h). The logical AND of each effective bit in the ECAT event request register and the corresponding bit of this register is taken and the result produces the interrupt signal.

Address(es): A00D 0200h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECATEVMASK															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	ECATEVMASK	Event Request Mask Setting	0: The corresponding bit of the ECAT event request register (ECAT_EVENT_REQ at 0210h) is not mapped. 1: The corresponding bit of the ECAT event request register is mapped.	R	R/W

22.3.9.2 AL Event Mask Register (AL_EVENT_MASK)

The AL event request (PDI interrupt) is used to transmit the ESC interrupt to the slave application. This register is used to set mask to each event of the AL event request register (AL_EVENT_REQ at 0220h). The logical AND of each effective bit in the AL event request register and the corresponding bit of this register is taken and the result produces the interrupt signal.

Address(es): A00D 0204h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ALEVMASK															
Value after reset:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ALEVMASK															
Value after reset:	1	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	ALEVMASK	Event Request Mask Setting	0: The corresponding bit of the AL event request register (AL_EVENT_REQ at 0220h) is not mapped. 1: The corresponding bit of the AL event request register is mapped.	R/W	R

22.3.9.3 ECAT Event Request Register (ECAT_EVENT_REQ)

This register indicates the source of ECAT event requests (ECAT interrupts).

Address(es): A00D 0210h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	SMSTA7	SMSTA6	SMSTA5	SMSTA4	SMSTA3	SMSTA2	SMSTA1	SMSTA0	ALSTA	DLSTA	—	DCLATCH
Value after reset:	x	x	x	x	0	0	0	0	0	0	0	0	0	0	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	DCLATCH	DC Latch Event State Indication	DC latch event 0: No change on DC latch Inputs 1: At least one change on DC latch Inputs This bit is cleared by reading DC latch event times from the ECAT for ECAT controlled latch units, so the latch 0 and 1 status registers (DC_LATCH_STAT0 and DC_LATCH_STAT1 at 09AEh and 09AFh) indicate no event.	R	R
b1	—	Reserved	When read, the value returned is undefined.	R	R
b2	DLSTA	DL Status Event State Indication	DL status event 0: No change in DL status 1: DL status change This bit is cleared by reading out the DL status register (ESC_DL_STATUS at 0110h or 0111h) from the ECAT.	R	R
b3	ALSTA	AL Status Event State Indication	AL status event 0: No change in AL status 1: AL status change This bit is cleared by reading out the AL status register (AL_STATUS at 0130h or 0131h) from the ECAT.	R	R
b4	SMSTA0	SyncManager 0 Status Indication	Mirror value of SyncManager 0 status 0: No Sync channel 0 event 1: Sync channel 0 event pending	R	R
b5	SMSTA1	SyncManager 1 Status Indication	Mirror value of SyncManager 1 status 0: No Sync channel 1 event 1: Sync channel 1 event pending	R	R
b6	SMSTA2	SyncManager 2 Status Indication	Mirror value of SyncManager 2 status 0: No Sync channel 2 event 1: Sync channel 2 event pending	R	R
b7	SMSTA3	SyncManager 3 Status Indication	Mirror value of SyncManager 3 status 0: No Sync channel 3 event 1: Sync channel 3 event pending	R	R
b8	SMSTA4	SyncManager 4 Status Indication	Mirror value of SyncManager 4 status 0: No Sync channel 4 event 1: Sync channel 4 event pending	R	R
b9	SMSTA5	SyncManager 5 Status Indication	Mirror value of SyncManager 5 status 0: No Sync channel 5 event 1: Sync channel 5 event pending	R	R
b10	SMSTA6	SyncManager 6 Status Indication	Mirror value of SyncManager 6 status 0: No Sync channel 6 event 1: Sync channel 6 event pending	R	R
b11	SMSTA7	SyncManager 7 Status Indication	Mirror value of SyncManager 7 status 0: No Sync channel 7 event 1: Sync channel 7 event pending	R	R
b15 to b12	—	Reserved	When read, the value returned is undefined.	R	R

22.3.9.4 AL Event Request Register (AL_EVENT_REQ)

This register indicates the source of AL event requests (PDI interrupts).

Address(es): A00D 0220h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SMINT 7	SMINT 6	SMINT 5	SMINT 4	SMINT 3	SMINT 2	SMINT 1	SMINT 0	—	WDPD	—	SYNCA CT	DCSYN C1STA	DCSYN C0STA	DCLAT CH	ALCTR L
Value after reset:	0	0	0	0	0	0	0	0	x	0	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	ALCTRL	AL Control Event State Indication	AL control event 0: No change in the AL control register 1: The AL control register has been written. This bit is cleared by reading the AL control register (AL_CONTROL at 0120h or 0121h) from the PDI.	R	R
b1	DCLATCH	DC Latch Event State Indication	DC latch event 0: No change on DC latch Inputs 1: At least one change on DC latch inputs This bit is cleared by reading DC latch event times from the PDI for PDI controlled latch units, so the latch 0 and 1 status registers (DC_LATCH_STAT0 and DC_LATCH_STAT1 at 09AEh and 09AFh) indicate no event.	R	R
b2	DCSYNC0STA	DC SYNC0 State Indication	State of DC SYNC0 This bit is cleared by reading the SYNC0 status register (DC_SYNC0_STAT at 098Eh) from the PDI.	R	R
b3	DCSYNC1STA	DC SYNC1 State Indication	State of DC SYNC1 This bit is cleared by reading the SYNC1 status register (DC_SYNC1_STAT at 098Fh) from the PDI.	R	R
b4	SYNCACT	SyncManager Activation Indication	Change of the SyncManager activation register (SMm.ACT at 0806h + 8h*m) 0: No change in any SyncManager 1: At least one SyncManager changed This bit is cleared by reading SyncManager activation registers from the PDI.	R	R
b5	—	Reserved	When read, the value returned is undefined.	R	R
b6	WDPD	Watchdog Process Data Indication	Watchdog process data 0: Valid 1: Timeout This bit is cleared by reading the watchdog status process data register (WDS_DATA at 0440h) from the PDI.	R	R
b7	—	Reserved	When read, the value returned is undefined.	R	R
b8	SMINT0	SyncManager 0 Interrupt Status	SyncManager 0 interrupt (bit 0 or 1 of the SyncManager status register (0805h)) 0: No SyncManager 0 interrupt 1: SyncManager 0 interrupt pending	R	R
b9	SMINT1	SyncManager 1 Interrupt Status	SyncManager 1 interrupt (bit 0 or 1 of the SyncManager status register (080Dh)) 0: No SyncManager 1 interrupt 1: SyncManager 1 interrupt pending	R	R
b10	SMINT2	SyncManager 2 Interrupt Status	SyncManager 2 interrupt (bit 0 or 1 of the SyncManager status register (0815h)) 0: No SyncManager 2 interrupt 1: SyncManager 2 interrupt pending	R	R

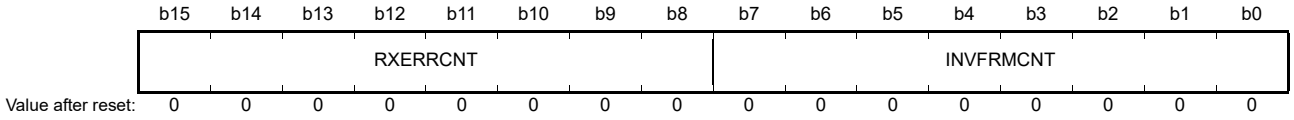
Bit	Symbol	Bit Name	Description	PDI	ECAT
b11	SMINT3	SyncManager 3 Interrupt Status	SyncManager 3 interrupt (bit 0 or 1 of the SyncManager status register (081Dh)) 0: No SyncManager 3 interrupt 1: SyncManager 3 interrupt pending	R	R
b12	SMINT4	SyncManager 4 Interrupt Status	SyncManager 4 interrupt (bit 0 or 1 of the SyncManager status register (0825h)) 0: No SyncManager 4 interrupt 1: SyncManager 4 interrupt pending	R	R
b13	SMINT5	SyncManager 5 Interrupt Status	SyncManager 5 interrupt (bit 0 or 1 of the SyncManager status register (082Dh)) 0: No SyncManager 5 interrupt 1: SyncManager 5 interrupt pending	R	R
b14	SMINT6	SyncManager 6 Interrupt Status	SyncManager 6 interrupt (bit 0 or 1 of the SyncManager status register (0835h)) 0: No SyncManager 6 interrupt 1: SyncManager 6 interrupt pending	R	R
b15	SMINT7	SyncManager 7 Interrupt Status	SyncManager 7 interrupt (bit 0 or 1 of the SyncManager status register (083Dh)) 0: No SyncManager 7 interrupt 1: SyncManager 7 interrupt pending	R	R
b31 to b16	—	Reserved	When read, the value returned is undefined.	R	R

22.3.10 Error Counter Registers

22.3.10.1 Rx Error Counter n Register (RX_ERR_COUNTn)

This register counts RX frame errors.

Address(es): A00D 0300h + 0002h*n



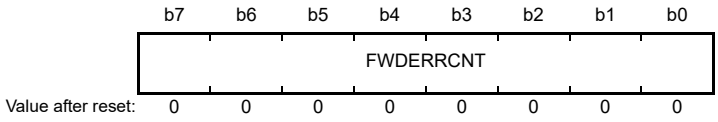
Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	INVFRMCNT	Invalid Frame Counter Value Indication	Counter value of invalid frames for port n Counting is stopped when FFh is reached. These bits are cleared if one of the RX error counters (RX_ERR_COUNTn, FWD_RX_ERR_COUNTn) is written.	R	R/W (clr)
b15 to b8	RXERRCNT	RX Frame Error Counter Value Indication	Counter value of RX errors for port n Counting is stopped when FFh is reached. The number of RX errors of MII interface is counted. These bits are cleared if one of the RX error counters (RX_ERR_COUNTn, FWD_RX_ERR_COUNTn) is written.	R	R/W (clr)

n = 0, 1
n = 0 for port 0; n = 1 for port 1

22.3.10.2 Forwarded Rx Error Counter n Register (FWD_RX_ERR_COUNTn)

This register counts forwarded RX frame errors.

Address(es): A00D 0308h + 0001h*n



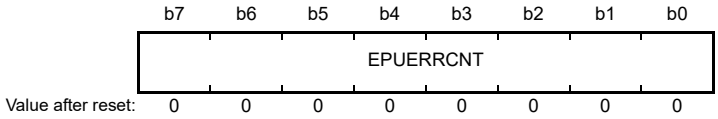
Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	FWDERRCNT	Forwarded Error Counter Value Indication	Counter value of forwarded RX error frames for port n Counting is stopped when FFh is reached. These bits are cleared if one of the RX error counters (RX_ERR_COUNTn, FWD_RX_ERR_COUNTn) is written.	R	R/W (clr)

n = 0, 1
n = 0 for port 0; n = 1 for port 1

22.3.10.3 ECAT Processing Unit Error Counter Register (ECAT_PROC_ERR_COUNT)

This register counts frame errors passing the ECAT processing unit.

Address(es): A00D 030Ch

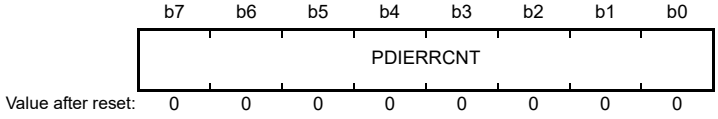


Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	EPUERRCNT	Processing Unit Error Counter Value Indication	ECAT processing unit error counter value Counting is stopped when FFh is reached. This register counts errors of frames passing the processing unit. Writing to this register clears it.	R	R/W (clr)

22.3.10.4 PDI Error Counter Register (PDI_ERR_COUNT)

This register counts PDI access errors.

Address(es): A00D 030Dh

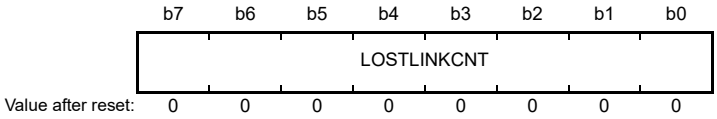


Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	PDIERRCNT	PDI Error Counter Value Indication	PDI error counter value Counting is stopped when FFh is reached. Counting starts when an interface error occurs due to access to the PDI. Writing to this register clears it.	R	R/W (clr)

22.3.10.5 Lost Link Counter n Register (LOST_LINK_COUNTn)

This register counts lost links at port n.

Address(es): A00D 0310h + 0001h*n



Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	LOSTLINKCNT	Lost Link Counter Value Indication	Lost link counter value for port n Counting is stopped when FFh is reached. Counting starts only when port loop is Auto or Auto-Close. Only lost links at open ports are counted. Writing to one of the lost link counter registers clears it.	R	R/W (clr)

n = 0, 1

n = 0 for port 0; n = 1 for port 1

22.3.11 Watchdog Registers

22.3.11.1 Watchdog Divider Register (WD_DIVIDE)

This register is used to set the ratio for dividing 25 MHz to obtain the basic period for incrementing the watchdog timer.

Address(es): A00D 0400h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	WDDIV															
Value after reset:	0	0	0	0	1	0	0	1	1	1	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	WDDIV	Watchdog Clock Frequency Divisor Setting	Set the frequency divisor of the clock to drive counting by the watchdog timer in units of ticks at 25 MHz. The clock that drives counting by the watchdog timer is obtained by dividing 25 MHz by the value in this register plus two. The default value is 2498, which corresponds to a period of 100 µs.	R	R/W

22.3.11.2 Watchdog Time PDI Register (WDT_PDI)

This register is used to set the time until the PDI watchdog timer overflows.

Address(es): A00D 0410h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	WDTIMPDI															
Value after reset:	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	WDTIMPDI	Watchdog Overflow Time Setting	Set the time until the PDI watchdog timer overflows as a number of times the watchdog is incremented. With the default values for these bits and the setting of the watchdog divider, the time for a single incrementation is 100 µs, so the watchdog timer overflows when 100 µs × 1000 = 100 ms elapse. Setting these bits to 0 disables the watchdog timer. Access to the PDI restarts the watchdog timer.	R	R/W

22.3.11.3 Watchdog Time Process Data Register (WDT_DATA)

This register is used to set the time until the process data watchdog timer overflows.

Address(es): A00D 0420h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	WDTIMPD															
Value after reset:	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	WDTIMPD	Watchdog Overflow Time Setting	Set the time until the process data watchdog timer overflows as a number of times the watchdog is incremented. With the default values for these bits and the setting of the watchdog divider, the time for a single incrementation is 100 µs, so the watchdog timer overflows when 100 µs × 1000 = 100 ms elapse. There is one Watchdog for all SyncManagers. Setting these bits to 0 disables the watchdog timer. Access to the watchdog trigger enable bit of SyncManager restarts the watchdog timer.	R	R/W

22.3.11.4 Watchdog Status Process Data Register (WDS_DATA)

This register indicates the state of the process data watchdog timer.

Address(es): A00D 0440h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WDSTAPD
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0

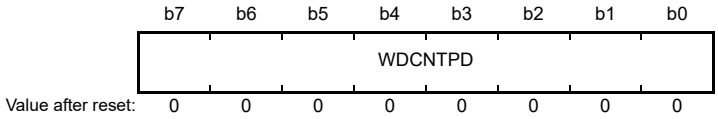
x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	WDSTAPD	Watchdog State Indication	Indicates the state of the process data watchdog timer triggered by SyncManagers. 0: The timeout period of the process data watchdog timer elapses. 1: The process data watchdog timer is active or disabled Reading this register clears bit 6 of the AL event request register (AL_EVENT_REQ at 0220h).	R (ack)	R
b15 to b1	—	Reserved	When read, the value returned is undefined.	R (ack)	R

22.3.11.5 Watchdog Counter Process Data Register (WDC_DATA)

This register indicates the timeout counter value of the process data watchdog timer.

Address(es): A00D 0442h

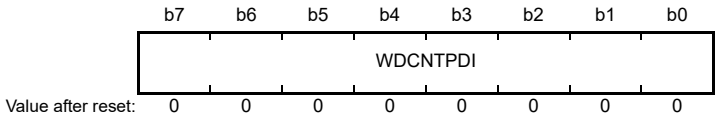


Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	WDCNTPD	Watchdog Counter Value Indication	Counter value of the process data watchdog timer Counting stops when FFh is reached. Counting starts on a timeout of the process data watchdog timer. Writing to one of the watchdog counter registers (WDC_DATA, WDC_PDI at 0442h and 0443h) clears the counter.	R	R/W (clr)

22.3.11.6 Watchdog Counter PDI Register (WDC_PDI)

This register indicates the timeout counter value of the PDI watchdog timer.

Address(es): A00D 0443h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b7 to b0	WDCNTPDI	Watchdog Counter Value Indication	Counter value of the PDI watchdog timer Counting stops when FFh is reached. Counting starts on a timeout of the PDI watchdog timer. Writing to one of the watchdog counter registers (WDC_DATA, WDC_PDI at 0442h and 0443h) clears the counter.	R	R/W (clr)

22.3.12 SII EEPROM Interface Registers

The EtherCAT controls the SII EEPROM interface if bit 0 is 0 in the EEPROM configuration register (EEP_CONF at 0500h) and in the EEPROM PDI access state register (EEP_PDI_ACCESS at 0501h). Otherwise, the PDI controls the EEPROM interface.

22.3.12.1 EEPROM Configuration Register (EEP_CONF)

This register is used to configure EEPROM access.

Address(es): A00D 0500h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	FORCE ECAT	CTRLP DI
Value after reset:	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	CTRLPDI	PDI EEPROM Control	Specifies whether EEPROM control is offered to the PDI. 0: The PDI has no EEPROM control. 1: The PDI has EEPROM control.	R	R/W
b1	FORCEECAT	EEPROM Access Right Change	Forcibly changes the right of access to the EEPROM by the ECAT. 0: No change 1: Reset bit 0 of the EEPROM PDI access state register (EEP_PDI_ACCESS at 0501h) to 0. That is, prohibit access to the EEPROM by the PDI.	R	R/W
b7 to b2	—	Reserved	When read, the value returned is undefined.	R	R

22.3.12.2 EEPROM PDI Access State Register (EEP_STATE)

This register is used to set the right of access to the EEPROM by the PDI.

Address(es): A00D 0501h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	PDIAC CESS
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	PDIACCESS	EEPROM Access Right Setting	Sets the right of access to the EEPROM. 0: Prohibits the PDI from access to the EEPROM. 1: The PDI has access to the EEPROM. Write access from the PDI is only possible when bit 0 is 1 and bit 1 is 0 in the EEPROM configuration register (EEP_CONF at 0500h).	R/(W)	R
b7 to b1	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.	R/(W)	R

22.3.12.3 EEPROM Control/Status Register (EEP_CONT_STAT)

This register is used to control access to the EEPROM and indicate the status.

Address(es): A00D 0502h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BUSY	WREN ERR	ACKCM DERR	LOADS TA	CKSU MERR	COMMAND			PROM SIZE	READB YTE	—	—	—	—	—	ECATW REN
Value after reset:	0	0	0	0	0	0	0	0	0	0	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	ECATWREN	Write Enable	ECAT write enable*2 0: Write requests are disabled. 1: Write requests are enabled. This bit is always 1 if the PDI has EEPROM control.	R	R/(W)
b5 to b1	—	Reserved	When read, the value returned is undefined.	R	R
b6	READBYTE	EEPROM Read Byte Indication	Indicates supported EEPROM read bytes. 0: 4 bytes 1: 8 bytes	R	R
b7	PROMSIZE	EEPROM Algorithm Indication	Indicates the selected EEPROM algorithm. 0: 1 address byte (1-Kbit to 16-Kbit EEPROMs) 1: 2 address bytes (32-Kbit to 4-Mbit EEPROMs)	R	R
b10 to b8	COMMAND	Command Setting/ Indication	Command*2 Write: Initiates the commands below. Read: Indicates the currently executed command. Commands: 000: No command/EEPROM idle (clear error bits) 001: Read 010: Write 100: Reload Others: Reserved/invalid commands (must not be issued)	R/(W)	R/(W)
b11	CKSUMERR	Checksum Error Indication	Indicates checksum error in the ESC configuration area. 0: No error in the checksum 1: Error in the checksum	R	R
b12	LOADSTA	EEPROM Loading Status Indication	Indicates EEPROM loading status. 0: EEPROM has been loaded and device information has no problem. 1: EEPROM has not been loaded and device information is not available (EEPROM loading in progress or finished with a failure).	R	R
b13	ACKCMDERR	Acknowledge/Command Error Indication	Indicates error acknowledge/command*1. 0: No error 1: Missing EEPROM acknowledge or invalid command	R	R
b14	WRENERR	Write Enable Error Indication	Indicates error write enable*1. 0: No error 1: Write command without write enable	R	R
b15	BUSY	EEPROM Interface State Indication	Indicates a busy state of the EEPROM interface. 0: The EEPROM Interface is idle. 1: The EEPROM Interface is busy.	R	R

Write access depends on the assignment of the EEPROM interface (ECAT/PDI). Write access is generally blocked if the EEPROM interface is busy (bit 15 = 1).

Note 1. Error bits are cleared by writing “000” (or any valid command) to command bits 10 to 8.

Note 2. ECAT write enable bit 0 is self-cleared at the SOF of the next frame. Command bits 10 to 8 are also self-cleared after the command is executed (EEPROM busy ends).
Writing “000” to command bits 10 to 8 will also clear the error bits 14 and 13. Command bits 10 to 8 are ignored if the acknowledge/command error bit 13 is 1.

22.3.12.4 EEPROM Address Register (EEP_ADR)

This register is used to set the EEPROM address to be accessed.

Address(es): A00D 0504h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ADDRESS															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADDRESS															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	ADDRESS	EEPROM Address Setting	EEPROM address 0: First word (= 16 bits) 1: Second word : Actually used EEPROM address bits: [9:0]: EEPROM size of up to 16 Kbits [17:0]: EEPROM size of 32 Kbits to 4 Mbits	R/(W)	R/(W)

Write access depends on the assignment of the EEPROM interface (ECAT/PDI). Write access is generally blocked if the EEPROM interface is busy (bit 15 is 1 in the EEPROM control/status register, EEP_CONT_STAT, at 0502h).

22.3.12.5 EEPROM Data Register (EEP_DATA)

This register is used to set write data to the EEPROM or indicates read data from the EEPROM. It can be written in 1-word units and read in 2-word units.

Address(es): A00D 0508h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	HIDATA															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LODATA															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	LODATA	EEPROM Write Data Setting	Data to be written to the EEPROM or data read from the EEPROM (lower 2 bytes)	R/(W)	R/(W)
b31 to b16	HIDATA	EEPROM Read Data Indication	Data read from the EEPROM (upper 2 bytes)	R	R

Write access depends on the assignment of the EEPROM interface (ECAT/PDI). Write access is generally blocked if the EEPROM interface is busy (bit 15 is 1 in the EEPROM control/status register, EEP_CONT_STAT, at 0502h).

22.3.13 MII Management Interface Registers

22.3.13.1 MII Management Control/Status Register (MII_CONT_STAT)

This register is used to control the MII management interface and to indicate the status.

Address(es): A00D 0510h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BUSY	CMDERR	READERR	—	—	—	COMMAND		PHYOFFSET					MILINK	PDICTRL	WREN
Value after reset:	0	0	0	x	x	x	0	0	0	0	0	0	0	0	1	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	WREN	Write Enable	Write enable 0: Disabled 1: Enabled This bit is always 1 if the PDI controls the MII management interface.	R	R/(W)
b1	PDICTRL	PDI Control Indication	Indicates whether the MII management interface can be controlled by the PDI. 0: Only ECAT control 1: PDI control possible The interface is controlled by the MII management ECAT access state register (MII_ECAT_ACS_STAT at 0516h) and the MII management PDI access state register (MII_PDI_ACS_STAT at 0517h).	R	R
b2	MILINK	Link Detection Availability Indication	MI link detection 0: Not available 1: Available	R	R
b7 to b3	PHYOFFSET	PHY Address Offset Indication	Indicate the PHY address offset.	R	R
b9, b8	COMMAND	Command	Command Write: Initiates the commands below. Read: Indicates the currently executed command. Commands: 00: No command/MI idle (clear error bits) 01: Read 10: Write Others: Reserved/invalid commands (must not be issued)	R/(W)	R/(W)
b12 to b10	—	Reserved	When read, the value returned is undefined.	R	R
b13	READERR	Read Error Indication	Indicates whether a read error occurred. 0: No read error 1: Read error occurred (PHY or register not available) This bit is cleared by writing to this register.	R/(W)	R/(W)
b14	CMDERR	Command Error Indication	Indicates whether a command error occurred. 0: Last command was successful 1: Invalid command or write command without write enable This bit is cleared by executing a valid command or writing “00” to command bits 9 and 8.	R	R
b15	BUSY	MII Management State Indication	Indicates that the MII management interface is busy. 0: MII management interface is idle. 1: MII management interface is busy.	R	R

Write access depends on the assignment of the management interface (ECAT/PDI). Write access is generally blocked if the management interface is busy (b15 in this register = 1).

Note: Write enable bit (b0) is self-cleared at the SOF of the next frame. Command bits (b9 and b8) are also self-cleared after the command is executed (busy ends).
Writing “00” to command bits will also clear the error bits (b14 and b13). Command bits are cleared after the command is executed.

22.3.13.2 PHY Address Register (PHY_ADR)

This register is used to set the PHY address.

Address(es): A00D 0512h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PHYADDR				
Value after reset:	x	x	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b4 to b0	PHYADDR	PHY Address Setting	PHY address	R/(W)	R/(W)
b7 to b5	—	Reserved	When read, the value returned is undefined.	R	R

Write access depends on the assignment of the management interface (ECAT/PDI). Write access is generally blocked if the management interface is busy (bit 15 is 1 in the MII management control/status register, MII_CONT_STAT, at 0510h).

22.3.13.3 PHY Register Address Register (PHY_REG_ADR)

This register is used to set the PHY register address.

Address(es): A00D 0513h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PHYREGADDR				
Value after reset:	x	x	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b4 to b0	PHYREGADDR	PHY Address Setting	Address of PHY register	R/(W)	R/(W)
b7 to b5	—	Reserved	When read, the value returned is undefined.	R	R

Write access depends on the assignment of the management interface (ECAT/PDI). Write access is generally blocked if the management interface is busy (bit 15 is 1 in the MII management control/status register, MII_CONT_STAT, at 0510h).

22.3.13.4 PHY Data Register (PHY_DATA)

This register is used to set data to write to PHY registers or to indicate data read from PHY registers.

Address(es): A00D 0514h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PHYREGDATA															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	PHYREGDATA	PHY Register Data Indication/Setting	PHY register read/write data	R/(W)	R/(W)

Write access depends on the assignment of the management interface (ECAT/PDI). Write access is generally blocked if the management interface is busy (bit 15 is 1 in the MII management control/status register, MII_CONT_STAT, at 0510h).

22.3.13.5 MII Management ECAT Access State Register (MII_ECAT_ACS_STAT)

This register is used to set the right of access to the MII management interface.

Address(es): A00D 0516h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ACSMII
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	ACSMII	MII Management Interface Access Right Setting	Right of access to the MII management interface 0: Enables access to the MII management interface by the PDI. 1: Exclusive access to the MII management interface by the ECAT	R	R/(W)
b7 to b1	—	Reserved	When read, the value returned is undefined.	R	R

Write access is only possible when bit 0 is 1 in the MII management PDI access state register (MII_PDI_ACS_STAT at 0517h).

22.3.13.6 MII Management PDI Access State Register (MII_PDI_ACS_STAT)

This register is used to set the right of access to the MII management interface.

Address(es): A00D 0517h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	FORPDI	ACSMII
Value after reset:	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	ACSMII	MII Management Interface Access Right Change	Right of access to the MII management interface 0: Access to the MII management interface by the PDI 1: Access to the MII management interface by the ECAT	R/(W)	R
b1	FORPDI	PDI Access State Change	Forced change of access by the PDI (forced change of bit 0) 0: Bit 0 of this register is not changed. 1: Bit 0 of this register is reset to 0 (the right of access is changed to the ECAT)	R	R/W
b7 to b2	—	Reserved	When read, the value returned is undefined.	R	R

Write access to bit 0 from the PDI is only possible if the following two conditions are satisfied.
- Bit 0 is 0 in the MII management ECAT access state register (MII_ECAT_ACS_STAT at 0516h).
- Bit 1 is 0 in the MII management PDI access state register (MII_PDI_ACS_STAT at 0517h).

22.3.14 FMMU Registers

22.3.14.1 FMMU Logical Start Address m Register (FMMUm_L_START_ADR)

This register is used to set the logical start address within the EtherCAT address space for FMMU.

Address(es): A00D 0600h : FMMU0_L_START_ADR
A00D 0610h : FMMU1_L_START_ADR
A00D 0620h : FMMU2_L_START_ADR
A00D 0630h : FMMU3_L_START_ADR
A00D 0640h : FMMU4_L_START_ADR
A00D 0650h : FMMU5_L_START_ADR
A00D 0660h : FMMU6_L_START_ADR
A00D 0670h : FMMU7_L_START_ADR

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LSTAADR															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LSTAADR															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	LSTAADR	Logical Start Address Setting	Set the start of the logical address within the EtherCAT address space.	R	R/W

m = 0 to 7

22.3.14.2 FMMU Length m Register (FMMUm_LEN)

This register is used to set the length for FMMU area in byte units.

Address(es): A00D 0604h : FMMU0_LEN
A00D 0614h : FMMU1_LEN
A00D 0624h : FMMU2_LEN
A00D 0634h : FMMU3_LEN
A00D 0644h : FMMU4_LEN
A00D 0654h : FMMU5_LEN
A00D 0664h : FMMU6_LEN
A00D 0674h : FMMU7_LEN

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FMMULEN															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	FMMULEN	Area Size Specification	Set the area size in byte units. Address at the end of the logical address range set by the FMMU minus the address at the start of the logical address range set by the FMMU plus one	R	R/W

m = 0 to 7

22.3.14.3 FMMU Logical Start Bit m Register (FMMUm_L_START_BIT)

This register is used to set the start bits of the logical start address for FMMU.

Address(es): A00D 0606h : FMMU0_L_START_BIT
A00D 0616h : FMMU1_L_START_BIT
A00D 0626h : FMMU2_L_START_BIT
A00D 0636h : FMMU3_L_START_BIT
A00D 0646h : FMMU4_L_START_BIT
A00D 0656h : FMMU5_L_START_BIT
A00D 0666h : FMMU6_L_START_BIT
A00D 0676h : FMMU7_L_START_BIT

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	LSTABIT		
Value after reset:	x	x	x	x	x	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b2 to b0	LSTABIT	Start Bit Setting	Set the start bits of the logical start address for FMMU.	R	R/W
b7 to b3	—	Reserved	When read, the value returned is undefined.	R	R

m = 0 to 7

22.3.14.4 FMMU Logical Stop Bit m Register (FMMUm_L_STOP_BIT)

This register is used to set the last bits of the logical end address for FMMU.

Address(es): A00D 0607h : FMMU0_L_STOP_BIT
A00D 0617h : FMMU1_L_STOP_BIT
A00D 0627h : FMMU2_L_STOP_BIT
A00D 0637h : FMMU3_L_STOP_BIT
A00D 0647h : FMMU4_L_STOP_BIT
A00D 0657h : FMMU5_L_STOP_BIT
A00D 0667h : FMMU6_L_STOP_BIT
A00D 0677h : FMMU7_L_STOP_BIT

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	LSTPBIT		
Value after reset:	x	x	x	x	x	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b2 to b0	LSTPBIT	Last Bit Setting	Set the last bits of the logical end address for FMMU.	R	R/W
b7 to b3	—	Reserved	When read, the value returned is undefined.	R	R

m = 0 to 7

22.3.14.5 FMMU Physical Start Address m Register (FMMUm_P_START_ADR)

This register is used to set the physical start address of the ESC to which the logical start address will be mapped by the FMMU.

Address(es): A00D 0608h : FMMU0_P_START_ADR
A00D 0618h : FMMU1_P_START_ADR
A00D 0628h : FMMU2_P_START_ADR
A00D 0638h : FMMU3_P_START_ADR
A00D 0648h : FMMU4_P_START_ADR
A00D 0658h : FMMU5_P_START_ADR
A00D 0668h : FMMU6_P_START_ADR
A00D 0678h : FMMU7_P_START_ADR

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PHYSTAADR															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	PHYSTAADR	Physical Start Address Setting	Set the physical start address to which the logical start address will be mapped. The address is set as an offset from the base address (A00D 0000h).	R	R/W

m = 0 to 7

22.3.14.6 FMMU Physical Start Bit m Register (FMMUm_P_START_BIT)

This register is used to set the start bits of the physical start address of the ESC to which the start bits of the logical start address will be mapped by the FMMU.

Address(es): A00D 060Ah : FMMU0_P_START_BIT
A00D 061Ah : FMMU1_P_START_BIT
A00D 062Ah : FMMU2_P_START_BIT
A00D 063Ah : FMMU3_P_START_BIT
A00D 064Ah : FMMU4_P_START_BIT
A00D 065Ah : FMMU5_P_START_BIT
A00D 066Ah : FMMU6_P_START_BIT
A00D 067Ah : FMMU7_P_START_BIT

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	PHYSTABIT		
Value after reset:	x	x	x	x	x	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b2 to b0	PHYSTABIT	Physical Start Bit Setting	Set the start bits of the physical start address to which the start bits of the logical start address will be mapped.	R	R/W
b7 to b3	—	Reserved	When read, the value returned is undefined.	R	R

m = 0 to 7

22.3.14.7 FMMU Type m Register (FMMUm_TYPE)

This register is used to set the type of FMMU access.

Address(es): A00D 060Bh : FMMU0_TYPE
A00D 061Bh : FMMU1_TYPE
A00D 062Bh : FMMU2_TYPE
A00D 063Bh : FMMU3_TYPE
A00D 064Bh : FMMU4_TYPE
A00D 065Bh : FMMU5_TYPE
A00D 066Bh : FMMU6_TYPE
A00D 067Bh : FMMU7_TYPE

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	WRITE	READ
Value after reset:	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	READ	Read Access Mapping Setting	Sets the mapping for read access. 0: Disabled 1: Enabled	R	R/W
b1	WRITE	Write Access Mapping Setting	Sets the mapping for write access. 0: Disabled 1: Enabled	R	R/W
b7 to b2	—	Reserved	When read, the value returned is undefined.	R	R

m = 0 to 7

22.3.14.8 FMMU Activate m Register (FMMUm_ACT)

This register is used to enable or disable FMMU.

Address(es): A00D 060Ch : FMMU0_ACT
A00D 061Ch : FMMU1_ACT
A00D 062Ch : FMMU2_ACT
A00D 063Ch : FMMU3_ACT
A00D 064Ch : FMMU4_ACT
A00D 065Ch : FMMU5_ACT
A00D 066Ch : FMMU6_ACT
A00D 067Ch : FMMU7_ACT

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ACTIVATE
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	ACTIVATE	FMMU Enable/Disable	Enables or disables FMMU. 0: Disabled 1: Enabled	R	R/W
b7 to b1	—	Reserved	When read, the value returned is undefined.	R	R

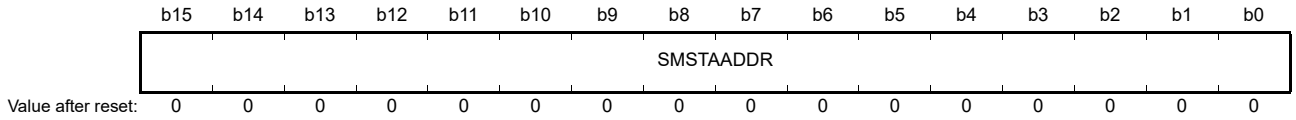
m = 0 to 7

22.3.15 SyncManager Registers

22.3.15.1 SyncManager Physical Start Address m Register (SMm_P_START_ADR)

This register is used to set the physical start address for the area assigned to SyncManager.

Address(es): A00D 0800h : SM0_P_START_ADR
A00D 0808h : SM1_P_START_ADR
A00D 0810h : SM2_P_START_ADR
A00D 0818h : SM3_P_START_ADR
A00D 0820h : SM4_P_START_ADR
A00D 0828h : SM5_P_START_ADR
A00D 0830h : SM6_P_START_ADR
A00D 0838h : SM7_P_START_ADR



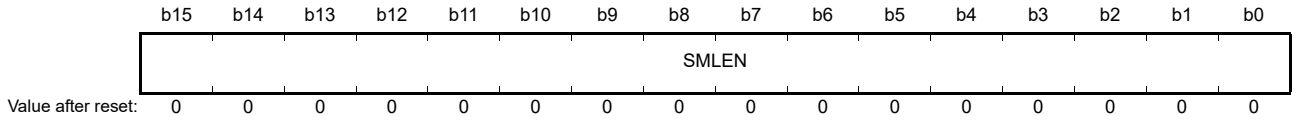
Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	SMSTAADDR	Physical Start Address Setting	Specify the physical start address for the area assigned to SyncManager.	R	R/(W)

m = 0 to 7
This register can only be written when SyncManager is disabled (bit 0 is 0 in the SyncManager activate m register (SMm.ACT at 0806h+8h*m).

22.3.15.2 SyncManager Length m Register (SMm_LEN)

This register is used to set the size of the area assigned to SyncManager.

Address(es): A00D 0802h : SM0_LEN
A00D 080Ah : SM1_LEN
A00D 0812h : SM2_LEN
A00D 081Ah : SM3_LEN
A00D 0822h : SM4_LEN
A00D 082Ah : SM5_LEN
A00D 0832h : SM6_LEN
A00D 083Ah : SM7_LEN



Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	SMLEN	Area Size Setting	Set the number of bytes assigned to SyncManager. Set a value greater than 1. Otherwise, SyncManager is not activated.	R	R/(W)

m = 0 to 7
This register can only be written when SyncManager is disabled (bit 0 is 0 in the SyncManager activate m register (SMm.ACT at 0806h+8h*m).

22.3.15.3 SyncManager Control m Register (SMm_CONTROL)

This register is used to set operation of SyncManager.

Address(es): A00D 0804h : SM0_CONTROL
A00D 080Ch : SM1_CONTROL
A00D 0814h : SM2_CONTROL
A00D 081Ch : SM3_CONTROL
A00D 0824h : SM4_CONTROL
A00D 082Ch : SM5_CONTROL
A00D 0834h : SM6_CONTROL
A00D 083Ch : SM7_CONTROL

	b7	b6	b5	b4	b3	b2	b1	b0
	—	WDTRGEN	IRQPDI	IRQECAT	DIR		OPEMODE	
Value after reset:	x	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b1, b0	OPEMODE	Operating Mode Setting	Operating mode 00: Buffer mode (3 buffer mode) 10: Mailbox mode (single buffer mode) Others: Reserved	R	R/(W)
b3, b2	DIR	Transfer Direction Setting	Transfer direction 00: Read (ECAT: read access; PDI: write access) 01: Write (ECAT: write access; PDI: read access) Others: Reserved	R	R/(W)
b4	IRQECAT	ECAT Event Interrupt Setting	Enables or disables interrupts (ECAT interrupts) by the ECAT event request register (ECAT_EVENT_REQ at 0210h). 0: Disabled 1: Enabled	R	R/(W)
b5	IRQPDI	AL Event Interrupt Setting	Enables or disables interrupts (PDI interrupts) by the AL event request register (AL_EVENT_REQ at 0220h). 0: Disabled 1: Enabled	R	R/(W)
b6	WDTRGEN	Watchdog Trigger Setting	Enables or disables watchdog trigger. 0: Disabled 1: Enabled	R	R/(W)
b7	—	Reserved	When read, the value returned is undefined.	R	R

m = 0 to 7
This register can only be written when SyncManager is disabled (bit 0 is 0 in the SyncManager activate m register (SMm.ACT at 0806h+8h*m).

22.3.15.4 SyncManager Status m Register (SMm_STATUS)

This register indicates the state of SyncManager.

Address(es): A00D 0805h : SM0_STATUS
A00D 080Dh : SM1_STATUS
A00D 0815h : SM2_STATUS
A00D 081Dh : SM3_STATUS
A00D 0825h : SM4_STATUS
A00D 082Dh : SM5_STATUS
A00D 0835h : SM6_STATUS
A00D 083Dh : SM7_STATUS

	b7	b6	b5	b4	b3	b2	b1	b0
	WRBUF	RDBUF	BUFFERED		MAILBOX	—	INTRD	INTWR
Value after reset:	0	0	1	1	0	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	INTWR	Write Complete Interrupt State Indication	Indicates write complete interrupt. 1: Indicates that the first byte of the buffer was read (interrupt cleared) 0: Indicates that the buffer was successfully written.	R	R
b1	INTRD	Read Complete Interrupt State Indication	Indicates read complete interrupt. 1: Indicates that the first byte of the buffer was written (interrupt cleared) 0: Indicates that the buffer was successfully read.	R	R
b2	—	Reserved	When read, the value returned is undefined.	R	R
b3	MAILBOX	Mailbox Status Indication	Indicates the mailbox status in mailbox mode. 0: Mailbox empty 1: Mailbox full This bit is not used in buffer mode.	R	R
b5, b4	BUFFERED	Buffer Status Indication	Indicates the buffer status in buffer mode (last written buffer). 00: 1st buffer 01: 2nd buffer 10: 3rd buffer 11: No buffer written This bit is not used in mailbox mode.	R	R
b6	RDBUF	Read State Indication	Indicates that the buffer is being read.	R	R
b7	WRBUF	Write State Indication	Indicates that the buffer is being written.	R	R

m = 0 to 7

22.3.15.5 SyncManager Activate m Register (SMm_ACT)

This register is used to set operation of SyncManager.

Address(es): A00D 0806h : SM0_ACT
A00D 080Eh : SM1_ACT
A00D 0816h : SM2_ACT
A00D 081Eh : SM3_ACT
A00D 0826h : SM4_ACT
A00D 082Eh : SM5_ACT
A00D 0836h : SM6_ACT
A00D 083Eh : SM7_ACT

	b7	b6	b5	b4	b3	b2	b1	b0
	LATCH PDI	LATCH ECAT	—	—	—	—	REPEA TREQ	SMEN
Value after reset:	0	0	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	SMEN	SyncManager Enable/ Disable	Enables or disables SyncManager. 0: Disabled. Memory is accessed without SyncManager control. 1: Enabled. SyncManager is active and controls memory area set in configuration.	R (ack)	R/W
b1	REPEATREQ	Repeat Request	Repeat request Toggling of the repeat request signal means that retrying the mailbox is required (primarily used in conjunction with reading of the ECAT mailbox).	R (ack)	R/W
b5 to b2	—	Reserved	When read, the value returned is undefined.	R (ack)	R
b6	LATCHECAT	ECAT Latch Event Specification	ECAT latch event 0: No 1: Generates latch events if the EtherCAT master switches the buffers.	R (ack)	R/W
b7	LATCHPDI	PDI Latch Event Specification	PDI latch event 0: No 1: Generates latch events if the PDI switches the buffers or accesses the buffer start address.	R (ack)	R/W

m = 0 to 7
Reading this register from the PDI in all SyncManagers which have changed activation clears bit 4 of the AL event request register (AL_EVENT_REQ at 0220h).

22.3.15.6 SyncManager PDI Control m Register (SMm_PDI_CONT)

This register is used to set operation of SyncManager from the PDI.

Address(es): A00D 0807h : SM0_PDI_CONT
A00D 080Fh : SM1_PDI_CONT
A00D 0817h : SM2_PDI_CONT
A00D 081Fh : SM3_PDI_CONT
A00D 0827h : SM4_PDI_CONT
A00D 082Fh : SM5_PDI_CONT
A00D 0837h : SM6_PDI_CONT
A00D 083Fh : SM7_PDI_CONT

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	REPEATACK	DEACTIVE
Value after reset:	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	DEACTIVE	SyncManager Operation Indication/Setting	Deactivates SyncManager. Read: 0: Normal operation. SyncManager is activated. 1: SyncManager is deactivated and reset. SyncManager locks access to memory area. Write: 0: Activates SyncManager. 1: Deactivates SyncManager. Note: Writing 1 is delayed until the end of a frame which is currently processed.	R/W	R
b1	REPEATACK	Repeat Acknowledge	Repeat Acknowledge If this bit is set to the same value as set by bit 0 (repeat request) of the SyncManager activate register (SMm.ACT at 0806h+8h*m), the PDI acknowledges the execution of a previous set repeat request.	R/W	R
b7 to b2	—	Reserved	When read, the value returned is undefined.	R	R

m = 0 to 7

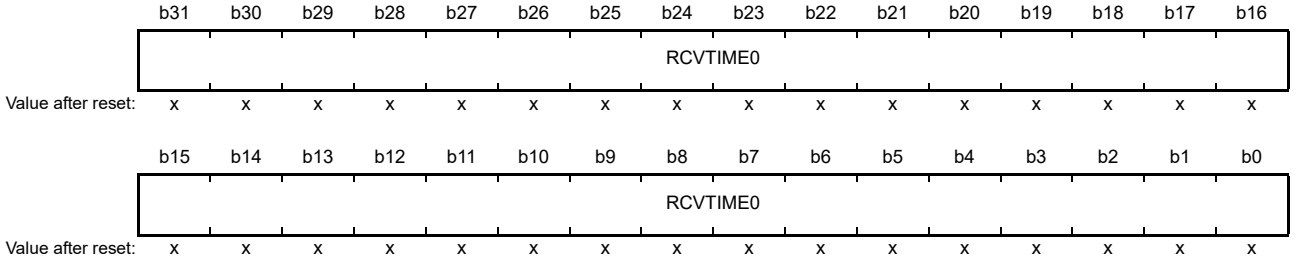
22.3.16 Distributed Clock Registers

22.3.16.1 DC Receive Time Registers

(1) Receive Time Port 0 Register (DC_RCV_TIME_PORT0)

Writing to this register latches the received time of frames at all ports and reading this register indicates the received time of a frame latched at port 0.

Address(es): A00D 0900h



x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	RCVTIME0	Receive Time Indication/ Latch	Write: A write access to this register with the BWR, APWR (any address) or FPWR (configured address) command latches the local time when each port starts to receive a frame (first start bit of preamble). Read: Indicate the local time at the beginning of reception of the last frame containing a write access to this register. Note: The time stamps cannot be read in the same frame in which this register was written.	R	R/W

(2) Receive Time Port 1 Register (DC_RCV_TIME_PORT1)

This register indicates the received time of the frame latched at port 1.

Address(es): A00D 0904h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RCVTIME1															
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RCVTIME1															
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	RCVTIME1	Receive Time Indication	Indicate the local time when port 1 starts to receive a frame (first start bit of preamble) containing the BWR, APWR or FPWR command to the receive time port 0 register (DC_RCV_TIME_PORT0 at 0900h).	R	R

22.3.16.2 Time Loop Control Unit Registers

(1) System Time Register (DC_SYS_TIME)

This register indicates the local copy of the system time.

Address(es): A00D 0910h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	SYSTIME	System Time Indication	Access from the ECAT Read: Indicate the local copy of the system time when the frame passed the reference clock (i.e., including a system time delay). The time latched at the start of frame (SOF) is indicated. Write: A written value is compared with the local copy of the system time. The result is input to the time control loop unit.	R	R/W
			Access from the PDI Read: Indicate the local copy of the system time. The time latched when the first byte of this register was read is indicated.	R	R

(2) Receive Time ECAT Processing Unit Register (DC_RCV_TIME_UNIT)

This register indicates the received time of a frame latched at EtherCAT processing unit.

Address(es): A00D 0918h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	RCVTIMEEPU															
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	RCVTIMEEPU															
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RCVTIMEEPU															
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RCVTIMEEPU															
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	RCVTIMEEPU	Receive Time Indication	This register indicates the local time at the beginning of reception by the EtherCAT processing unit of a frame (i.e. the first start bit of the preamble), including write access to the receive time port 0 register (DC_RCV_TIME_PORT0 at 0900h).	R	R

If port 0 is open, the value in this register reflects the same time as the value in the receive time port 0 register (DC_RCV_TIME_PORT0 at 0900h), but as 64 bits.

(3) System Time Offset Register (DC_SYS_TIME_OFFSET)

This register is used to indicate a difference (offset) between the local time and system time.

Address(es): A00D 0920h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	SYSTIMOFST															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	SYSTIMOFST															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SYSTIMOFST															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYSTIMOFST															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	SYSTIMOFST	System Time and Local Time Difference Indication	Indicate a difference between the local time and system time. This offset is added to the local time to obtain the local system time.	R	R/W

(4) System Time Delay Register (DC_SYS_TIME_DELAY)

This register indicates a propagation delay between the reference clock and slave (ESC).

Address(es): A00D 0928h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SYSTIMDLY															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYSTIMDLY															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	SYSTIMDLY	Propagation Delay Indication	Indicate a delay between the reference clock and the ESC.	R	R/W

(5) System Time Difference Register (DC_SYS_TIME_DIFF)

This register indicates a mean difference between the local copy of the system time and received system time.

Address(es): A00D 092Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LOCAL COPY	DIFF														
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DIFF															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31	LOCALCOPY	System Time Greater/Less Indication	Indicates whether the local copy of the system time is greater than or equal to, or is less than, the latest received copy of the system time. 0: Local copy of the system time greater than or equal to the received system time 1: Local copy of the system time less than the received system time	R	R
b30 to b0	DIFF	System Time Mean Difference Indication	Indicates a mean difference between the local copy of the system time and received system time.	R	R

(6) Speed Counter Start Register (DC_SPEED_COUNT_START)

This register is used to set the bandwidth for drift correction of the local copy of the system time.

Address(es): A00D 0930h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	SPDCNTSTRT														
Value after reset:	x	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b14 to b0	SPDCNTSTRT	Drift Correction Bandwidth Setting	Indicate the bandwidth for adjustment of the local copy of the system time (larger values → smaller bandwidth and smoother adjustment). A write access resets the system time difference register (DC_SYS_TIME_DIFF at 092Ch) and the speed counter difference register (DC_SPEED_COUNT_DIFF at 0932h). Valid range: 0080h to 3FFFh	R	R/W
b15	—	Reserved	When read, the value returned is undefined. When writing to this bit, write 0.	R	R/W

(7) Speed Counter Difference Register (DC_SPEED_COUNT_DIFF)

This register indicates the deviation between the local clock period and reference clock's clock period.

Address(es): A00D 0932h

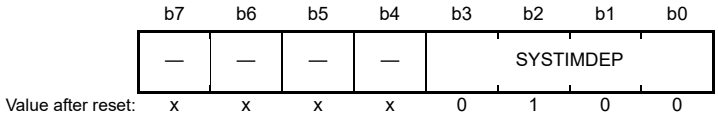
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SPDCNTDIFF															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	SPDCNTDIFF	Clock Period Deviation Indication	Indicate the deviation between the local clock period and reference clock's clock period (represented by two's complements). Range: ± (speed counter start value – 7Fh)	R	R

(8) System Time Difference Filter Depth Register (DC_SYS_TIME_DIFF_FIL_DEPTH)

This register is used to set the filter depth for averaging the received system time deviation.

Address(es): A00D 0934h



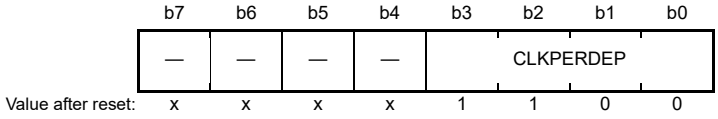
x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b3 to b0	SYSTIMDEP	Filter Depth Setting	Set the filter depth for averaging the received system time deviation. A write access resets the system time difference register (DC_SYS_TIME_DIFF at 092Ch).	R	R/W
b7 to b4	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.	R	R/W

(9) Speed Counter Filter Depth Register (DC_SPEED_COUNT_FIL_DEPTH)

This register is used to set the filter depth for averaging the clock period deviation.

Address(es): A00D 0935h



x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b3 to b0	CLKPERDEP	Filter Depth Setting	Set the filter depth for averaging the clock period deviation. A write access resets the internal speed counter filter.	R	R/W
b7 to b4	—	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.	R	R/W

22.3.16.3 Cyclic Unit Control Registers

(1) Cyclic Unit Control Register (DC_CYC_CONT)

This register sets whether to control SYNC and latch units by the ECAT or PDI.

Address(es): A00D 0980h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	LATCH1	LATCH0	—	—	—	SYNCOUT
Value after reset:	x	x	0	0	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	SYNCOUT	SYNC Output Unit Control Setting	Sets control of the SYNC output unit. 0: ECAT control 1: PDI control	R	R/W
b3 to b1	—	Reserved	When read, the value returned is undefined.	R	R
b4	LATCH0	Latch Input Unit 0 Control Setting	Sets control of latch input unit 0. 0: ECAT control 1: PDI control Note: Latch interrupt is routed to the ECAT or PDI in accord with this setting.	R	R/W
b5	LATCH1	Latch Input Unit 1 Control Setting	Sets control of latch input unit 1. 0: ECAT control 1: PDI control Note: Latch interrupt is routed to the ECAT or PDI in accord with this setting.	R	R/W
b7, b6	—	Reserved	When read, the value returned is undefined.	R	R

22.3.16.4 SYNC Output Unit Registers

(1) Activation Register (DC_ACT)

This register is used to activate the Sync output unit.

Address(es): A00D 0981h

	b7	b6	b5	b4	b3	b2	b1	b0
	DBGPULSE	NEARFUTURE	STARTTIME	EXTSTARTTIME	AUTOACT	SYNC1	SYNC0	SYNCACT
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	SYNCACT	Sync Output Unit Activation	Activates the Sync output unit. 0: Deactivated 1: Activated Note: Write 1 after the start time was written.	R/(W)	R/(W)
b1	SYNC0	SYNC0 Output Setting	Sets SYNC0 output. 0: Deactivated 1: SYNC0 pulse output is generated.	R/(W)	R/(W)
b2	SYNC1	SYNC1 Output Setting	Sets SYNC1 output. 0: Deactivated 1: SYNC1 pulse output is generated.	R/(W)	R/(W)
b3	AUTOACT	SYNC Output Unit Activation	Sets whether to activate the Sync output unit automatically by writing to the start time cyclic operation register (DC_CYC_START_TIME at 0990h): 0: Deactivated 1: Activated. Bit 0 is automatically set to 1 in this register after the start time is written.	R/(W)	R/(W)
b4	EXTSTARTTIME	Start Time Cyclic Operation Extension	Extends start time cyclic operation. 0: No extension 1: Extends the start time written with 32 bits to 64 bits	R/(W)	R/(W)
b5	STARTTIME	Start Time Plausibility	Selects whether checking the plausibility of the start time and response to implausible start times is to proceed. 0: Disabled. Sync signal is generated if the start time is reached. 1: Sync signal is generated immediately if the start time is outside the range of the near future.	R/(W)	R/(W)
b6	NEARFUTURE	Near Future Range Setting	Sets the range to be considered the near future. 0: Up to 2^{63} ns from now (1/2 of the DC width) 1: Up to 2^{31} ns from now (about 2.1 s)	R/(W)	R/(W)
b7	DBGPULSE	Debug Pulse Setting	Sets Sync signal debug pulse. 0: Deactivated 1: Immediately generates a single debug ping on the SYNC0 and SYNC1 pins in accord with the setting of bits 2 and 1 of this register. This bit is self-cleared and always read as 0.	R/(W)	R/(W)

Writing to this register depends on the setting of bit 0 of the cyclic unit control register (DC_CYC_CONT at 0980h).

(2) SYNC Signal Pulse Length Register (DC_PULSE_LEN)

This register indicates the pulse length of SYNC signals.

Address(es): A00D 0982h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PULSELEN															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b15 to b0	PULSELEN	SYNC Signal Pulse Length Indication	Indicate the pulse length of SYNC signals (in units of 10 ns) 0: Acknowledge mode. In this mode, SYNC signal is cleared by reading the SYNC0 or SYNC1 status register (DC_SYNC0/1_STAT at 098Eh, 098Fh).	R	R

(3) Activation Status Register (DC_ACT_STAT)

This register indicates the activation status of SYNC output signals.

Address(es): A00D 0984h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	START TIME	SYNC1 ACT	SYNC0 ACT
Value after reset:	x	x	x	x	x	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	SYNC0ACT	SYNC0 Status Indication	Indicates the activation state of SYNC0. 0: First SYNC0 pulse is not pending. 1: First SYNC0 pulse is pending.	R	R
b1	SYNC1ACT	SYNC1 Status Indication	Indicates the activation state of SYNC1. 0: First SYNC1 pulse is not pending. 1: First SYNC1 pulse is pending.	R	R
b2	STARTTIME	Plausibility Result Indication	Indicates the plausibility check result of the start time cyclic operation register (DC_CYC_START_TIME at 0990h) while the Sync output unit is activated. 0: The start time was within the near future. 1: The start time was out of the near future.	R	R
b7 to b3	—	Reserved	When read, the value returned is undefined.	R	R

(4) SYNC0 Status Register (DC_SYNC0_STAT)

This register indicates the state of SYNC0 output. It is only used in acknowledge mode.

Address(es): A00D 098Eh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	SYNC0 STA
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	SYNC0STA	SYNC0 State Indication	Indicates the SYNC0 state for acknowledge mode. SYNC0 in acknowledge mode is cleared by reading this register from the PDI. This bit is only used in acknowledge mode.	R (ack)	R
b7 to b1	—	Reserved	When read, the value returned is undefined.	R	R

(5) SYNC1 Status Register (DC_SYNC1_STAT)

This register indicates the state of SYNC1 output. It is only used in acknowledge mode.

Address(es): A00D 098Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	SYNC1 STA
Value after reset:	x	x	x	x	x	x	x	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	SYNC1STA	SYNC1 State Indication	Indicates the SYNC1 state for acknowledge mode. SYNC1 in acknowledge mode is cleared by reading this register from the PDI. This bit is only used in acknowledge mode.	R (ack)	R
b7 to b1	—	Reserved	When read, the value returned is undefined.	R	R

(6) Start Time Cyclic Operation/Next SYNC0 Pulse Register (DC_CYC_START_TIME)

Writing to this register sets the start time of cyclic operation. Reading this register indicates the system time of the next SYNC0 pulse.

Address(es): A00D 0990h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	STATIM															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	STATIM															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	STATIM															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	STATIM															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

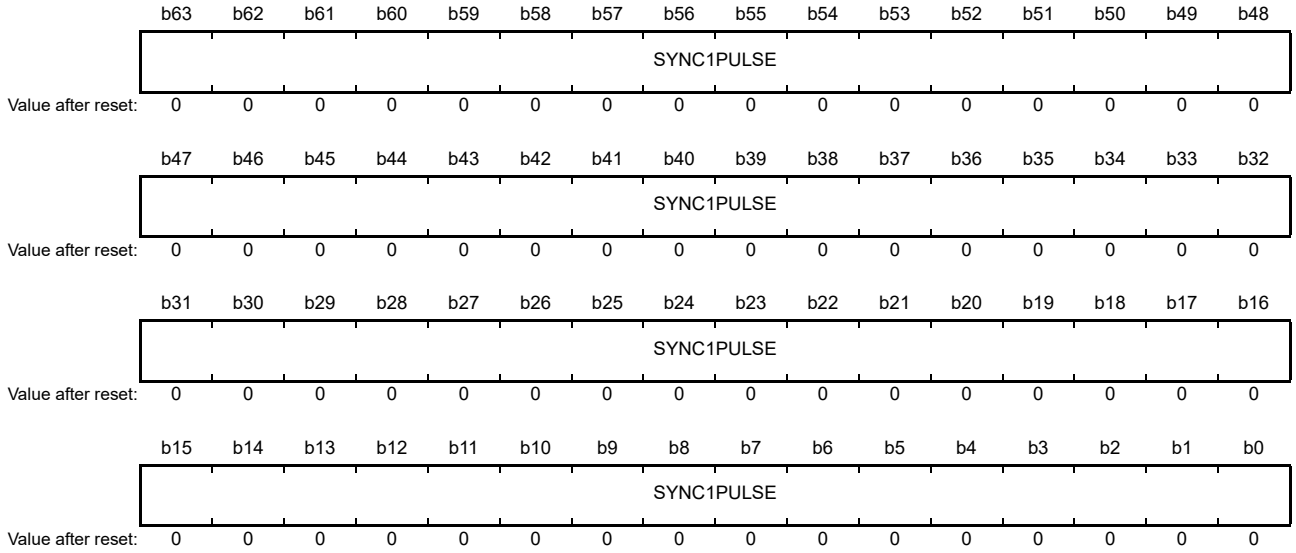
Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	STATIM	Start Time Setting/ System Time Indication	Write: Set the start time (in the system time) of cyclic operation in ns units. Read: Indicate the system time of the next SYNC0 pulse in ns units.	R/(W)	R/(W)

Writing to this register depends on the setting of bit 0 of the cyclic unit control register (DC_CYC_CONT at 0980h). Only writable when bit 0 is 0 in the SYNC activation register (DC_ACT at 0981h).
When auto-activation is enabled, upper 32 bits are automatically extended if only lower 32 bits are written within one frame.

(7) Next SYNC1 Pulse Register (DC_NEXT_SYNC1_PULSE)

This register indicates the system time of the next SYNC1 pulse.

Address(es): A00D 0998h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	SYNC1PULSE	SYNC1 Pulse System Time Indication	Indicate the system time of the next SYNC1 pulse in ns units.	R	R

(8) SYNC0 Cycle Time Register (DC_SYNC0_CYC_TIME)

This register is used to set the time between consecutive SYNC0 pulses.

Address(es): A00D 09A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SYNC0CYC															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC0CYC															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	SYNC0CYC	Time Between Consecutive SYNC0 Pulses	Set the time between two consecutive SYNC0 pulses in ns units. 0: Single shot mode. Only one SYNC0 pulse is generated in single shot mode.	R/(W)	R/(W)

Writing to this register depends on the setting of bit 0 of the cyclic unit control register (DC_CYC_CONT at 0980h).

(9) SYNC1 Cycle Time Register (DC_SYNC1_CYC_TIME)

This register is used to set the time between SYNC1 and SYNC0 pulses.

Address(es): A00D 09A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SYNC1CYC															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYNC1CYC															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	SYNC1CYC	Time between SYNC1 and SYNC0 Pulses	Set the time between SYNC1 and SYNC0 pulses in ns units.	R/(W)	R/(W)

Writing to this register depends on the setting of bit 0 of the cyclic unit control register (DC_CYC_CONT at 0980h).

22.3.16.5 Latch Input Unit Registers

(1) Latch 0 Control Register (DC_LATCH0_CONT)

This register is used to control the edge function of the latch 0 input signal.

Address(es): A00D 09A8h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	NEGEDGE	POSEDGE
Value after reset:	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	POSEDGE	Latch 0 Positive Edge Function Setting	Sets the function of the rising edge of the latch 0 input signal. 0: Continuous latch active 1: Single event (only first event active)	R/(W)	R/(W)
b1	NEGEDGE	Latch 0 Negative Edge Function Setting	Sets the function of the falling edge of the latch 0 input signal. 0: Continuous latch active 1: Single event (only first event active)	R/(W)	R/(W)
b7 to b2	—	Reserved	When read, the value returned is undefined.	R/(W)	R

Writing to this register depends on the setting of bit 4 of the cyclic unit control register (DC_CYC_CONT at 0980h).

(2) Latch 1 Control Register (DC_LATCH1_CONT)

This register is used to control the edge function of the latch 1 input signal.

Address(es): A00D 09A9h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	NEGEDGE	POSEDGE
Value after reset:	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	POSEDGE	Latch 1 Positive Edge Function Setting	Sets the function of the rising edge of the latch 1 input signal. 0: Continuous latch active 1: Single event (only first event active)	R/(W)	R/(W)
b1	NEGEDGE	Latch 1 Negative Edge Function Setting	Sets the function of the falling edge of the latch 1 input signal. 0: Continuous Latch active 1: Single event (only first event active)	R/(W)	R/(W)
b7 to b2	—	Reserved	When read, the value returned is undefined.	R/(W)	R

Writing to this register depends on the setting of bit 5 of the cyclic unit control register (DC_CYC_CONT at 0980h).

(3) Latch 0 Status Register (DC_LATCH0_STAT)

This register indicates the state of the latch 0 input signal.

Address(es): A00D 09AEh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	PINSTATE	EVENTNEG	EVENTPOS
Value after reset:	x	x	x	x	x	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	EVENTPOS	Latch 0 Positive Edge Event Indication	Indicates detection of rising edges of the event latch 0 signal. 0: Rising edge not detected or continuous mode 1: Rising edge detected and mode is single-event. This flag is cleared by reading the latch 0 time positive edge register (DC_LATCH0_TIME_POS at 09B0h).	R	R
b1	EVENTNEG	Latch 0 Negative Edge Event Indication	Indicates detection of falling edges of the event latch 0 signal. 0: Falling edge not detected or continuous mode 1: Falling edge detected and mode is single-event. This flag is cleared by reading the latch 0 time negative edge register (DC_LATCH0_TIME_NEG at 09B8h).	R	R
b2	PINSTATE	Latch 0 Input Pin State Indication	Indicates the state of the latch 0 input pin.	R	R
b7 to b3	—	Reserved	When read, the value returned is undefined.	R	R

(4) Latch 1 Status Register (DC_LATCH1_STAT)

This register indicates the state of the latch 1 input signal.

Address(es): A00D 09AFh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	PINSTATE	EVENTNEG	EVENTPOS
Value after reset:	x	x	x	x	x	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b0	EVENTPOS	Latch 1 Positive Edge Event Indication	Indicates detection of rising edges of the event latch 1 signal. 0: Rising edge not detected or continuous mode 1: Rising edge detected and mode is single-event. This flag is cleared by reading the latch 1 time positive edge register (DC_LATCH1_TIME_POS at 09C0h).	R	R
b1	EVENTNEG	Latch 1 Negative Edge Event Indication	Indicates detection of falling edges of the event latch 1 signal. 0: Falling edge not detected or continuous mode 1: Falling edge detected and mode is single-event. This flag is cleared by reading the latch 1 time negative edge register (DC_LATCH1_TIME_NEG at 09C8h).	R	R
b2	PINSTATE	Latch 1 Input Pin State Indication	Indicates the state of the latch 1 input pin.	R	R
b7 to b3	—	Reserved	When read, the value returned is undefined.	R	R

(5) Latch 0 Time Positive Edge Register (DC_LATCH0_TIME_POS)

This register indicates the system time at the rising edge of the latch 0 input signal.

Address(es): A00D 09B0h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	SYSTIME	System Time Indication	Indicate the system time captured at the rising edge of the latch 0 input signal. Reading this register clears bit 0 of the latch 0 status register (DC_LATCH0_STAT at 09AEh).	R (ack)	R (ack)

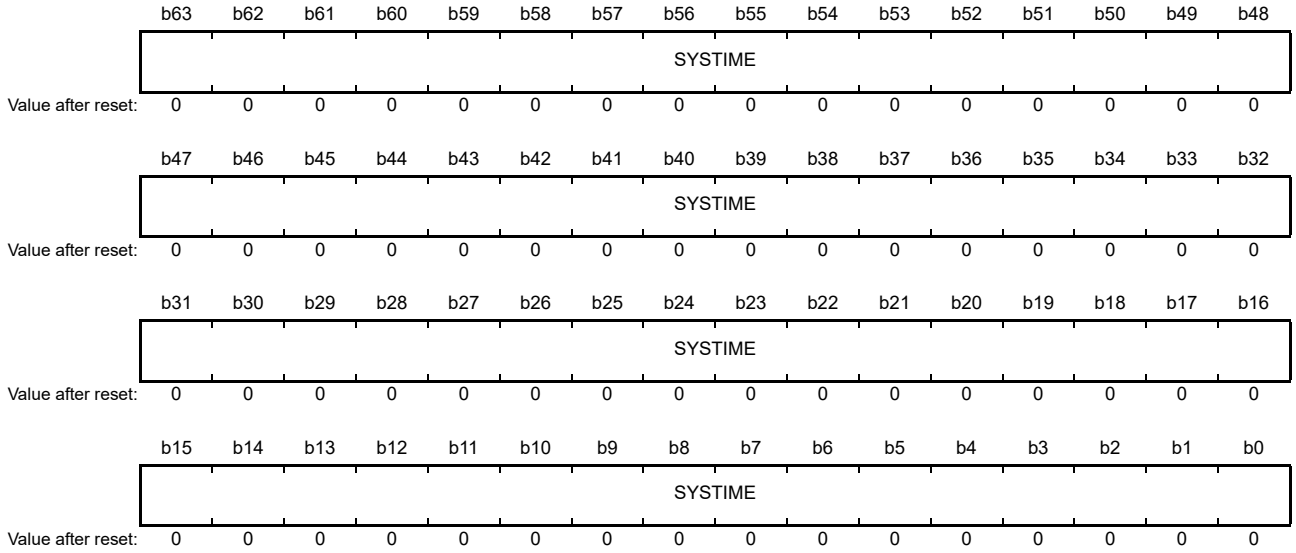
Bits 63 to 8 are internally latched (ECAT/PDI independently) when bits 7 to 0 are read, which guarantees reading a consistent value.

Clearing the latch 0 status flag function depends on the setting of bit 4 of the cyclic unit control register (DC_CYC_CONT at 0980h).

(6) Latch 0 Time Negative Edge Register (DC_LATCH0_TIME_NEG)

This register indicates the system time at the falling edge of the latch 0 input signal.

Address(es): A00D 09B8h



Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	SYSTIME	System Time Indication	Indicate the system time captured at the falling edge of the latch 0 input signal. Reading this register clears bit 1 of the latch 0 status register (DC_LATCH0_STAT at 09AEh).	R (ack)	R (ack)

Bits 63 to 8 are internally latched (ECAT/PDI independently) when bits 7 to 0 are read, which guarantees reading a consistent value.

Clearing the latch 0 status flag function depends on the setting of bit 4 of the cyclic unit control register (DC_CYC_CONT at 0980h).

(7) Latch 1 Time Positive Edge Register (DC_LATCH1_TIME_POS)

This register indicates the system time at the rising edge of the latch 1 input signal.

Address(es): A00D 09C0h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	SYSTIME	System Time Indication	Indicate the system time captured at the rising edge of the latch 1 input signal. Reading this register clears bit 0 of the latch 1 status register (DC_LATCH1_STAT at 09AFh).	R (ack)	R (ack)

Bits 63 to 8 are internally latched (ECAT/PDI independently) when bits 7 to 0 are read, which guarantees reading a consistent value.

Clearing the latch 1 status flag function depends on the setting of bit 5 of the cyclic unit control register (DC_CYC_CONT at 0980h).

(8) Latch 1 Time Negative Edge Register (DC_LATCH1_TIME_NEG)

This register indicates the system time at the falling edge of the latch 1 input signal.

Address(es): A00D 09C8h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SYSTIME															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	SYSTIME	System Time Indication	Indicate the system time captured at the falling edge of the latch 1 input signal. Reading this register clears bit 1 of the latch 1 status register (DC_LATCH1_STAT at 09AFh).	R (ack)	R (ack)

Bits 63 to 8 are internally latched (ECAT/PDI independently) when bits 7 to 0 are read, which guarantees reading a consistent value.

Clearing the latch 1 status flag function depends on the setting of bit 5 of the cyclic unit control register (DC_CYC_CONT at 0980h).

22.3.16.6 SyncManager Event Time Registers

(1) Buffer Change Event Time Register (DC_ECAT_CNG_EV_TIME)

This register indicates the local time at the beginning of a frame which causes SyncManager to generate an ECAT event (switching the buffers).

Address(es): A00D 09F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECATCHANGE															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECATCHANGE															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	ECATCHANGE	Local Time Indication	Indicate the local time at the beginning of a frame which causes at least one SyncManager to generate an ECAT event (switching the buffers).	R	R

Bits 31 to 8 are internally latched (ECAT/PDI independently) when bits 7 to 0 are read, which guarantees reading a consistent value.

(2) PDI Buffer Start Event Time Register (DC_PDI_START_EV_TIME)

This register indicates the local time when SyncManager has generated a PDI event (access to the address where a buffer starts).

Address(es): A00D 09F8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PDISTART															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PDISTART															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	PDISTART	Local Time Indication	Indicate the local time when at least one SyncManager has generated a PDI event (access to the address where a buffer starts).	R	R

Bits 31 to 8 are internally latched (ECAT/PDI independently) when bits 7 to 0 are read, which guarantees reading a consistent value.

(3) PDI Buffer Change Event Time Register (DC_PDI_CNG_EV_TIME)

This register indicates the local time when SyncManager has generated a PDI event (switching the buffers).

Address(es): A00D 09FCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PDICHANGE															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PDICHANGE															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	PDICHANGE	Local Time Indication	Indicate the local time when at least one SyncManager has generated a PDI event (switching the buffers).	R	R

Bits 31 to 8 are internally latched (ECAT/PDI independently) when bits 7 to 0 are read, which guarantees reading a consistent value.

22.3.17 Other Registers

22.3.17.1 Product ID Register (PRODUCT_ID)

This register indicates the product ID.

Address(es): A00D 0E00h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	PROID															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	PROID															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PROID															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PROID															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	PDI	ECAT
b63 to b0	PROID	Product ID Indication	Product ID	R	R

22.3.17.2 Vendor ID Register (VENDOR_ID)

This register indicates the vendor ID.

Address(es): A00D 0E08h

	b63	b62	b61	b60	b59	b58	b57	b56	b55	b54	b53	b52	b51	b50	b49	b48
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b47	b46	b45	b44	b43	b42	b41	b40	b39	b38	b37	b36	b35	b34	b33	b32
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	VENDORID															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	VENDORID															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	PDI	ECAT
b31 to b0	VENDORID	Vendor ID Indication	Vendor ID	R	R
b63 to b32	—	Reserved	When read, the value returned is undefined.	R	R

22.3.17.3 User RAM (USER_RAM)

This area of RAM indicates the supported features dependent on the IP core configuration, and takes up the 128 bits from A00D 0F80h to A00D 0FFFh. An initial value of 1 means that the implementation of this module supports the corresponding feature, except in the case of bits 7 to 0, which indicate the number of bits defined in the user RAM and the initial value is 33h for this module.

Bit Position	Description	Initial Value
7 to 0	Number of bits for defining extended functionality. The value is 51 (33h) for this module.	33h
8	Extended DL control register (0102h, 0103h)	1
9	AL status code register (0134h, 0135h)	1
10	ECAT event mask (0200h, 0201h)	1
11	Configured station alias (0012h, 0013h)	1
12	General input (0F18h, 0F1Fh)	0
13	General output (0F10h, 0F17h)	0
14	AL event mask (0204h, 0207h)	1
15	Physical read/write offset (0108h, 0109h)	1
16	Watchdog divider writable (0400h, 0401h) and watchdog PDI (0410h, 0F11h)	1
17	Watchdog counter (0442h, 0443h)	1
18	Write protection (0020h, 0031h)	1
19	Reset (0040h, 0041h)	1
20	Reserved	0
21	DC SyncManager event time (09F0h, 09FFh)	1
22	ECAT processing unit/PDI error counter (030Ch, 030Dh)	1
23	EEPROM size configurable (Bit 7 at 0502h) 0: EEPROM size fixed up to 16 Kbits 1: EEPROM Size configurable	1
26 to 24	Reserved	0
27	Lost link counter (0310h, 0313h)	1
28	MII management interface (0510h, 0515h)	1
29	Enhanced link detection MII	1
30	Enhanced link detection EBUS	0
31	Run LED	1
32	Link/activity LED	1
33	Reserved	0
35 to 34	Reserved	1
36	Reserved	0
37	Reserved	1
38	DC Time loop control assigned to PDI	0
39	Link detection and configuration by MI	0
40	MI control by PDI	1
41	Automatic TX shift	1
42	EEPROM emulation	0
49 to 43	Reserved	0
50	ERR LED, RUN/ERR LED override	1
Others	Reserved	Reserved

22.3.17.4 Process Data RAM (DATA_RAM)

The process data RAM is used for process data and mailbox, and takes up 8 Kbytes from A00D 1000h to A00D 2FFFh. This RAM is only accessible when the EEPROM is correctly loaded (i.e., when bit 0 is 1 in the ESC DL status register, ESC_DL_STATUS, at 0110h).

22.4 Notes on Usage

22.4.1 Protect Command Register

If a program runs out of control, the application might stop unexpectedly. To prevent such a problem, protect command registers are used to protect registers from write operations that might significantly affect the system. ETHERC has the following protect command registers:

- System protect command register (SPCMD)
- Ethernet system protect command register (ETSPCMD)

A write operation for a protected register can be performed only when the register is unlocked by setting the protection unlock enable bit of the SPCMD or ETSPCMD protect command register to ON (1).

To set a protect command register (to 1), only write operations performed in the following specific sequence are accepted. There is no special sequence for clearing the register (to 0) or reading it.

(1) Write 0000 00A5h to the protect command register as a specific value.
(2) Write 0000 00001h to the protect command register as an expected value.
(3) Write 0000 FFFEh to the protect command register as a reversal value.
(4) Write 0000 0001h to the protect command register as an expected value.

Note 1. In steps (2) and (3), no write operation is performed for the target register.
Note 2. After a write operation for the target register is completed, make sure that you clear the protect unlock enable bit (to 0) to enable protection.

Figure 22.2 shows the state transitions.

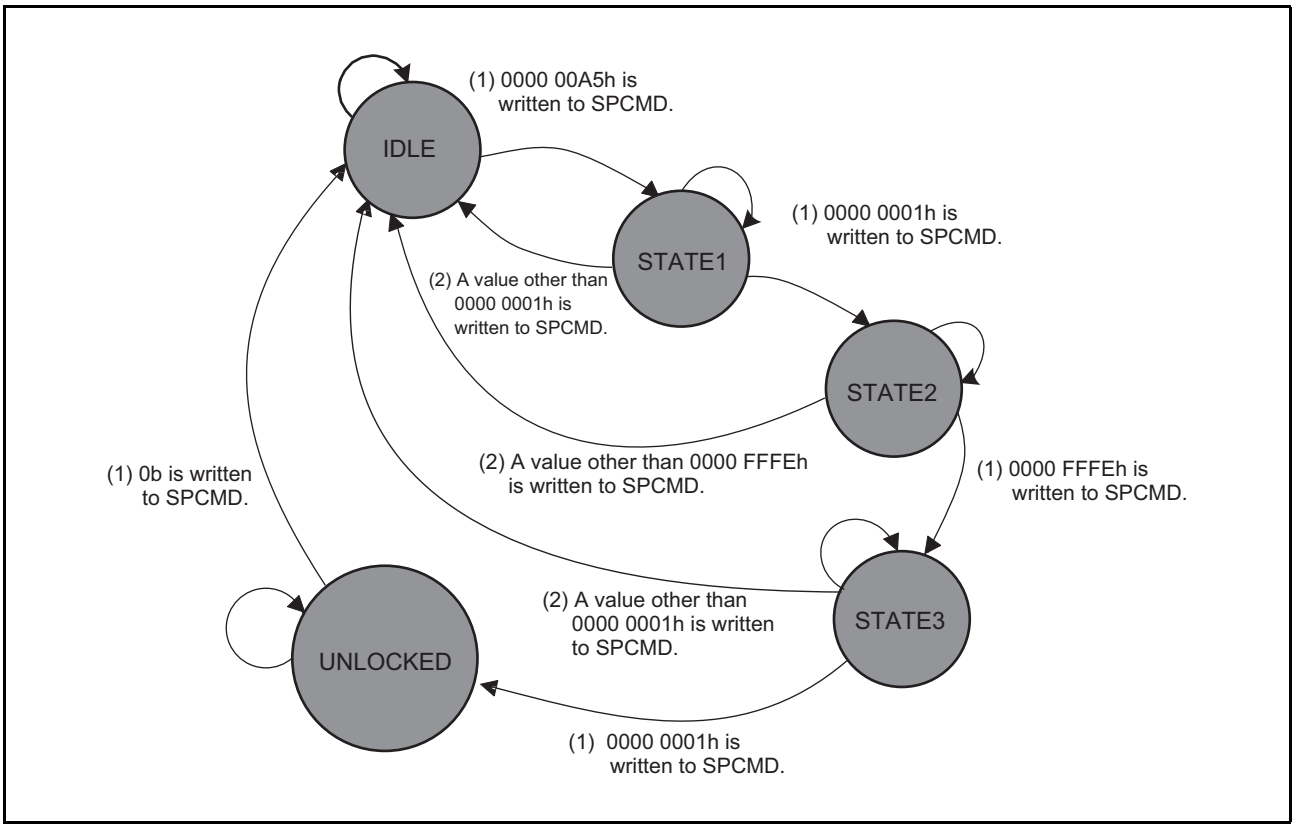


Figure 22.2 State Transition Diagram for a Protect Command Register (Example for the SPCMD Register)

22.4.2 Setting the Module-Stop Function

The modules associated with the EtherCAT slave controller are stopped in the initial state. If the modules are to be used, set the MSTPCRB.MSTPCRB15, MSTPCRB16, MSTPCRB18, and MSTPCRB19 bits for release from the module-stop state. Note that re-setting the module-stop state is prohibited following release from the module-stop state. Operation of the modules if they are stopped following release and then released again is not guaranteed. Release from the module-stop state can proceed again after the modules have been returned to their initial states (stopped) by a reset.

22.4.3 Initial Settings

To initialize the EtherCAT, follow the procedure below.

- Set the offset address of the PHY module in the CATOFFADD register.
- Set the size of the EEPROM in the CATEMMD register.
- Set the delay time of the TXC pin in the CATTXCSFT register.
- Release the EtherCAT from the module stop state by using the MSTPCRB15 bit in the MSTPCRB register.
- Release the ESC and PHY modules from the reset state by using the CATRST and PHYRST bits in the ETHSFTRST register.

On release from the reset state, the ESC will automatically load the EEPROM data and be activated.

22.4.4 Configuration of the Reset Circuit

Figure 22.3 shows the configuration of the reset circuit of the ESC. On reception of a reset request (0040h) from the ECAT or a reset request (0041h) from the PDI, the ESC stops and the output of the reset signal from the ESC becomes high. The output of this reset signal from the ESC drives the signal on the PHYRESETOUT# pin to the low level and the externally connected Ethernet PHY chip is reset. An ETHCRSTI interrupt is generated at the same time.
To release the ESC from the reset state, the CATRST bit in the ETHSFTRST register must be switched from 1 to 0 then back to 1. Note that the output of the reset signal from the ESC becomes low when the input of the reset signal to the ESC changes from high to low. Restarting of the ESC begins when the input of the reset signal to the ESC changes from low to high and loading of the EEPROM starts. Loading of the EEPROM is completed in about 1 ms. The timing with which the Ethernet PHY chip is released from the reset state must be set so that it will only recommence operations after the ESC has started. Figure 22.4 shows the timing chart.
Resetting the ESC by using the CATRST bit of the ETHSFTRST register instead of by the ECAT or PDI issuing a reset request (0040h or 0041h) is also possible. In this case, the PHYRESETOUT# pin does not automatically go to the low level, so use the PHYRST bit of the ETHSFTRST register to place the Ethernet PHY chip in the reset state beforehand. Figure 22.5 shows the timing chart.

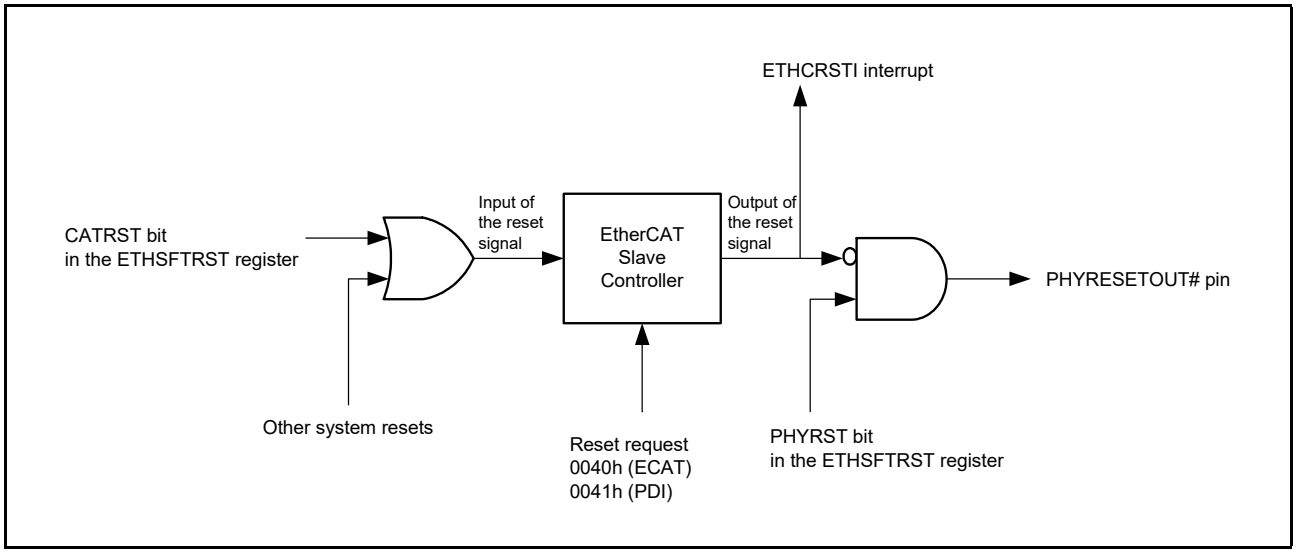


Figure 22.3 Configuration of the Reset Circuit of the EtherCAT Slave Controller

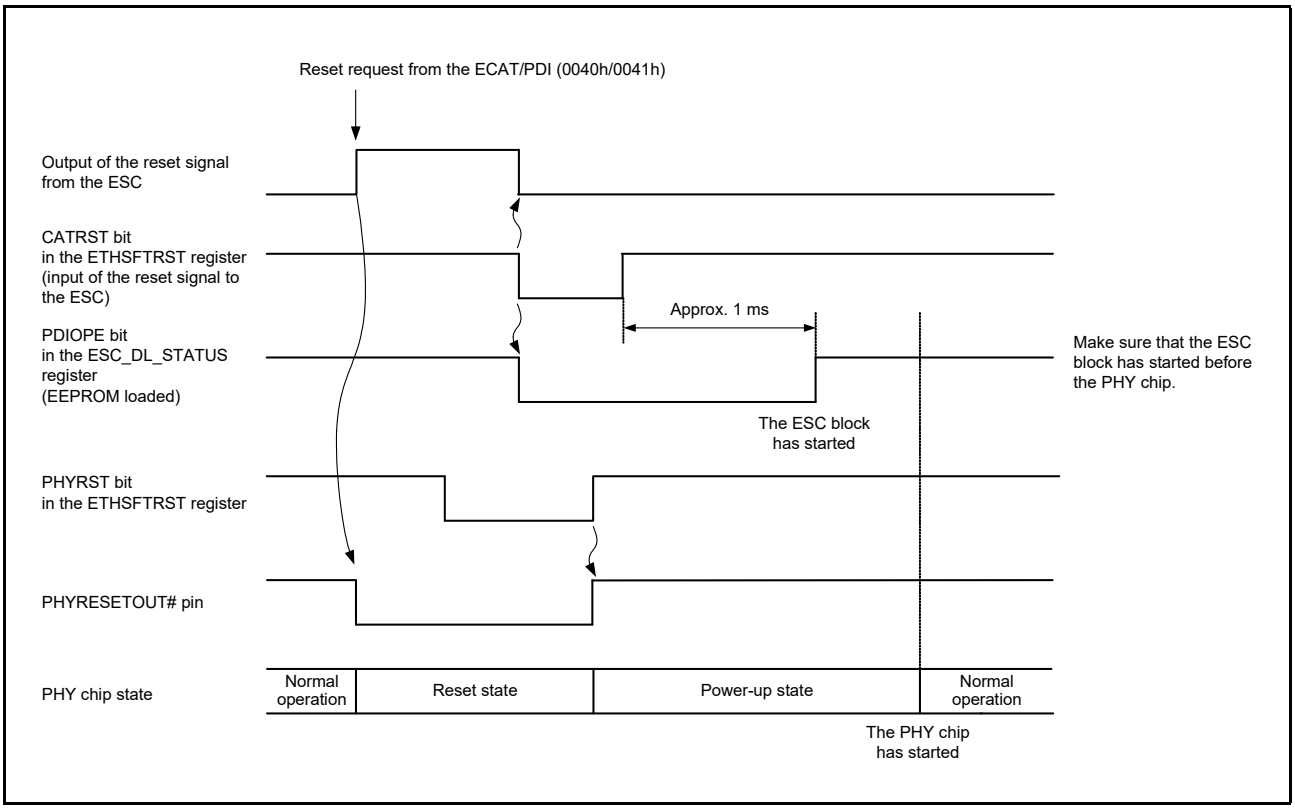


Figure 22.4 Timing with which the EtherCAT Slave Controller is Reset (in the Case of a Reset Requested by the ECAT or PDI)

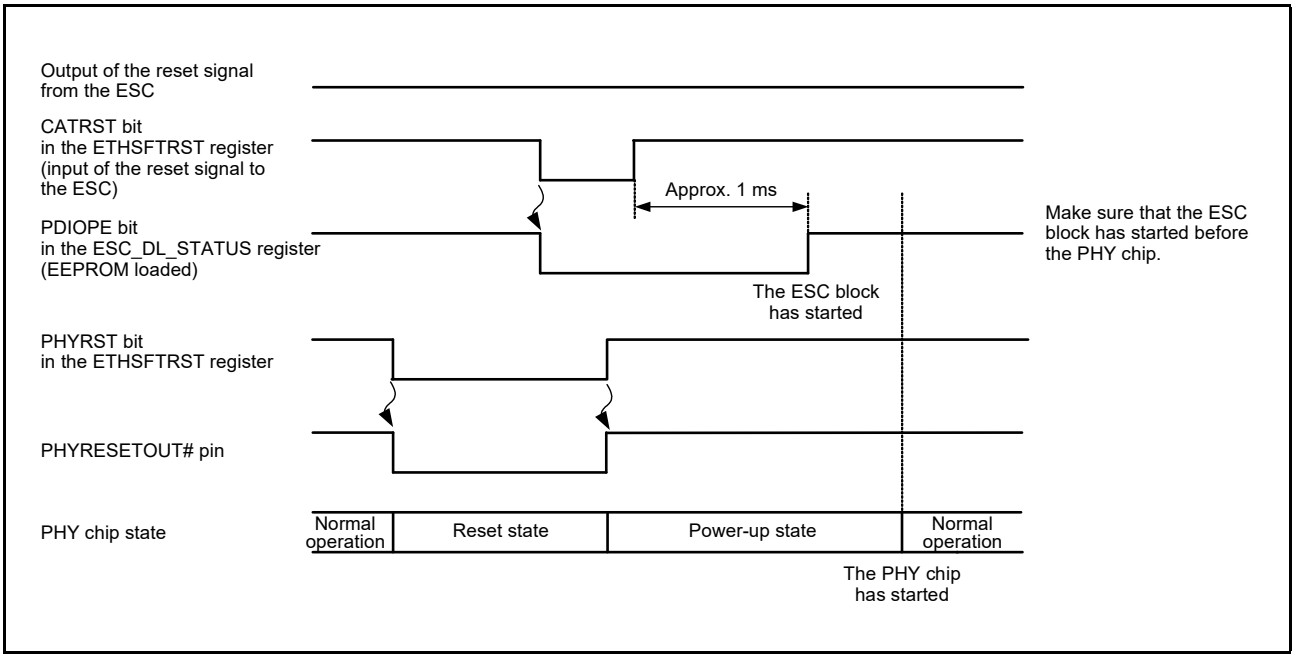


Figure 22.5 Timing with which the EtherCAT Slave Controller is Reset (in the Case of the CATRST Bit of the ETHSFTRST Register being Used to Reset the ESC)

23. USB2.0HS Host Module (USBh)

23.1 Overview

The USB module of this LSI is a dual-role device that has USB 2.0 host and function modules. However, it does not have a facility for detecting the ID and so does not support On-the-Go (OTG) functionality.
This LSI handles a single USB port for either host controller or function controller operation. The port connection path select input signal bits (PHYSET1.P1PORTSEL[1:0]) are used to switch between host controller and function controller operation.
Since operation as a host controller or a function controller are exclusive of each other, dynamic switching between the types of operation is not possible.

This section describes host controller operation.

USB2.0 Host Controller Operation

- Conforms to Universal Serial Bus Specification Revision 2.0.
- Conforms to Open Host Controller Interface (OHCI) Specification for USB Rev 1.0a.
- Conforms to Enhanced Host Controller Interface (EHCI) Specification for USB Rev 1.0.
- Supports USB2.0 high-speed (480 Mbps) and full-speed (12 Mbps) transfer.

Note: Low-speed (1.5 Mbps) is not supported.

- Supports the USB2.0 compliance test function.

Figure 23.1 is a block diagram of the USB module.

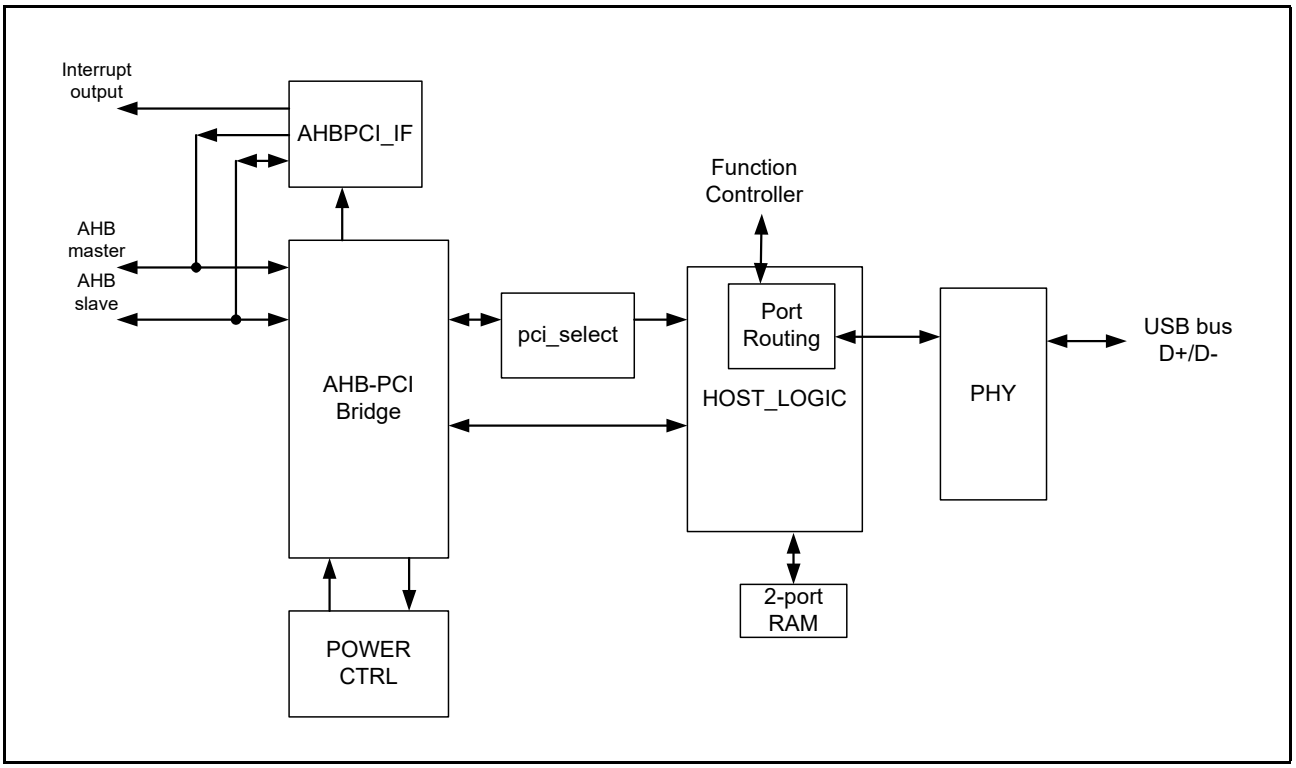


Figure 23.1 Block Diagram of the USB Module

(1) Host Logic

A USB 2.0 logic conforming to the EHCI and OHCI standards. It has control circuits such as a list processing circuit, a serial-parallel circuit, a USB buffer, and executes high-speed, and full-speed USB transfer.

(2) UTMI+ Transceiver

A USB 2.0 UTMI transceiver with an interface conforming to the UTMI+ standard.

(3) AHB-PCI Bridge

A module for conversion between AHB bus cycles and PCI bus cycles to the hot logic. Access from the CPU to registers in the bridge or host logic is performed through the slave interface in the bridge. Access by the host logic with the PCI as the master is transferred to the AHB bus through the master interface in the bridge.

23.1.1 Precautions on Use of USB Host Controller

The following describes precautions on use of the USB host controller.

23.1.1.1 General Precautions

(1) Dynamic changes in PCLKA, USBPCLK, and USBMCLK are not supported except for the clock stop state.
(2) When using isochronous transfer, evaluate performance at the application level to confirm that the expected performance is obtained.
(3) The USB host controller outputs the logical OR of four internal interrupt signals. See section 23.6, Interrupts.
(4) It may take a long time to actually clear an interrupt after a register is accessed to clear the interrupt. For the actionnecessary during this wait time, see section 23.6.3, Time Required to Clear Interrupt Signals.

23.1.1.2 AHB Interface

(1) Only 32-bit access is available for AHB slave access to the USB host controller; 8- or 16-bit access is not allowed.
(2) When using the wait mode with HRESP = RETRY, access the following mode switch register in the AHB bridge first.
- Bit 17 (SMODE_READY_CTR) in the AHB_BUS_CTR register
(3) The read or written data is not guaranteed for register access while the clock is stopped.

23.1.1.3 Operating Procedures

(1) See the operating procedures described in the following sections.
- Initial settings: section 23.8, Operating Procedures
- Mapping in the AHB space and PCI space: section 23.4.1, Register Access.

23.2 Register Mapping

23.2.1 Register Mapping

The register space is broadly divided into the following three areas.

1. OHCI/EHCI operational register area
2. PCI configuration space area
3. AHB-PCI bridge PCI communication area

To access each PICI configuration space, the AHPBPCI_WIN1_CTR register in the AHB-PCI bridge register area should be used. In addition to register mapping in the AHB space, the addresses of the OHCI/EHCI operational registers and PCI communication spaces should be correctly mapped to the PCI space in the USB host controller. For details of access and address mapping of each register, see section 23.4.1, Register Access.
The range of address input is from A004 0000h to A005 FFFFh. Do not access the reserved area (A005 0C00h to A005 FFFFh).

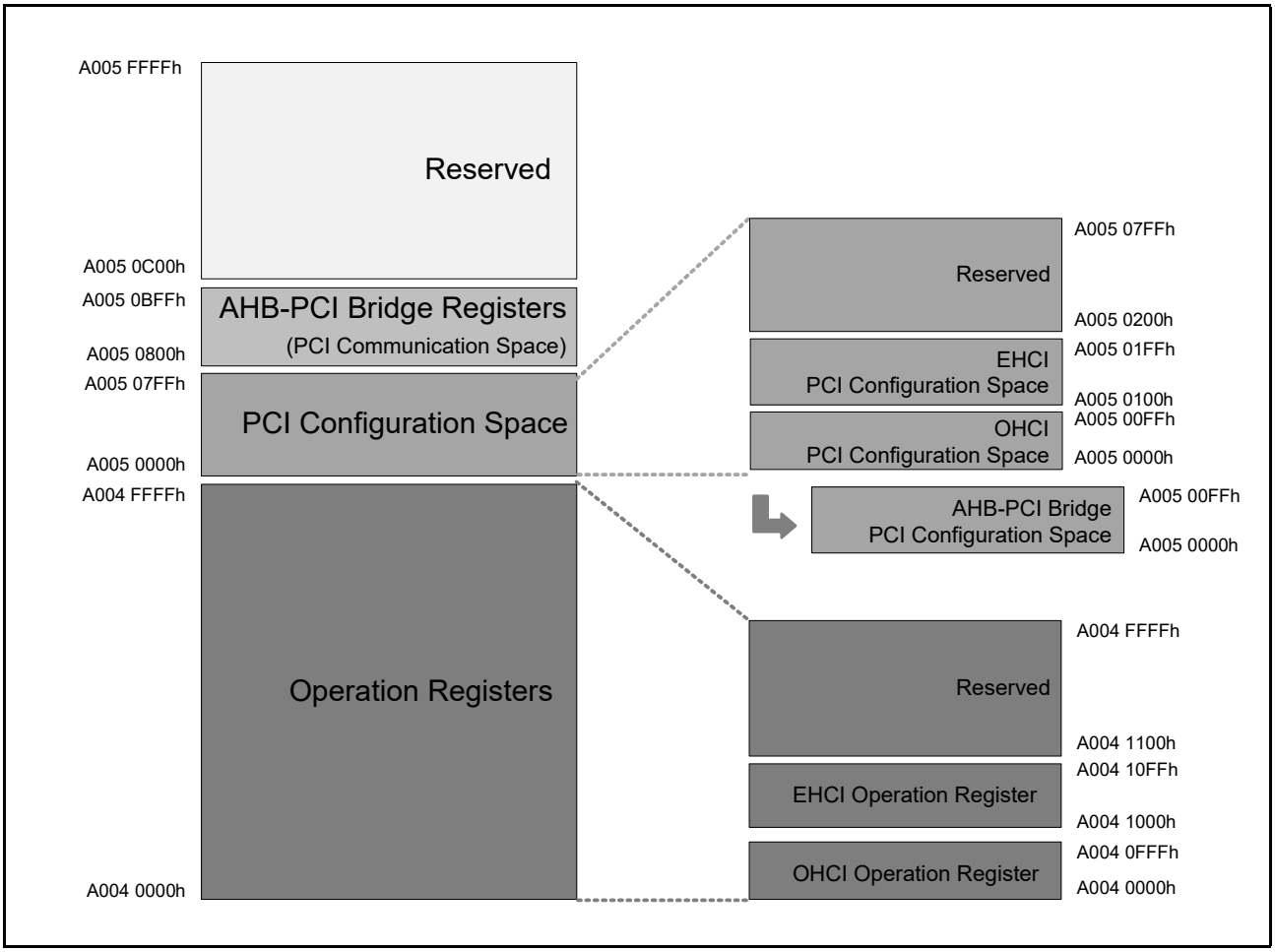


Figure 23.2 Register Mapping Image

Table 23.1 Register Mapping List (1 / 2)

Address	Register Name	Symbol
A004 0000h	HcRevision	HcRevision
A004 0004h	HcControl	HcControl
A004 0008h	HcCommandStatus	HcCommandStatus
A004 000Ch	HcInterruptStatus	HcIntStatus
A004 0010h	HcInterruptEnable	HcIntEnable
A004 0014h	HcInterruptDisable	HcIntDisable
A004 0018h	HcHCCA	HcHCCA
A004 001Ch	HcPeriodicCurrentED	HcPeriodCurED
A004 0020h	HcControlHeadED	HcContHeadED
A004 0024h	HcControlCurrentED	HcContCurrentED
A004 0028h	HcBulkHeadED	HcBulkHeadED
A004 002Ch	HcBulkCurrentED	HcBulkCurrentED
A004 0030h	HcDoneHead	HcDoneHead
A004 0034h	HcFmInterval	HcFmInterval
A004 0038h	HcFmRemaining	HcFmRemaining
A004 003Ch	HcFmNumber	HcFmNumber
A004 0040h	HcPeriodicStart	HcPeriodicStart
A004 0044h	Reserved	—
A004 0048h	HcRhDescriptorA	HcRhDescriptorA
A004 004Ch	HcRhDescriptorB	HcRhDescriptorB
A004 0050h	HcRhStatus	HcRhStatus1_A, HcRhStatus1_B
A004 0054h	HcRhPortStatus1	HcRhPortStatus1_A, HcRhPortStatus1_B
A004 0058Ch to A004 0FFCh	Reserved	—
A004 1000h	HCIVERSION / CAPLENGTH	CAPL_VERSION
A004 1004h	HCSPARAMS	HCSPARAMS
A004 1008h	HCCPARAMS	HCCPARAMS
A004 100Ch	HCSP_PORTROUTE	HCSP_PORTROUTE
A004 1010h to A004 101Ch	Reserved	—
A004 1020h	USBCMD	USBCMD
A004 1024h	USBSTS	USBSTS
A004 1028h	USBINTR	USBINTR
A004 102Ch	FRINDEX	FRINDEX
A004 1030h	CTRLDSSEGMENT	CTRLDSSEGMENT
A004 1034h	PERIODICLISTBASE	PERIODICLIST
A004 1038h	ASYNCLISTADDR	ASYNCLISTADDR
A004 103Ch to A004 105Ch	Reserved	—
A004 1060h	CONFIGFLAG	CONFIGFLAG
A004 1064h	PORTSC1	PORTSC1
A004 1068h to A004FFFCh	Reserved	—
A005 0000h to A005 07FCh	PCI Configuration Space (AHB-PCI Bridge / OHCI / EHCI)	—
A005 0800h	PCIAHB_WIN1_CTR	PCIAHB_WIN1_CTR
A005 0804h	Reserved	—
A005 0808h to A005 080Ch	Reserved	—
A005 0810h	AHBPCI_WIN1_CTR	AHBPCI_WIN1_CTR

Table 23.1 Register Mapping List (2 / 2)

Address	Register Name	Symbol
A005 0814h	AHBPCI_WIN2_CTR	AHBPCI_WIN2_CTR
A005 0818h to A005 081Ch	Reserved	—
A005 0820h	PCI_INT_ENABLE	PCI_INT_ENABLE
A005 0824h	PCI_INT_STATUS	PCI_INT_STATUS
A005 0828h to A005 082Ch	Reserved	—
A005 0830h	AHB_BUS_CTR	AHB_BUS_CTR
A005 0834h	USBCTR	USBCTR
A005 0838h to A005 083Ch	Reserved	—
A005 0840h	PCI_ARBITER_CTR	PCI_ARBITER_CTR
A005 0844h	Reserved	—
A005 0848h	PCI_UNIT_REV	PCI_UNIT_REV
A005 084Ch to A005 FFFCh	Reserved	—

23.2.2 PCI Configuration Space for AHB-PCI Bridge

Table 23.2 shows the register mapping in the PCI configuration space for the AHB-PCI bridge.

Table 23.2 PCI Configuration Space For AHB-PCI Bridge

Offset	31 24	23 16	15 8	7 0	Symbol
000h	Device ID		Vendor ID		VID_DID_A
004h	Status		Command		CMND_STS_A
008h	Class Code			Revision ID	REVID_CC_A
00Ch	BIST	Header Type	Latency Timer	Cache Line Size	CLS_LT_HT_BIST_A
010h	AHB-PCI Bridge Registers Base Address				BASEAD_A
014h	PCI-AHB Window1 Base Address				WIN1_BASEAD
01Ch	Reserved				—
020h					
024h					
028h					
02Ch	Subsystem ID		Subsystem Vendor ID		SSVID_SSID_A
030h	Reserved				—
034h					
038h					
03Ch	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	INTR_LINE_PIN_A
040h	Reserved				—
|					
0FCh					

23.2.3 PCI Configuration Space for OHCI Host Logic

Table 23.3 shows the register mapping in the PCI configuration space for the host logic (OHCI).

Table 23.3 PCI Configuration Space for OHCI

Offset	31 24	23 16	15 8	7 0	Symbol
000h	Device ID		Vendor ID		VID_DID_O
004h	Status		Command		CMND_STS_O
008h	Class Code			Revision ID	REVID_CC_O
00Ch	BIST	Header Type	Latency Timer	Cache Line Size	CLS_LT_HT_BIST_O
010h	OHCI Base Address				BASEAD_O
014h	Reserved				—
018h					
01Ch					
020h					
024h					
028h					
02Ch	Subsystem ID		Subsystem Vendor ID		SSVID_SSID_O
030h	Expansion ROM Base Address				EROM_BASEAD
034h	Reserved			Cap_ptr	CAPPTR
038h	Reserved				—
03Ch	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	INTR_LINE_PIN_O
040h	PMC		Next_Item_Ptr	Cap_ID	CAPID_NIP_PMCAP
044h	Data	PMCSR_BSE	PMCSR		PMC_STS_PMCSR
048h to 0DCh	Reserved				—
0E0h	EXT1				EXT1
0E4h	EXT2				EXT2
0E8h to 0ECh	Reserved				—
0F0h	Reserved				—
0F4h	Reserved				—
0F8h to 0FCh	Reserved				—

23.2.4 PCI Configuration Space for EHCI Host Logic

Table 23.4 shows the register mapping in the PCI configuration space for the host logic (EHCI).

Table 23.4 PCI Configuration Space for EHCI

Offset	31 24	23 16	15 8	7 0	Symbol
100h	Device ID		Vendor ID		VID_DID_E
104h	Status		Command		CMND_STS_E
108h	Class Code			Revision ID	REVID_CC_E
10Ch	BIST	Header Type	Latency Timer	Cache Line Size	CLS_LT_HT_BIST_E
110h	EHCI Base Address				BASEAD_E
114h	Reserved				—
118h					
11Ch					
120h					
124h					
128h					
12Ch	Subsystem ID		Subsystem Vendor ID		SSVID_SSID_E
130h	Expansion ROM Base Address				EROM_BASEAD_E
134h	Reserved			Cap_ptr	CAPPTR_E
138h	Reserved				—
13Ch	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	INTR_LINE_PIN_E
140h	PMC		Next_Item_Ptr	Cap_ID	CAPID_NIP_PMCAP_E
144h	Data	PMCSR_BSE	PMCSR		PMC_STS_PMCSR_E
148h	Reserved				—
15Ch					
160h	PORTWAKECAP		FLAD	SBRN	SBRN_FLADJ_PW
164h to 1DCh	Reserved				—
1E0h	EXT1				EXT1_E
1E4h	EXT2				EXT2_E
1E8h to 1ECh	Reserved				—
1F0h	Reserved				—
1F4h	Reserved				—
1F8h to 1FCh	Reserved				—

23.3 Register Descriptions

This section describes details of the register functions.

23.3.1 OHCI Operational Registers

Access the OHCI operational registers after starting up the PHY internal PLL. For details, see Figure 23.13, Initial Setting Sequence.

23.3.1.1 HcRevision Register

Address(es) A004 0000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	Revision[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Revision[7:0]	HCI revision	These bits indicate the version of the HCI specifications implemented in the host logic. As the host logic conforms to OHCI standard 1.0a, this value is set to 10h.	R
b31 to b8	—	Reserved	Don't care	R

23.3.1.2 HcControl Register

Address(es) A004 0004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	RWE	RWC	—	HCFS[1:0]		BLE	CLE	IE	PLE	CBSR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CBSR[1:0] (ControlBulk ServiceRatio)	Control / bulk transfer service ratio specifying	These bits specify the service ratio of control transfer to bulk transfer. In periodic list processing, the service ratio specified in these bits is kept during transfer. **CBSR** / **Service Ratio of Bulk ED : Control ED** 00 — 1:1 01 — 2:1 10 — 3:1 11 — 4:1	R/W
b2	PLE (PeriodicList Enable)	Periodic list setting	This bit specifies whether to perform periodic list processing. The setting of this bit takes effect from the next frame. 0: Periodic list processing is not done. 1: Period list processing is done.	R/W
b3	IE (Isochronous Enable)	Isochronous ED processing setting	This bit specifies whether to perform isochronous ED processing. The setting of this bit takes effect from the next frame. When an isochronous ED is found during list processing, the host logic checks this bit to determine whether to perform isochronous ED processing. 0: Isochronous transfer processing is not done. 1: Isochronous transfer processing is done.	R/W
b4	CLE (ControlListEnable)	Control list processing setting	This bit specifies whether to perform control list processing. The setting of this bit takes effect from the next frame. The control list can be modified only when this value is 0. 0: Control list processing is not done. 1: Control list processing is done.	R/W
b5	BLE (BulkListEnable)	Bulk list processing setting	This bit specifies whether to perform bulk list processing. The setting of this bit takes effect from the next frame. The bulk list can be modified only when this value is 0. 0: Bulk list processing is not done. 1: Bulk list processing is done.	R/W
b7, b6	HCFS[1:0] (Host Controller FunctionalState)	Host logic operation status	These bits indicate the operating state of the host logic. Upon entering the USB operational state, the host logic starts management of frames delimited at 1-ms intervals. The operating state is always controlled by software except for transition to the USB resume state due to remote wakeup in the USB suspend state. These bits indicate the USB reset state after a hard ware reset. After a software reset, the state shifts to the USB suspend state. b7 b6 0 0: USB Reset 0 1: USB Resume 1 0: USB Operational 1 1: USB Suspend	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	—	Reserved	When writing, write 0.	R/W
b9	RWC (RemoteWakeUpConnect)	Remote WakeUp support setting	This bit indicates whether the host logic supports remote wakeup. To support remote wakeup by the system, set this bit during initialization. 0: Remote WakeUp is not supported. 1: Remote WakeUp is supported.	R/W
b10	RWE (RemoteWakeUpEnable)	PME assertion control	This bit controls PME assertion. When this bit is 1, PME is asserted when bit 3 (RD) in the HcInterruptStatus register is set to 1. 0: PME is not asserted when Resume is detected (PME is disabled). 1: PME is asserted when Resume is detected (PME is enabled).	R/W
b31 to b11	—	Reserved	When writing, write 0.	R/W

23.3.1.3 HcCommandStatus Register

Address(es) A004 0008h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SOC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	OCR	BLF	CLF	HCR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	HCR (HostController Reset)	Host logic software reset start	This bit starts software reset of the host logic. When this bit is set, the USB suspend state is entered regardless of the functional state of the host logic.	W
b1	CLF (ControlList Filled)	Control list TD	This bit indicates whether TD exists in the control list. The host logic checks this bit when beginning the processing of the start ED in the control list. When this bit is 0, the host logic does not begin list processing. When this bit is 1, the host logic starts control list processing and clears this bit to 0. Upon detecting TD, the host logic restores this bit to 1 and continues control list processing. When the host logic completes list processing, it clears this bit to 0. If TD is not found in the list or software does not set this bit to 1, this bit remains at 0 and list processing stops. When creating a list again and executing processing of the list, set this bit before starting list processing by setting bit 4 (CLE) in the HcCommand register.	R/W
b2	BLF (BulkListFilled)	Bulk list TD	This bit indicates whether TD exists in the bulk list. The host logic checks this bit when beginning the processing of the start ED in the bulk list. When this bit is 0, the host logic does not begin list processing. When this bit is 1, the host logic starts bulk list processing and clears this bit to 0. Upon detecting TD, the host logic restores this bit to 1 and continues bulk list processing. When the host logic completes list processing, it clears this bit to 0. If TD is not found in the list or software does not set this bit to 1, this bit remains at 0 and list processing stops. When creating a list again and executing processing of the list, set this bit before starting list processing by setting bit 5 (BLE) in the HcCommand register.	R/W
b3	OCR (Ownership Change Request)	Host logic control right change	This bit changes the control right of the host logic. (For details, check with the OHCI specification).	W
b15 to b4	—	Reserved	When writing, write 0.	R/W
b17, b16	SOC[1:0] (Scheduling OverrunCount)	Schedule overrun count	These bits count the number of times a schedule overrun occurs. This value is incremented every time a schedule overrun occurs. When the value reaches 11b, it returns to 00b. Counting continues even while bit 0 (SO) in the HcInterruptStatus register is set.	R
b31 to b18	—	Reserved	When writing, write 0.	R/W

23.3.1.4 HcInterruptStatus Register

Address(es) A004 000Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	RHSC	FNO	UE	RD	SF	WDH	SO
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SO (Scheduling Overrun)	USB schedule overrun	This is an interrupt bit for indicating that the USB schedule has overrun in a frame. When the USB schedule overruns, this bit is set after HccaFrameNumber for the next frame is updated. When this bit is set, bits [17:16] (SOC) in the HcCommandStatus register are also incremented. Writing 1 to this bit clears the interrupt. 0: No SO interrupt has occurred. 1: SO interrupt has occurred.	R/W
b1	WDH (Writeback Done Head)	Host logic HccaDoneHead update	This is an interrupt bit for indicating that the host logic has updated the contents of HccaDoneHead. The host logic sets this bit immediately after updating HccaDoneHead, and does not update HccaDoneHead until this bit is cleared. Writing 1 to this bit clears the interrupt. 0: No WDH interrupt has occurred. 1: WDH interrupt has occurred.	R/W
b2	SF (StartOfFrame)	HccaFrameNumber update	This is an interrupt bit for indicating that HccaFrameNumber has been updated at the beginning of a frame. The host logic updates HccaFrameNumber and sets this bit when transmitting an SOF packet. Writing 1 to this bit clears the interrupt. 0: No SF interrupt has occurred. 1: SF interrupt has occurred.	R/W
b3	RD (Resume Detected)	Resume detection	This is an interrupt bit for indicating that Resume has been detected. This bit is set when assertion of the Resume signal by a device on the USB is detected. This bit is not set when software issues USB Resume. Writing 1 to this bit clears the interrupt. 0: No RD interrupt has occurred. 1: RD interrupt has occurred.	R/W
b4	UE (Unrecoverable Error)	USB Non-related system error detection	This is an interrupt bit for indicating that a system error on the PCI bus that is not related to the USB has been detected. Writing 1 to this bit clears the interrupt. 0: No UE interrupt has occurred. 1: UE interrupt has occurred.	R/W
b5	FNO (Frame Number Overflow)	FrameNumber bit MSB change	This is an interrupt bit for indicating that the MSB of bits [15:0] (FrameNumber) in the HcFmNumber register has changed. This bit is set after HccaFrameNumber is updated in the frame where the MSB of the FrameNumber bits changes from 0 to 1 or from 1 to 0. Writing 1 to this bit clears the interrupt. 0: No FNO interrupt has occurred. 1: FNO interrupt has occurred.	R/W

Bit	Symbol	Bit Name	Description	R/W
b6	RHSC (RootHubStatus Change)	HcRhStatus/HcRhPortStatus register status	This is an interrupt bit for indicating that the state of the HcRhStatus register or the HcRhPortStatus register has changed. This bit is set when the HcRhStatus or HcRhPortStatus register state is changed by a hardware source. Writing 1 to this bit clears the interrupt. 0: No RHSC interrupt has occurred. 1: RHSC interrupt has occurred.	R/W
b31 to b7	—	Reserved	When writing, write 0.	R/W

23.3.1.5 HcInterruptEnable Register

Address(es) A004 0010h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	MIE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	RHSCE	FNOE	UEE	RDE	SFE	WDHE	SOE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SOE (Scheduling OverrunEnable)	SO interrupt source enable	This bit enables SO as an interrupt source. To set this bit, write 1 to it. To clear this bit, write 1 to the corresponding bit in the HcInterruptDisable register. 0: No monitoring and nothing is to be done. 1: SO interrupt is enabled.	R/W
b1	WDHE (WritebackDone HeadEnable)	WDH interrupt source enable	This bit enables WDH as an interrupt source. To set this bit, write 1 to it. To clear this bit, write 1 to the corresponding bit in the HcInterruptDisable register. 0: No monitoring and nothing is to be done. 1: WDH interrupt is enabled.	R/W
b2	SFE (StartOfFrame Enable)	SF interrupt source enable	This bit enables SF as an interrupt source. To set this bit, write 1 to it. To clear this bit, write 1 to the corresponding bit in the HcInterruptDisable register. 0: No monitoring and nothing is to be done. 1: SF interrupt is enabled.	R/W
b3	RDE (Resume DetectedEnable)	RD interrupt source enable	This bit enables RD as an interrupt source. To set this bit, write 1 to it. To clear this bit, write 1 to the corresponding bit in the HcInterruptDisable register. 0: No monitoring and nothing is to be done. 1: RD interrupt is enabled.	R/W
b4	UEE (Unrecoverable ErrorEnable)	UE interrupt source enable	This bit enables UE as an interrupt source. To set this bit, write 1 to it. To clear this bit, write 1 to the corresponding bit in the HcInterruptDisable register. 0: No monitoring and nothing is to be done. 1: UE interrupt is enabled.	R/W
b5	FNOE (FrameNumber OverflowEnable)	FNO interrupt source enable	This bit specifies FNO as an interrupt source. To set this bit, write 1 to it. To clear this bit, write 1 to the corresponding bit in the HcInterruptDisable register. 0: No monitoring and nothing is to be done. 1: FNO interrupt is enabled.	R/W
b6	RHSCE (RootHubStatus ChangeEnable)	RHSC interrupt source enable	This bit enables RHSC as an interrupt source. To set this bit, write 1 to it. To clear this bit, write 1 to the corresponding bit in the HcInterruptDisable register. 0: No monitoring and nothing is to be done. 1: RHSC interrupt is enabled.	R/W
b30 to b7	—	Reserved	When writing, write 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b31	MIE (MasterInterrupt Enable)	Interrupt 7 source enable	This bit enables interrupt source settings in bits [6:0] in this register. When this bit is 0, all interrupts are masked. To clear this bit, write 1 to the corresponding bit in the HcInterruptDisable register. 0: No monitoring and nothing is to be done. 1: The specified interrupt is enabled.	R/W

23.3.1.6 HcInterruptDisable Register

Address(es) A004 0014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	MID	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

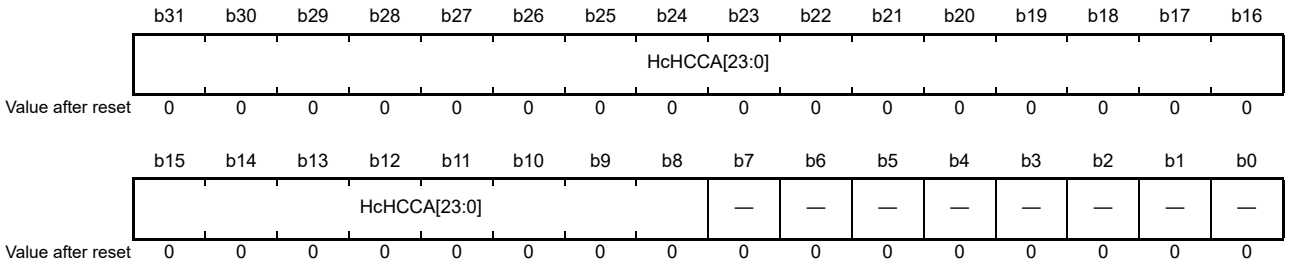
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	RHSCD	FNOD	UED	RDD	SFD	WDHD	SOD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SOD (Scheduling Overrun Disable)	SO interrupt source disable	This bit disables the interrupt source SO. Write 1 to this bit to clear the corresponding bit in the HcInterruptEnable register. To enable the interrupt, write 1 to the corresponding bit in the HcInterruptEnable register. 0: Nothing is to be done. 1: The interrupt source SO is disabled.	R/W
b1	WDHD (Writeback DoneHead Disable)	WDH interrupt source disable	This bit disables the interrupt source WDH. Write 1 to this bit to clear the corresponding bit in the HcInterruptEnable register. To enable the interrupt, write 1 to the corresponding bit in the HcInterruptEnable register. 0: Nothing is to be done. 1: The interrupt source WDH is disabled.	R/W
b2	SFD (StartOfFrame Disable)	SF interrupt source disable	This bit disables the interrupt source SF. Write 1 to this bit to clear the corresponding bit in the HcInterruptEnable register. To enable the interrupt, write 1 to the corresponding bit in the HcInterruptEnable register. 0: Nothing is to be done. 1: The interrupt source SF is disabled.	R/W
b3	RDD (Resume Detected Disable)	RD interrupt source disable	This bit disables the interrupt source RD. Write 1 to this bit to clear the corresponding bit in the HcInterruptEnable register. To enable the interrupt, write 1 to the corresponding bit in the HcInterruptEnable register. 0: Nothing is to be done. 1: The interrupt source RD is disabled.	R/W
b4	UED (Unrecoverable ErrorDisable)	UE interrupt source disable	This bit disables the interrupt source UE. Write 1 to this bit to clear the corresponding bit in the HcInterruptEnable register. To enable the interrupt, write 1 to the corresponding bit in the HcInterruptEnable register. 0: Nothing is to be done. 1: The interrupt source UE is disabled.	R/W
b5	FNOD (FrameNumber Overflow Disable)	FNO interrupt source disable	This bit disables the interrupt source FNO. Write 1 to this bit to clear the corresponding bit in the HcInterruptEnable register. To enable the interrupt, write 1 to the corresponding bit in the HcInterruptEnable register. 0: Nothing is to be done. 1: The interrupt source FNO is disabled.	R/W

Bit	Symbol	Bit Name	Description	R/W
b6	RHSCD (RootHub StatusChange Disable)	RHSC interrupt source disable	This bit disables the interrupt source RHSC. Write 1 to this bit to clear the corresponding bit in the HcInterruptEnable register. To enable the interrupt, write 1 to the corresponding bit in the HcInterruptEnable register. 0: Nothing is to be done. 1: The interrupt source RHSC is disabled.	R/W
b30 to b7	—	Reserved	When writing, write 0.	R/W
b31	MID (Master Interrupt Disable)	Interrupt 7 source disable	This bit disables interrupt source settings in bits [6:0] in HcInterruptEnable. When this bit is 0, all interrupts are masked. Write 1 to this bit to clear the corresponding bit in the HcInterruptEnable register. To enable the interrupt, write 1 to the corresponding bit in the HcInterruptEnable register. 0: Nothing is to be done. 1: The interrupt set is disabled.	R/W

23.3.1.7 HcHCCA Register

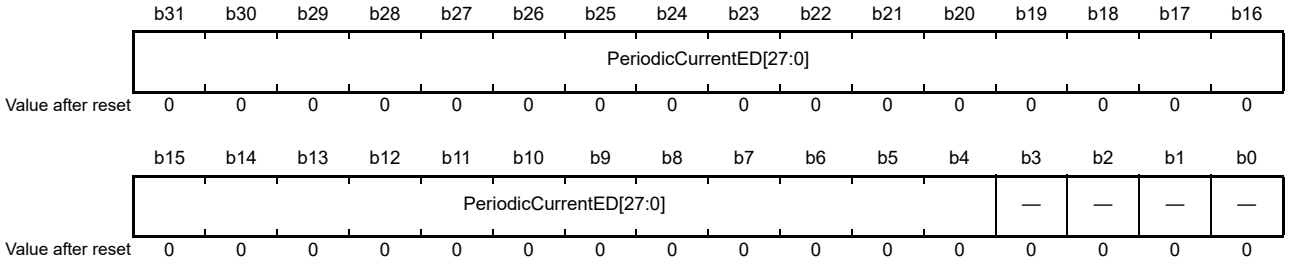
Address(es) A004 0018h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	When writing, write 0.	R/W
b31 to b8	HcHCCA [23:0]	RAM base address setting	These bits specify the base address of the RAM allocated to the host controller communication area. Specify a value in these bits during initialization. The host logic requests a 256-byte area starting from the base address specified in these bits as HCCA.	R/W

23.3.1.8 HcPeriodicCurrentED Register

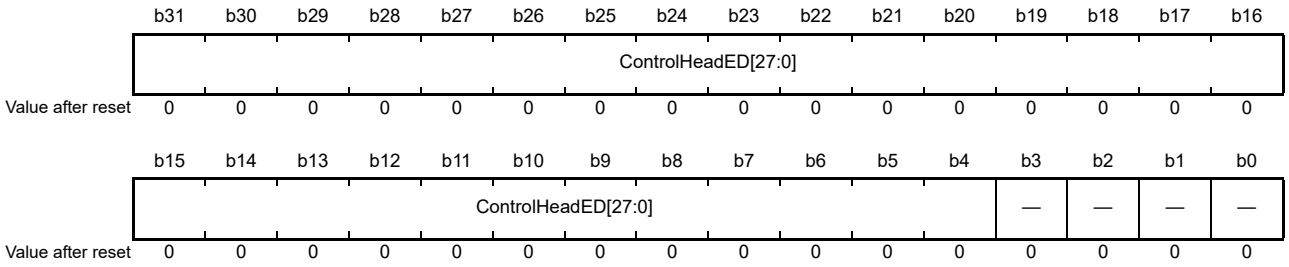
Address(es) A004 001Ch



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	Don't care	R
b31 to b4	Periodic CurrentED [27:0]	ED physical address	These bits indicate the physical address of the current ED being processed in the periodic list. The host logic updates these bits when completing the ED list processing.	R

23.3.1.9 HcControlHeadED Register

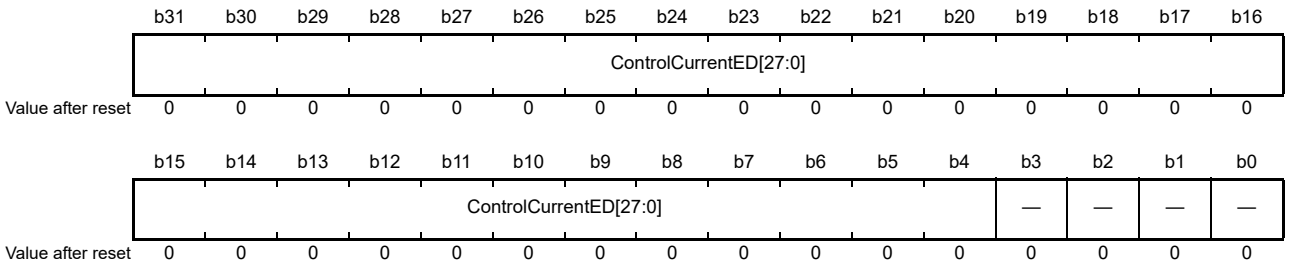
Address(es) A004 0020h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	When writing, always write 0.	R/W
b31 to b4	Control HeadED [27:0]	Start ED physical address specifying	These bits specify the physical address of the start ED in the control list. To execute control transfer, set up these bits before setting the CLE bit in the HcControl register.	R/W

23.3.1.10 HcControlCurrentED Register

Address(es) A004 0024h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	When writing, always write 0.	R/W
b31 to b4	Control CurrentED [27:0]	ED physical address	These bits indicate the physical address of the current ED being processed in the control list. The host logic updates these bits every time the control ED processing is completed. When creating a new list, set these bits to 0000 0000h, which indicates the end of the list. When transfer is temporarily stopped and then restarted, make sure that the ED indicated by the link pointer in ControlCurrentED exists.	R/W

23.3.1.11 HcBulkHeadED Register

Address(es) A004 0028h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	BulkHeadED[27:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BulkHeadED[27:0]												—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	When writing, always write 0.	R/W
b31 to b4	BulkHeadED [27:0]	Start ED physical address specifying	These bits specify the physical address of the start ED in the bulk list. To execute bulk transfer, set up these bits before setting bit 5 (BLE) in the HcControl register.	R/W

23.3.1.12 HcBulkCurrentED Register

Address(es) A004 002Ch

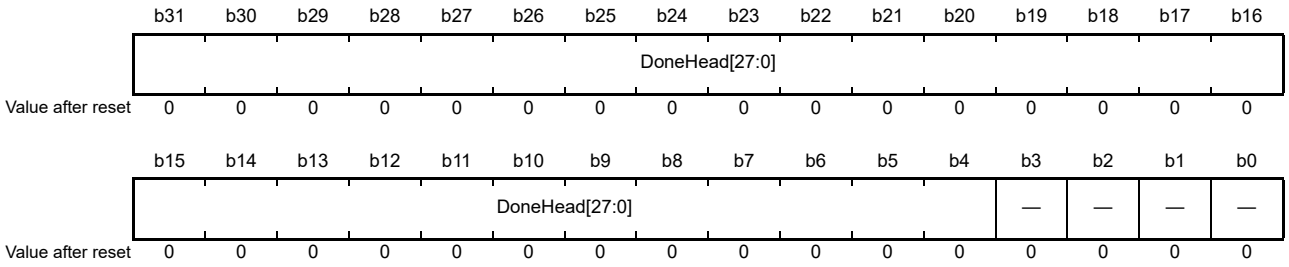
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	BulkCurrentED[27:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BulkCurrentED[27:0]												—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	When writing, always write 0.	R/W
b31 to b4	BulkCurrentED [27:0]	ED physical address	These bits indicate the physical address of the current ED being processed in the bulk list. The host logic updates these bits every time the bulk ED processing is completed. When creating a new list, set these bits to 0000 0000h, which indicates the end of the list. When transfer is temporarily stopped and then restarted, make sure that the ED indicated by the link pointer in BulkCurrentED exists.	R/W

23.3.1.13 HcDoneHead Register

Address(es) A004 0030h



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	Don't care	R
b31 to b4	DoneHead [27:0]	HcDoneHead physical address	These bits indicate the physical address of HcDoneHead in the host logic. This is the physical address of the last completed TD to be added to the Done queue.	R

23.3.1.14 HcFmInterval Register

Address(es) A004 0034h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	FIT	FSMPS[14:0]														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	FI[13:0]													
Value after reset	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b13 to b0	FI[13:0] (FrameInterval)	Frame interval setting	These bits specify the length (bit time) of a frame used for FS transfer. To conform to the USB standard (one frame = 1 ms), set these bits to 2EDFh.	R/W
b15, b14	—	Reserved	When writing, always write 0.	R/W
b30 to b16	FSMPS[14:0] (FSLagest DataPacket)	FSt transfer packet maximum size setting	These bits specify the maximum amount of data that can be transferred without generating a schedule overrun. This setting is compared with the current location in the frame to determine up to which data in the frame can be transferred. The maximum amount depends on the system bus performance, and it should be specified by the driver.	R/W
b31	FIT (FrameInterval Toggle)	Frame synchronization	This bit is used to match the frame settings between software and the host logic. When updating the FI bits, software should toggle this bit. When loading the FI bit value, the host logic copies the FIT value to bit 31 (FRT) in the HcFmRemaining register. Software can check if the new FI bit setting has been reflected by comparing the value specified in this bit when the FI bits are written to, and the value read from the FRT bit.	R/W

23.3.1.15 HcFmRemaining Register

Address(es) A004 0038h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	FRT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	FR[13:0]													
Value after reset	0	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b13 to b0	FR[13:0] (Frame Remaining)	Down counter frame	These are the register bits of a 14-bit down counter which counts down the current remaining time value during transmission of a frame. That is, the value of these bits is decremented over time. When the value reaches 0000h, these bits are reloaded with the frame interval value by copying this value from bits [13:0] (FI) in the HcFmInterval register, after which counting down is restarted.	R
b30 to b14	—	Reserved	Don't care	R
b31	FRT (Frame Remaining Toggle)	Frame synchronization	This bit is used to match the frame settings between software and the host logic. When the FR bits reach 0000h, the host logic copies the value of bits [13:0] (FI) in the HcFmInterval register to the FR bits and also copies the value of bit 31 (FIT) in the HcFmInterval register to this bit. Software can check if the FI bit setting has been copied to the FR bits by comparing the values of the FIT bit and this bit.	R

23.3.1.16 HcFmNumber Register

Address(es) A004 003Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FrameNumber[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	FrameNumber [15:0]	Elapsed frame number	These bits indicate the elapsed frame count. The count in these bits is incremented when the value of bits [13:0] (FR) in the HcFmRemaining register reaches 0000h.	R
b31 to b16	—	Reserved	Don't care	R

23.3.1.17 HcPeriodicStart Register

Address(es) A004 0040h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PeriodicStart[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b13 to b0	PeriodicStart [13:0]	Periodic list processing start time	These bits indicate the time when the host logic starts periodic list processing in a frame. Specify a value in these bits by software during initialization of the host logic. When the value of bits [13:0] (FR) in the HcFmRemaining register is greater than the value specified in these bits, non-periodic lists take priority over periodic lists. The OHCI standard recommends that this value be set to about 90% of the value of bits [13:0] (FI) in the HcFmInterval register. Therefore, the recommended value is 2A2Fh	R/W
b31 to b14	—	Reserved	When writing, write 0.	R/W

23.3.1.18 HcRhDescriptorA Register

Address(es) A004 0048h

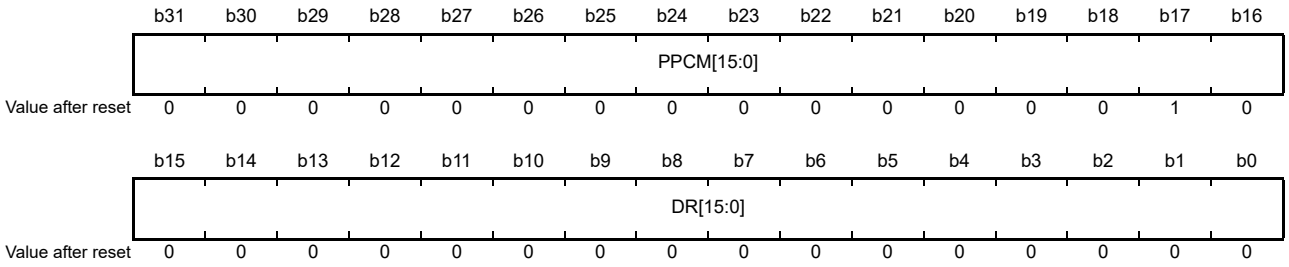
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	POTPGT[7:0]								—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	NOCP	OCPM	DT	NPS	PSM	NDP[7:0]							
Value after reset	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	NDP[7:0] (Number Downstream Port)	Downstream port number	These bits specify the number of downstream ports supported by the root hub in the host logic.	R
b8	PSM (PowerSwitchingMode)	Power switch control	This bit specifies how to control the power switch to each port of the root hub. While bits [31:16] (PPCM) in the HcRhDescriptorB register are set, each port responds only to Set/ClearPortPower. While the bits are cleared, each port is controlled by Set/ClearGlobalPower. This bit setting is valid only when the NPS bit is cleared. 0: Power to all ports is controlled together. 1: Power to each port is separately controlled.	R/W
b9	NPS (NoPower Switching)	Power control	This bit specifies how to control the port power. 0: Port power can be switched on and off. 1: Port power is always turned on while the host logic is operating.	R/W
b10	DT (DeviceType)	Device type	This bit indicates that the root hub is not a compound device. As the root hub is not allowed to be a compound device, this bit is always read as 0.	R
b11	OCPM (OverCurrent Protection Mode)	Overcurrent state reporting	This bit specifies how to report the overcurrent state of the root hub. This bit should be set to the same mode as that specified in the PSM bit. This setting is valid only when the NOCP bit is cleared. 0: Overcurrent state of all ports is collectively reported. 1: Overcurrent state of each port is separately reported.	R/W
b12	NOCP (NoOver Current Protection)	Overcurrent function support	This bit specifies whether to support the overcurrent protection function for the root hub. 0: Overcurrent protection function is supported. 1: Overcurrent protection function is not supported.	R/W
b23 to b13	—	Reserved	When writing, write 0.	R/W
b31 to b24	POTPGT[7:0] (PowerOnTo PowerGood Time)	Wait time	These bits indicate the time that software should wait before accessing a root hub port after power supply to the port is started. The unit of time is 2 ms. Therefore, the wait time is obtained by POTPGT × 2 ms.	R/W

23.3.1.19 HcRhDescriptorB Register

Address(es) A004 004Ch



Bit	Symbol	Bit Name	Description	R/W
b0	DR[15:0] (Device Removable)	Device removable	These bits indicate whether the device connected to each port in the root hub is removable. Each bit is dedicated to a single port. The USB host controller has port 1 only.	R
b1				R/W
b15 to b2				R

Bit	Value	Description
0	—	Reserved
1	0	Device connected to port 1 is not removable.
	1	Device connected to port 1 is removable.
[15:2]	—	Reserved

Bit	Symbol	Bit Name	Description	R/W
b16	PPCM[15:0] (PortPower ControlMask)	Power control command	These bits specify power control commands for each port. This setting is valid when bit 8 (PSM) in the HcRhDescriptorA register is set. Each bit is dedicated to a single port. The USB host controller has port 1 only.	R
b17				R/W
b31 to b18				R

Bit	Value	Description
0	—	Reserved
1	0	All ports are collectively controlled (Set/ClearGlobalPower).
	1	Port 1 is separately controlled (Set/ClearPortlPower).
[15:2]	—	Reserved

23.3.1.20 HcRhStatus_A, HcRhStatus_B Register

(1) HcRhStatus_A Register

Address(es) A004 0050h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCIC	SGP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCI	CGP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CGP (ClearGlobal Power)	Global power clear	Setting this bit to 1 turns off the power to a port. The port to be turned off is determined according to the settings of bit 8 (PSM) in the HcRhDescriptorA register and bits [31:16] (PPCM) in the HcRhDescriptorB register. The USB host controller has port 1 only.	W
b1	OCI (OverCurrent Indicator)	Overcurrent indicator	This bit reports the overcurrent state in global overcurrent detection mode. When overcurrent is reported separately for each port, this value is always 0. 0: Port is in normal state. 1: Port is in overcurrent state.	R
b14 to b2	—	Reserved	When writing, write 0.	R/W
b15	SRWE (SetRemote Wakeup Enable)	DRWE bit enable	This bit sets the DRWE bit. Setting this bit sets the DRWE bit. Writing 0 has no effect.	W
b16	SGP (SetGlobal Power)	Port power setting	Setting this bit to 1 turns on the power to a port. The port to be turned on is determined according to the settings of bit 8 (PSM) in the HcRhDescriptorA register and bits [31:16] (PPCM) in the HcRhDescriptorB register. The USB host controller has port 1 only.	W

value	PSM	PPCM[1]	Description
0	—	—	No Change
1	0	—	The PPS bit in the HcRhPortStatus1 register is cleared.
	1	0	The PPS bit in the HcRhPortStatus1 register is cleared.
		1	No Change

value	PSM	PPCM[1]	Description
0	—	—	No Change
1	0	—	The PPS bit in the HcRhPortStatus1 register is set.
	1	0	The PPS bit in the HcRhPortStatus1 register is set.
		1	No Change

Bit	Symbol	Bit Name	Description	R/W
b17	OCIC (OverCurrent Indicate Change)	OCI bit change report	This bit notifies that the OCI bit value has changed. It is set when the OCI bit changes. Writing 1 to this bit while it is set to 1 clears it. Writing 0 has no effect. 0: Nothing is to be done in case of an overcurrent. 1: OverCurrent state has changed.	R/W
b30 to b18	—	Reserved	When writing, write 0.	R/W
b31	CRWE (Clear Remote Wakeup Enable)	DRWE bit clear	This bit clears the DRWE bit. Setting this bit clears the DRWE bit. Writing 0 has no effect.	W

(2) HcRhStatus_B Register

Address(es) A004 0050h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCIC	LPSC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DRWE	—	—	—	—	—	—	—	—	—	—	—	—	—	OCI	LPS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	LPS (LocalPower Status)	Local power status	This product does not support the local power status function; this bit is always read as 0.	R
b1	OCI (OverCurrent Indicator)	Overcurrent indicator	This bit reports the overcurrent state in global overcurrent detection mode. When overcurrent is reported separately for each port, this value is always 0. 0: Port is in normal state. 1: Port is in overcurrent state.	R
b14 to b2	—	Reserved	When writing, write 0.	R/W
b15	DRWE (Device Remote Wakeup Enable)	Device remote start enable	This bit specifies whether to include bit 16 (CSC) in the HcRhPortStatus1 register as a remote wakeup event. When a connect status change event occurs while this bit is set, transition from the USB suspend state to the USB resume state occurs and the resume detect interrupt is generated. 0: ConnectStatusChange is not a RemoteWakeup source. 1: ConnectStatusChange is a RemoteWakeup source.	R
b16	LPSC (LocalPower Status Change)	Local power status change	This product does not support the local power status function and this bit is always read as 0.	R
b17	OCIC (OverCurrent Indicate Change)	OCI bit change report	This bit notifies that the OCI bit value has changed. It is set when the OCI bit changes. Writing 1 to this bit while it is set to 1 clears it. Writing 0 has no effect. 0: Nothing is to be done in case of an overcurrent. 1: OverCurrent state has changed.	R/W
b30 to b18	—	Reserved	When writing, write 0.	R/W
b31	CRWE (Clear Remote Wakeup Enable)	DRWE bit clear	This bit clears the DRWE bit. Setting this bit clears the DRWE bit. Writing 0 has no effect.	W

23.3.1.21 HcRhPortStatus1_A, HcRhPortStatus1_B Register

(1) HcRhPortStatus1_A Register

Address(es) A004 0054h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	PRSC	OCIC	PSSC	PESC	CSC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	CPP	SPP	—	—	—	SPR	CSS	SPS	SPE	CPE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CPE (ClearPort Enable)	PES bit clear	This bit is used to clear the PES bit. Writing 1 makes the port enter the disabled state. Writing 0 has no effect.	W
b1	SPE (SetPort Enable)	PES bit enable	This bit is used to set the PES bit. Writing 1 makes the port enter the enabled state. Writing 0 has no effect. Note 1. To control port state transition, use bit 4 (SPR) in the HcRhPortStatus1_A register. The OHCI standard supports use of the SPE bit to make the port enter the enabled state, but the USB standard does not support this.	W
b2	SPS (SetPort Suspend)	Port state Suspend transition	This bit shifts the port to the suspend state. Writing 1 makes the port enter the suspend state. Writing 0 has no effect. If this bit is written to while the CCS bit is cleared, the CSC bit is set to notify the driver of an attempt to suspend a disconnected port.	W
b3	CSS (ClearSuspend Status)	Suspend clear	This bit terminates the suspend state and starts the Resume sequence. Writing 1 starts the Resume sequence. Writing 0 has no effect. The Resume sequence starts only while the PSS bit is set.	W
b4	SPR (SetPortReset)	Port reset issuing	This bit applies a port reset to a downstream port. Writing 1 to this bit starts a 10-ms port reset. If this bit is written to while the CCS bit is cleared, the PRS bit will not be set. Instead, the CSC bit is set to notify software of an attempt to reset a port where no device is connected. Writing 0 has no effect.	W
b7 to b5	—	Reserved	When writing, write 0.	R/W
b8	SPP (SetPort Power)	Port power-on	This bit turns on the port power when power is separately controlled for each port. Writing 1 turns on the port. Writing 0 has no effect.	W
b9	CPP (ClearPort Power)	Port power clear	This bit turns off the port power when power is separately controlled for each port. Writing 1 turns off the port. Writing 0 has no effect.	W
b15 to b10	—	Reserved	When writing, write 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b16	CSC (ConnectStatus Change)	CCS bit status	This bit indicates that the CCS bit value has changed. The host logic sets this bit when the CCS bit changes due to device connection or disconnection. In addition, when a request for port reset, port suspension, or port enable is issued in the disconnection state, this bit is set to make software re-evaluate the device connection state. It is cleared when 1 is written by software. 0: Nothing is to be done regarding CCS. 1: CCS has changed.	R/W
b17	PESC (PortEnable StatusChange)	PES bit status	This bit indicates that the PES bit value has changed. It is set when the port state has changed due to a hardware event, such as the overcurrent state, disconnection, power-off, or a babble error. It is cleared when 1 is written by software. 0: Nothing is to be done regarding PES. 1: PES state has changed.	R/W
b18	PSSC (PortSuspend StatusChange)	RESUME sequence complete	This bit indicates that the RESUME sequence has been completed. It is set when all RESUME processing by hardware has been completed. It is cleared when 1 is written by software. In addition, it is cleared when the PRSC bit is set. 0: RESUME has not been completed. 1: RESUME has been completed.	R/W
b19	OCIC (OverCurrent IndicateChange)	Overcurrent state detection	This bit is set when the overcurrent state of the port has been detected. It is valid only while the overcurrent state is reported separately for each port (OCPM = 1 in the HcRhDescriptorA register). It is cleared when 1 is written by software. 0: Nothing is to be done regarding overcurrent state. 1: Overcurrent state has changed.	R/W
b20	PRSC (PortReset StatusChange)	Port reset complete	This bit indicates that a port reset has been completed. It is set when the host logic has completed a 10-ms hardware reset. It is cleared when 1 is written by software. 0: Port reset has not been completed, or nothing is to be done regarding the PRS bit. 1: Port reset has been completed.	R/W
b31 to b21	—	Reserved	When writing, write 0.	R/W

(2) HcRhPortStatus1_B Register

Address(es) A004 0054h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	PRSC	OCIC	PSSC	PESC	CSC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	LSDA	PPS	—	—	—	PRS	POCI	PSS	PES	CCS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CCS (Current ConnectStatus)	Connection status indication	This bit indicates the current connection state of the downstream port. 0: No device is connected. 1: Device is connected.	R
b1	PES (PortEnable Status)	Port state enable status	This bit indicates whether the port is enabled or disabled. The host logic automatically clears this bit upon detecting the overcurrent state, disconnection, power-off, or a babble error. At this time, the PESC bit is set. This bit cannot be set while the CCS bit is cleared (no device is connected). The host logic sets this bit when a port reset is completed or suspension of the port is terminated. 0: Port is disabled. 1: Port is enabled.	R
b2	PSS (PortSuspend Status)	Suspend/Resume status	This bit indicates that the port is in the suspend state or the Resume sequence. Writing to the SPS bit sets this bit. This bit is not set while the CCS bit is cleared (no device is connected). This bit is cleared with the following timing: • When the Resume sequence is completed and the PSSC bit is set. • When a port reset is completed and the PRSC bit is set. • In the USB RESUME state. 0: Port is in normal transfer state. 1: Port is in suspend state.	R
b3	POCI (PortOver Current Indicator)	Downstream port overcurrent detection	This bit indicates that a downstream port has entered the overcurrent state. It is valid only while the overcurrent state is reported separately for each port (OCPM = 1 in the HcRhDescriptorA register). While the overcurrent state in all ports is collectively reported, this bit is 0b. 0: Port is in normal state. 1: Port is in overcurrent state.	R
b4	PRS (PortReset Status)	Port reset status	This bit indicates the reset state of the port. It is cleared at the same time as when the PRSC bit is set after a 10-ms port reset is completed. This bit cannot be set while the CCS bit is cleared (device is not connected). 0: Port reset is not in progress. 1: Port reset is in progress.	R
b7 to b5	—	Reserved	When writing, write 0.	R/W
b8	PPS (PortPower Status)	Power status	This bit indicates the power state of the port. It is cleared when an overcurrent is detected. 0: Port power is off. 1: Port power is on.	R

Bit	Symbol	Bit Name	Description	R/W
b9	LSDA (LowSpeed DeviceAttached)	Device speed	This bit indicates the speed of the device connected to the port. This status bit is valid only while the CCS bit is set. 0: FS device is connected. 1: LS device is connected.	R
b15 to b10	—	Reserved	When writing, write 0.	R/W
b16	CSC (ConnectStatus Change)	CCS bit status	This bit indicates that the CCS bit value has changed. The host logic sets this bit when the CCS bit changes due to device connection or disconnection. In addition, when a request for port reset, port suspension, or port enable is issued in the disconnection state, this bit is set to make software re-evaluate the device connection state. It is cleared when 1 is written by software. 0: Nothing is to be done regarding CCS. 1: CCS has changed.	R/W
b17	PESC (PortEnable StatusChange)	PES bit status	This bit indicates that the PES bit value has changed. It is set when the port state has changed due to a hardware event, such as the overcurrent state, disconnection, power-off, or a babble error. It is cleared when 1 is written by software. 0: Nothing is to be done regarding PES. 1: PES state has changed.	R/W
b18	PSSC (PortSuspend StatusChange)	RESUME sequence complete	This bit indicates that the RESUME sequence has been completed. It is set when all RESUME processing by hardware has been completed. It is cleared when 1 is written by software. In addition, it is cleared when the PRSC bit is set. 0: RESUME has not been completed. 1: RESUME has been completed.	R/W
b19	OCIC (OverCurrent IndicateChange)	Overcurrent state detection	This bit is set when the overcurrent state of the port has been detected. It is valid only while the overcurrent state is reported separately for each port (OCPM = 1 in the HcRhDescriptorA register). It is cleared when 1 is written by software. 0: Nothing is to be done regarding overcurrent state. 1: Overcurrent state has changed.	R/W
b20	PRSC (PortReset StatusChange)	Port reset complete	This bit indicates that a port reset has been completed. It is set when the host logic has completed a 10-ms hardware reset. It is cleared when 1 is written by software. 0: Port reset has not been completed, or nothing is to be done regarding the PRS bit. 1: Port reset has been completed.	R/W
b31 to b21	—	Reserved	When writing, write 0.	R/W

23.3.2 EHCI Operational Registers

Access the EHCI operational registers after starting up the PHY internal PLL. For details, see Figure 23.13, Initial Setting Sequence.

23.3.2.1 HCIVERSION/CAPLENGTH Register

Address(es) A004 1000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Interface Version Number[15:0]															
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	Capability Registers Length[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Capability Registers Length [7:0]	Host logic operational register start address	These bits indicate the start address of the host logic operational registers. As the host logic operational registers are allocated from address 20h, this value is set to 20h.	R
b15 to b8	—	Reserved	Don't care	R
b31 to b16	Interface Version Number [15:0]	EHCI version	These bits indicate the version of the EHCI specifications supported by the host logic. As the host logic conforms to the EHCI Rev. 1.0, this value is set to 0100h.	R

23.3.2.2 HCSPARAMS Register

Address(es) A004 1004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	Debug Port Number[3:0]				—	—	—	P_INDICATOR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	N_CC[3:0]				N_PCC[3:0]				Port Routing Rules	—	—	PPC	N_PORTS[3:0]			
Value after reset	0	0	0	1	0	0	0	1	1	0	0	1	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	N_PORTS [3:0] (Number of Ports)	Number of downstream port	These bits indicate the number of physical downstream ports used by the host logic. These bits reflect the setting of bits [1:0] (Port_no) in the PCI Configuration EXT1 register. As the USB host controller has only one port, this value is set to 1h.	R
b4	PPC (Port Power Control)	Port power control	This bit indicates how to control the port power in the host logic. As the USB host controller supports the power supply control function, this value is set to 1.	R
b6, b5	—	Reserved	Don't care	R
b7	Port Routing Rules	Port routing rules	This bit indicates how ports are mapped in the OHCI host logic. As the value of the HCSP_PORTROUTE register shows the mapping in the host logic, this value is set to 1h.	R
b11 to b8	N_PCC[3:0] (Number of Ports per Companion Controller)	Number of Port	These bits indicate the number of the port supported by one OHCI host logic unit. These bits reflect the setting of bits [1:0] (Port_no) in the PCI Configuration EXT1 register. As the USB host controller has only one port, this value is set to 1h.	R
b15 to b12	N_CC[3:0] (Number of Companion Controller)	Number of OHCI host logic	These bits indicate the number of OHCI host logic units related to the EHCI host logic. As the host logic has one OHCI host logic unit, this value is set to 1h.	R
b16	P_INDICATOR	Port indicator control support	This bit indicates whether the host logic supports the port indicator control function. As the host logic does not support the port indicator control function, this value is set to 0.	R
b19 to b17	—	Reserved	Don't care	R
b23 to b20	Debug Port Number[3:0]	Debug port number	These bits indicate that the host logic ports are debug ports. As the host logic has no debug ports, this value is set to 0000b.	R
b31 to b24	—	Reserved	Don't care	R

23.3.2.3 HCCPARAMS Register

Address(es) A004 1008h

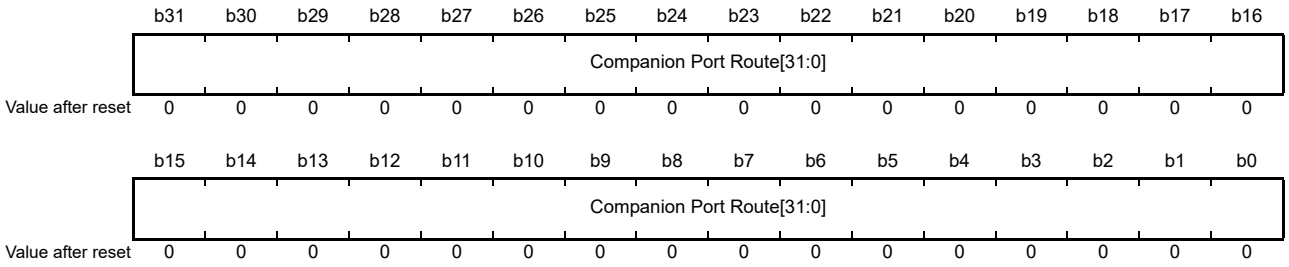
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	EECP[7:0]								IST[3:0]				—	ASPC	PFLF	AC64
Value after reset	1	1	1	0	1	0	0	0	0	0	0	0	0	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	AC64	Memory pointer selection	This bit indicates whether data structures use 32-bit address memory pointers or 64-bit address memory pointers. As the host logic supports data structures using 32-bit address memory pointers, this value is set to 0. 64-bit address memory pointers are not supported.	R
b1	PFLF	Programming frame list flag	This bit indicates the setting regarding the frame list size that software can use. This value is set to 1 in this host logic. When this bit is set to 1, the available frame list size can be specified through bits [3:2] (Frame List Size) in the USBCMD register. A frame list size smaller than 4 Kbytes can be specified.	R
b2	ASPC	Asynchronous schedule park support capability	This bit indicates whether to support the park mode for the high-speed QH (queue head) in the asynchronous schedule. As the host logic supports this function, this value is set to 1.	R
b3	—	Reserved	Don't care	R
b7 to b4	IST[3:0]	Isochronous data structure threshold	As the host logic in this product does not support caching of the isochronous data structure in the entire frame, these bits are set to 0h.	R
b15 to b8	EECP[7:0] (EHCI Extend Capabilities Pointer)	Offset address	These bits indicate the offset address for the EHCI extend capabilities registers. This indicates that extend registers are placed at addresses starting from E8h in the EHCI configuration space. As the host logic does not support the legacy function, the value read from these bits has no meaning.	R
b31 to b16	—	Reserved	Don't care	R

23.3.2.4 HCSP_PORTROUTE Register

Address(es) A004 100Ch



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	Companion Port Route[31:0]	OHCI Host Port Indication	These bits indicate the ports that the OHCI host logic controls. As the host logic has one OHCI host logic unit, these bits are set to 0000 0000h.	R

23.3.2.5 USBCMD Register

Address(es) A004 1020h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	Interrupt Threshold Control[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	Asynchronous Schedule Park Mode Enable	—	ASPMC[1:0]		Light Host Controller Reset	Interrupt on Async Advance Doorbell	ASPME	Periodic Schedule Enable	Frame List Size[1:0]		HCRESET	RS
Value after reset	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	RS (Run/Stop)	EHCI host logic run/stop	This bit activates or stops the EHCI host logic. When this bit is set to 1, the host logic starts operation. As long as this value is 1, the host logic continues operation. Note that this bit should be set to 1 only while the host logic is in the halt state. Bit 12 (HCHalted) in the USBSTS register indicates that the host logic has completed transactions and entered the halt state. 0: Stopped (the host logic has completed transactions and halted). 1: Activated (the host logic executes the schedule).	R/W
b1	HCRESET (Host Controller Reset)	Host logic initialization	This bit initializes the host logic. When this bit is set to 1, the host logic initializes the internal pipelines and state machine. Communication through the USB stops immediately. A USB reset is not issued to downstream ports at this time. This reset does not initialize the PCI configuration registers, but it initializes the EHCI operational registers and the port ownership is returned to OHCI. This bit is automatically cleared to 0 by the host logic when the reset is completed. The reset cannot be aborted by software by writing 0 to this bit. This bit should be set only while bit 12 (HCHalted) in the USBSTS register is 1.	R/W
b3, b2	Frame List Size[1:0]	Frame list size	These bits specify the frame list size. The setting in these bits determines the size of the Frame List Current index in the FRINDEX register. b3 b2 0 0: 1024 elements (4096 bytes) 0 1: 512 elements (2048 bytes) 1 0: 256 elements (1024 bytes) 1 1: Reserved	R/W
b4	Periodic Schedule Enable	Periodic schedule enable	This bit specifies whether to advance or skip periodic list processing. 0: Periodic list processing is not executed (skipped). 1: Periodic list processing is executed using the PERIODICLISTBASE register. Note: For transferring data via USB while this bit is being 0, set b12 of the EXT1 register to 0. For details, see section 23.3.3.12, Offset E0h Register (EXT1).	R/W
b5	ASPME	Asynchronous schedule enable	This bit specifies whether to advance or skip asynchronous list processing. 0: Asynchronous list processing is not executed (skipped). 1: Asynchronous list processing is executed using the ASYNCLISTADDR register.	R/W

Bit	Symbol	Bit Name	Description	R/W
b6	Interrupt on Async Advance Doorbell	Doorbell	This bit is used as a doorbell by software. Software sets this bit to 1 to generate an interrupt when processing proceeds with the next QH (queue head). While bit 5 (Interrupt on Async Advance Enable) in the USBINTR register is set to 1, an interrupt is generated with the next interrupt timing after 1 is written to this bit. If this bit is set while bit 5 (Interrupt on Async Advance Enable) in the USBINTR register is 0, correct operation is not guaranteed. This bit is cleared by the host logic. Upon completing the processing of a QH normally, the host logic clears this bit to 0 and sets bit 5 (Interrupt on Async Advance) in the USBSTS register to 1.	R/W
b7	Light Host Controller Reset	Light host controller reset execution status	This bit indicates the state of Light Host Controller Reset execution. As the host logic does not support Light Host Controller Reset, this value is fixed to 0.	R
b9, b8	ASPMC[1:0]	Asynchronous schedule park transaction count	These bits specify the number of transactions that can be executed in succession from one QH (queue head). A value from 1h to 3h can be specified. This setting is valid when bit 11 (Asynchronous Schedule Park Mode Enable) is 1.	R/W
b10	—	Reserved	When writing, write 0.	R/W
b11	Asynchronous Schedule Park Mode Enable	Park mode enable	This bit enables or disables the park mode. 0: Disabled 1: Enabled	R/W
b15 to b12	—	Reserved	When writing, write 0.	R/W
b23 to b16	Interrupt Threshold Control[7:0]	Host logic interrupt generation maximum rate	These bits indicate the maximum rate at which the host logic generates an interrupt. Correct operation is not guaranteed if a value not shown below is written to these bits. 00h: Reserved 01h: 1 micro-frame 02h: 2 micro-frames 04h: 4 micro-frames 08h: 8 micro-frames (1 ms) 10h: 16 micro-frames (2 ms) 20h: 32 micro-frames (4 ms) 40h: 64 micro-frames (8 ms)	R/W
b31 to b24	—	Reserved	When writing, write 0.	R/W

23.3.2.6 USBSTS Register

Address(es) A004 1024h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Asynchronous Schedule Status	Periodic Schedule Status	Reclamation	HCHalted	—	—	—	—	—	—	Interrupt on Async Advance	Host System Error	Frame List Rollover	Port Change Detect	USBERRINT	USBINT
Value after reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	USBINT (USB Interrupt)	USB transfer complete	This bit indicates that USB transfer has been completed. The host logic sets this bit to 1 when either of the following conditions is satisfied. • USB transfer is completed • A short packet is received Even if USB transfer has ended with an error, this bit is set to 1 when the IOC (Interrupt On Complete) bit in the TD is 1. Writing 1 can clear this bit. Writing 0 has no effect. 0: USB transfer has not been completed. 1: USB transfer has been completed.	R/W
b1	USBERRINT (USB Error Interrupt)	USB transaction status	This bit indicates that a USB transaction has ended with an error. The host logic sets this bit to 1 when a USB transaction has ended with an error. Such an error includes the case when the error counter underflows. Writing 1 can clear this bit. Writing 0 has no effect. 0: USB transaction has been completed normally. 1: USB transaction has ended with an error.	R/W
b2	Port Change Detect	Port state change detection	This bit indicates that the port state has changed. The host logic sets this bit to 1 when any of the following conditions is satisfied in a port for which bit 13 (Port Owner) in the PORTSC[n] register is set to 0. [1 setting condition] • Bit 1 (Connect Status Change) in the PORTSC[n] register has changed from 0 to 1. (Device connection or disconnection has been detected.) • Bit 3 (Port Enable/Disable Change) in the PORTSC[n] register has changed from 0 to 1. (The port enabled state has changed.) • Bit 5 (Over-current Change) in the PORTSC[n] register has changed from 0 to 1. (The overcurrent state has been detected.) • Bit 6 (Force Port Resume) in the PORTSC[n] register has changed from 0 to 1. (indicating detection of a J-to-K transition on a port while suspended.) Writing 1 by software can clear this bit. Writing 0 has no effect.	R/W
b3	Frame List Rollover	Frame list rollover	The host logic sets this bit to 1 when the value of the Frame Index bits in the FRINDEX register returns from the maximum value to 000h (rollover). The maximum value at which rollover occurs depends on bits [3:2] (Frame List Size) in the USBCMD register. Writing 1 can clear this bit. Writing 0 has no effect. 0: Frame list has not returned to 000h. 1: Frame list has returned to 000h.	R/W

Bit	Symbol	Bit Name	Description	R/W
b4	Host System Error	Host system error	This bit is set to 1 when a serious error occurs in the host logic. Such an error includes a parity error in the PCI system. If this error occurs, the host logic clears bit 0 (RS) in the USBCMD register to 0 to stop execution of the remaining TDs. Writing 1 can clear this bit. Writing 0 has no effect. 0: No system error has occurred. 1: System error has occurred.	R/W
b5	Interrupt on Async Advance	Async advance interrupt status	This bit indicates the Async Advance interrupt state. Upon fetching QH, the host logic checks bit 6 (Interrupt on Async Advance Doorbell (IAAD)) in the USBCMD register. When the IAAD bit is 1, the host logic clears the IAAD bit and sets this bit after completing QH processing normally. When bit 5 (Interrupt on Async Advance Enable) in the USBINTR register is 1, an interrupt is generated from this source with the next interrupt timing after this bit is set to 1. Writing 1 can clear this bit. Writing 0 has no effect. 0: No Async Advance interrupt has occurred. 1: Async Advance interrupt state has been detected.	R/W
b11 to b6	—	Reserved	When writing, write 0.	R/W
b12	HCHalted	EHCI host logic status	This bit is set to 0 when bit 0 (RS) in the USBCMD register is 1. When the RS bit is cleared to 0 by the host logic or software, the EHCI host logic stops execution and this bit is set to 1 by the host logic. 0: EHCI host logic is in execution. 1: EHCI host logic is stopped.	R
b13	Reclamation	Empty asynchronous schedule detection	This bit is used to detect an empty asynchronous schedule. The host logic clears this bit to 0 after a reset or when it fetches QH with H = 1. When the host logic executes an asynchronous transaction or has detected a start event, it sets this bit to 1. When the host logic fetches QH with H = 1b while this bit is 0, the Async Sched Sleeping mode is entered.	R
b14	Periodic Schedule Status	Periodic schedule status	This bit indicates the current state of the periodic schedule. The periodic schedule is enabled (1) or disabled (0) when the values of this bit and bit 4 (Periodic Schedule Enable) in the USBCMD register are the same. 0: Periodic schedule is disabled. 1: Periodic schedule is enabled.	R
b15	Asynchronous Schedule Status	Asynchronous schedule status	This bit indicates the current state of the asynchronous schedule. The asynchronous schedule is enabled (1) or disabled (0) when the values of this bit and bit 5 (ASPME) in the USBCMD register are the same. 0: Asynchronous schedule is disabled. 1: Asynchronous schedule is enabled.	R
b31 to b16	—	Reserved	When writing, write 0.	R/W

23.3.2.7 USBINTR Register

Address(es) A004 1028h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	Interrupt on Async Advance Enable	Host System Error Enable	Frame List Rollover Enable	Port Change Interrupt Enable	USB Error Interrupt Enable	USB Interrupt Enable
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	USB Interrupt Enable	USB interrupt enable	This bit enables or disables the setting in bit 0 (USBINT) in the USBSTS register. To clear the interrupt, use the USBINT bit. 0: Disabled 1: Enabled	R/W
b1	USB Error Interrupt Enable	USB error interrupt enable	This bit enables or disables the setting in bit 1 (USBERRINT) in the USBSTS register. To clear the interrupt, use the USBERRINT bit. 0: Disabled 1: Enabled	R/W
b2	Port Change Interrupt Enable	Port change interrupt enable	This bit enables or disables the setting in bit 2 (Port Change Detect) in the USBSTS register. To clear the interrupt, use the Port Change Detect bit. 0: Disabled 1: Enabled	R/W
b3	Frame List Rollover Enable	Frame list rollove enable	This bit enables or disables the setting in bit 3 (Frame List Rollover) in the USBSTS register. To clear the interrupt, use the Frame List Rollover bit. 0: Disabled 1: Enabled	R/W
b4	Host System Error Enable	Host system error enable	This bit enables or disables the setting in bit 4 (Host System Error) in the USBSTS register. To clear the interrupt, use the Host System Error bit. 0: Disabled 1: Enabled	R/W
b5	Interrupt on Async Advance Enable	Interrupt on async advance enable	This bit enables or disables the setting in bit 5 (Interrupt on Async Advance) in the USBSTS register. To clear the interrupt, use the Interrupt on Async Advance bit. 0: Disabled 1: Enabled	R/W
b31 to b6	—	Reserved	When writing, write 0.	R/W

23.3.2.8 FRINDEX Register

Address(es) A004 102Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	Frame Index[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b13 to b0	Frame Index[13:0]	Frame index	These bits are used by the host logic to add an index to the periodic frame list. This value is incremented at the end of each micro-frame. Bits [N:3] are used as the Frame List Current index. This means that the current frame list is accessed 8 times before proceeding with the next index number. The value of N is determined as follows according to the setting of bits [3:2] (Frame List Size) in the USBCMD register. **Frame List Size / Number Elements / N** 00 (1024) 12 01 (512) 11 10 (256) 10 11 Reserved This register should be accessed only while the host logic is stopped (bit 12 (HCHalted) = 1 in the USBSTS register). This setting is reflected in the SOF frame number for the SOF token.	R/W
b31 to b14	—	Reserved	When writing, write 0.	R/W

23.3.2.9 CTRLDSSEGMENT Register

Address(es) A004 1030h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CTRLDSSEGMENT[31:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CTRLDSSEGMENT[31:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CTRLDSSEGMENT[31:0]	—	The host logic does not support 64-bit addressing and does not use this register. Therefore, do not access this register.	R

23.3.2.10 PERIODICLISTBASE Register

Address(es) A004 1034h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	BaseAddress (Low) [21:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BaseAddress (Low) [21:0]				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b11 to b0	—	Reserved	When writing, write 0.	R/W
b31 to b12	BaseAddress (Low) [21:0]	Periodic frame list start address	These bits indicate the start address of the periodic frame list stored in the system memory. Software should specify the start address in this register before the host logic starts list processing. The host logic uses the values in these bits and bits [13:0] (Frame Index) in the FRINDEX register to determine the frame list to be processed. The address of the periodic frame list should be aligned with a 4-Kbyte boundary. Correct operation cannot be guaranteed if the value in these bits is modified during operation.	R/W

23.3.2.11 ASYNCLISTADDR Register

Address(es) A004 1038h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	LPL[26:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LPL[26:0]											—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	—	Reserved	When writing, write 0.	R/W
b31 to b5	LPL[26:0] (Link Pointer Low)	Asynchronous Queue Head link pointer address	These bits indicate the address of the next asynchronous queue head in the system memory to be processed. The address of the asynchronous queue head should be aligned with a 32-byte boundary.	R/W

23.3.2.12 CONFIGFLAG Register

Address(es) A004 1060h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	CF (Configure Flag)	Port routing control circuit configuration flag	This bit controls the default setting of the port routing control circuit regarding whether the ports are routed to OHCI or EHCI. Software should set this bit to 1 at the end of host logic configuration process. 0: Port routing control circuit routes each port to the OHCI host logic by default. 1: Port routing control circuit routes each port to the EHCI host logic by default.	R/W
b31 to b1	—	Reserved	When writing, write 0.	R/W

23.3.2.13 PORTSC1 Register

Address(es) A004 1064h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	WKOC_E	WKDSCNNT_E	WKCNNT_E	Port Test Control [3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Port Indicator Control[1:0]		Port Owner	PP	Line Status[1:0]		—	Port Reset	Suspend	Force Port Resume	Over-current Change	Over-current Active	Port Enable/ Disable Change	Port Enabled/ Disabled	Connect Status Change	Current Connect Status
Value after reset	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	Current Connect Status	Port connection status	This bit indicates the connection state of the port. Upon detecting device connection, the host logic sets this bit to 1. The host logic also sets this bit to 1 when Port Test Control[3:0] = 0101b (Test FORCE_ENABLE) even if no device is connected. Upon detecting device disconnection, the host logic clears this bit to 0. When the PP bit is 0 or the Port Owner bit is 0, this bit becomes 0. 0: No device is connected to the port. 1: Device is connected to the port.	R
b1	Connect Status Change	Connect status change	This bit indicates that the value of bit 0 (Current Connect Status) has changed. Writing 1 can clear this bit. Writing 0 has no effect. When the PP bit is 0, this bit becomes 0. 0: Nothing is to be done. 1: Value of bit 0 (Current Connect Status) has changed.	R/W
b2	Port Enabled/ Disabled	Port enable/disable status	This bit indicates the enabled/disabled state of the port. After resetting a port and detecting that the connected device is an HS device, the host logic enables the port and sets this bit to 1. Software cannot set this bit to 1. Upon detecting device disconnection or an error, the host logic disables the port and clears this bit to 0. Software can write 0 to disable the port. In this case, however, the bit value changes only after the port state actually changes. While the port is disabled, data transfer to downstream ports is blocked except for a port reset. When the PP bit is 0, this bit becomes 0. Note that when Port Test Control[3:0] = 0101b (Test FORCE_ENABLE), the port is enabled regardless of the port state and this bit is set to 1. 0: Port is disabled. 1: Port is enabled.	R/W
b3	Port Enable/ Disable Change	Port enable/disable status change	This bit indicates that the enabled/disabled state of the port has changed. Upon detecting a frame babble error, the host logic disables the port and sets this bit to 1. Writing 1 can clear this bit. Writing 0 has no effect. When the PP bit is 0, this bit becomes 0. 0: Nothing is to be done. 1: The port state has changed from enabled to disabled.	R/W
b4	Over-current Active	Port overcurrent status	This bit indicates the overcurrent state of the port. Upon detecting an overcurrent, the host logic clears bit 12 (PP) and its related bits and sets this bit to 1b. This bit is automatically cleared from 1 to 0 when the overcurrent state is resolved. 0: Port is not in overcurrent state. 1: Port is in overcurrent state.	R

Bit	Symbol	Bit Name	Description	R/W
b5	Over-current Change	Over-current Change	This bit indicates that the value of bit 4 (Over-current Active) has changed. Writing 1 can clear this bit. Writing 0 has no effect. 0: Nothing is to be done. 1: Value of bit 4 (Over-current Active) has changed.	R/W
b6	Force Port Resume	Port resume detection flag	This bit indicates detection of the resume state at the port. When a transition from the J to the K state (RemoteWakeup) is detected while the port is suspended, the host logic sets this bit and bit 2 (Port Change Detect) in the USBSTS register to 1. Software should also set this bit to 1 when the output of a resume signal is required. In this case, the Port Change Detect bit should not be set. The resume signal (FS K State) is driven on the USB port while this bit is 1. Clear this bit to 0 after an appropriate period has passed. Writing 0 to this bit while it is 1 returns the port to the HS Idle state. Until then, this bit remains at 1. When the PP bit is 0, this bit becomes 0. 0: Resume (K-state) has not been detected or output. 1: Resume (K-state) has been detected or output.	R/W
b7	Suspend	Port suspend	This bit indicates the suspend state of the port. 0: Port is not in suspend state. 1: Port is in suspend state. The combination of the settings of this bit and bit 2 (Port Enabled/Disabled) indicate the port state as follows. **Port Enabled/Disabled / Suspend / Port State** 0 / X / Disabled 1 / 0 / Enabled 1 / 1 / Suspend In the suspend state, data transfer from this port to the downstream port is blocked unless the port is reset. When this bit is set to 1 during data transfer, reflection in the status bit and blocking of transfer follow completion of the current transfer. Software should write 1 to this bit. It can write 1 only while PP bit = 1, Port Owner bit = 0, and Current Connect Status bit = 1 in the register in the host logic. This bit is always cleared to 0 under either of the following conditions. • Software clears the Force Port Resume bit to 0. • Software sets the Port Reset bit to 1. When the PP bit is 0, this bit becomes 0.	R/W
b8	Port Reset	Port reset status	This bit indicates the reset state of the port. 0: Port is not in reset state. 1: Port is in reset state. When software writes 1 to this bit while it is 0, the bus reset sequence defined in the USB2.0 standard begins. Write 0 to this bit to terminate the bus reset sequence. Note that this bit must be retained at 1 for long enough time for the bus reset sequence to be completed according to the USB2.0 standard. When bit 12 (HCHalted) in the USBSTS register is 1, the port should not be reset. This bit becomes 0 when the PP, Port Owner, or Current Connect Status bit satisfies the following conditions. • PP bit = 0, • Port Owner bit = 1, or • Current Connect Status bit = 0	R/W
b9	—	Reserved	When writing, write 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b11, b10	Line Status[1:0]	D+/D- logic level	These bits indicate the current logical levels on the D+ and D- lines of the USB port (bit 11: D+; bit 10: D-). They are used to detect an LS device before a port reset or a sequence for enabling the port. Therefore, the values in these bits are valid only while bit 3 (Port Enable/Disable) = 0 and bit 0 (Current Connect Status) = 1. When the PP bit is 0, these bits become 0.	R
b12	PP (Port Power)	Port power supply control	This bit controls power supply to the port. When this bit is 0, power is not supplied to the port; the port does not operate and connection or disconnection cannot be detected. If an overcurrent is detected while this bit is 1, the host logic clears this bit to 0 and stops power supply to the port. 0: No power is supplied to the port. 1: Power is supplied to the port. The function of this bit depends on the setting of bit 4 (PPC) in the HCSPARAMS register.	R/W
b13	Port Owner	Port ownership	This bit indicates whether OHCI or EHCI has the port ownership. 0: EHCI has the port ownership. 1: OHCI has the port ownership. This bit is cleared to 0 when bit 0 (CF) in the CONFIGFLAG register changes from 0 to 1. This bit is set to 1 when the CF bit is 0. Software should set this bit to 1 to pass the port ownership to OHCI when the connected device is not a high-speed device.	R/W
b15, b14	Port Indicator Control[1:0]	—	As the host logic does not support the port indicator control function, these bits are set to 00b. Writing to these bits does not affect the operation.	R
b19 to b16	Port Test Control[3:0]	Pin test control	These bits control the test mode. For details of the test mode, see Chapter 7 in the USB Specification Revision 2.0.	R/W

bit11 (D+)	bit10 (D-)	USB State	Interpretation
0	0	SE0	Not an LS device. Moves to the EHCI port reset execution process.
0	1	K State	An LS device has been connected. Passes the port ownership from EHCI to OHCI.
1	0	J State	Not an LS device. Moves to the EHCI port reset execution process.
1	1	Un-defined	Not an LS device. Moves to the EHCI port reset execution process.

PPC	PP	Description
0	1	This bit is fixed to 1, and power is always supplied to the port.
1	0/1	Power supply to the port is determined by this bit.

Port Test Control[3:0]	Mode
0000b	Normal
0001b	Test J_STATE
0010b	Test K_STATE
0011b	Test SE0_NAK
0100b	Test Packet
0101b	Test FORCE_ENABLE
Other than the above	Reserved

Bit	Symbol	Bit Name	Description	R/W
b20	WKCNNT_E (Wake on Connect Enable)	Device connection detection enable	Writing 1 to this bit enables detection of device connection as a wakeup event. This bit setting does not affect the host logic operation. When the PP bit is 0, this bit becomes 0.	R/W
b21	WKDSCNNT_E (Wake on Disconnect Enable)	Device disconnection detection enable	Writing 1 to this bit enables detection of device disconnection as a wakeup event. This bit setting does not affect the host logic operation. When the PP bit is 0, this bit becomes 0.	R/W
b22	WKOC_E (Wake on Over-current Enable)	Overcurrent state detection enable	Writing 1 to this bit enables detection of the overcurrent state as a wakeup event. This bit setting does not affect the host logic operation. When the PP bit is 0, this bit becomes 0.	R/W
b31 to b23	—	Reserved	When writing, write 0.	R/W

23.3.3 PCI Configuration Registers for OHCI

23.3.3.1 Offset 00h Register (Vendor ID, Device ID)

Address(es) A005 0000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Device ID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Vendor ID[15:0]															
Value after reset	0	0	0	1	0	0	0	0	0	0	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	Vendor ID [15:0]	Device vendor ID	This register indicates the vendor of the device. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R
b31 to b16	Device ID [15:0]	Device ID	This register indicates the device type. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R

23.3.3.2 Offset 04h Register (Command, Status)

Address(es) A005 0004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Detected Parity Error	Signaled System Error	Received Master Abort	Received Target Abort	Signaled Target Abort	Devsel Timing[1:0]		Data Parity Error Detected	Fast Back to Back Capable	—	—	Capabilities List	—	—	—	—
Value after reset	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0

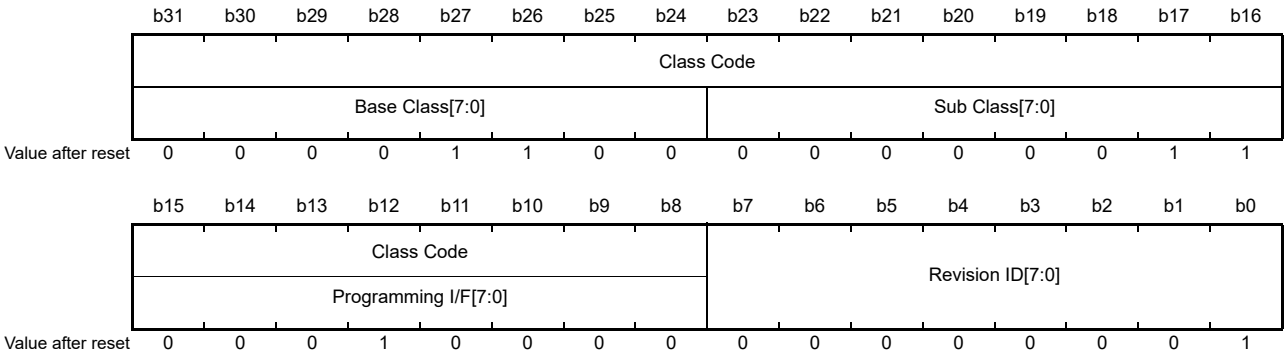
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	Fast Back to Back Enable	SERR Enable	Wait Cycle Control	Parity Error Response	VGA Palette Snoop	Memory Write and Invalidate Enable	Special Cycle	Bus Master	Memory Space	I/O Space
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	I/O Space	I/O space access enable	This bit enables access to the I/O space. As the host logic does not accept I/O access, this value is always 0.	R
b1	Memory Space	Memory space access enable	This bit enables access to the memory space. This is an enable signal for memory access specified in the PCI standard. Set to 1 when accessing registers. Set this bit to 1 during initial setting of the host logic.	R/W
b2	Bus Master	Bus master enable	This bit enables bus master operation. This is an enable signal for master access to the PCI bus. Set to 1 when accessing the SRAM on the system bus. Set this bit to 1 during initial setting of the host logic.	R/W
b3	Special Cycle	Special cycle enable	This bit enables Special Cycle operation. As the host logic does not support the Special Cycle operation, this value is always 0.	R
b4	Memory Write and Invalidate Enable	Memory write and invalidate enable	This bit enables the Memory Write and Invalidate command. In the USB host controller, do not change this from the value after reset (0). 0: Memory write and invalidate command is disabled. 1: Memory write and invalidate command is enabled.	R/W
b5	VGA Palette Snoop	VGA palette snoop enable	This bit enables VGA palette snooping. As the host logic does not support VGA palette snooping, this value is always 0.	R
b6	Parity Error Response	Parity error response enable	This bit enables parity error response. 0: PERR0 is not asserted. 1: PERR0 is asserted. When a parity error is detected, the Detected Parity Error bit is set to 1 even if this bit is cleared to 0.	R/W
b7	Wait Cycle Control	Wait cycle control enable	This bit enables wait cycle control. As the host logic does not support address and data stepping, this value is always 0.	R
b8	SERR Enable	System error response enable	This bit enables system error response. 0: SERR0 is not asserted. 1: SERR0 is asserted. To notify of a system error through the SERR signal, set this bit to 1.	R/W
b9	Fast Back to Back Enable	Fast back to back enable	This bit enables fast back to back transactions. As the host logic does not support fast back to back transactions, this value is always 0.	R
b19 to b10	—	Reserved	When writing, write 0.	R/W
b20	Capabilities List	Power management mode support	This value is fixed to 1, which indicates that the power management mode is supported.	R
b22, b21	—	Reserved	When writing, write 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b23	Fast Back to BackCapable	Fast back to back support	This bit indicates whether Fast Back to Back transactions are supported. As the host logic does not support Fast Back to Back transactions, this value is fixed to 0.	R
b24	Data Parity Error Detected	Data parity error detection flag	This bit is set when a parity error is detected in master operation. Writing 1 from the PCI bus clears this bit. The value is fixed to 0 while the Parity Error Response bit is set as disabled.	R/W
b26, b25	Devsel Timing[1:0]	DEVSEL response speed	These bits indicate the DEVSEL response speed. The value is fixed to 01b (medium-speed response).	R
b27	Signaled Target Abort	Slave/Target abort status	This is a status bit for Slave/Target abort. This bit is set to 1 in slave operation when the host logic has terminated, through Target Abort, the bus cycle in which the host logic is accessed. Writing 1 from the PCI bus clears this bit.	R/W
b28	Received Target Abort	Master/Target abort status	This is a status bit for Master/Target Abort. This bit is set to 1 in master operation when the bus cycle being executed by the host logic is terminated through Target Abort. Writing 1 from the PCI bus clears this bit.	R/W
b29	Received Master Abort	Master/Master abort status	This is a status bit for Master/Master Abort. This bit is set to 1 in master operation when the bus cycle being executed by the host logic is terminated through Master Abort. Writing 1 from the PCI bus clears this bit.	R/W
b30	Signaled System Error	SERR status	This is a status bit for SERR. This bit is set to 1 when a system error occurs. Writing 1 from the PCI bus clears this bit.	R/W
b31	Detected Parity Error	Parity error status	This is a status bit for parity error. This bit is set when an address or data parity error is detected. Writing 1 from the PCI bus clears this bit.	R/W

23.3.3.3 Offset 08h Register (Revision ID, Class Code)

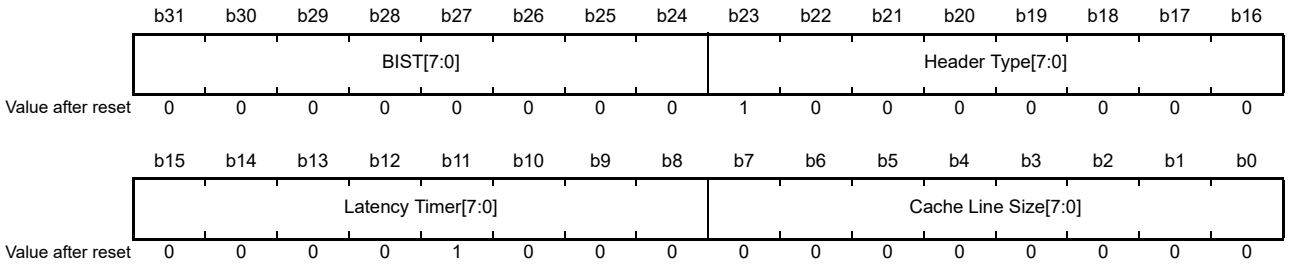
Address(es) A005 0008h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Revision ID [7:0]	Host logic revision	These bits indicate the revision of the host logic. The value is indicated as 01h.	R
b15 to b8	Programming I/F[7:0]	PCI standard program interface	These bits indicate the program interface specified in the PCI standard. The value is 10h, which indicates OHCI.	R
b23 to b16	Sub Class [7:0]	PCI standard subclass	These bits indicate the subclass specified in the PCI standard. The value is 03h, which indicates a USB device.	R
b31 to b24	Base Class [7:0]	PCI standard base class	These bits indicate the base class specified in the PCI standard. The value is 0Ch, which indicates a serial peripheral bus controller.	R

23.3.3.4 Offset 0Ch Register (Cache Line Size, Latency Timer, Header Type, BIST)

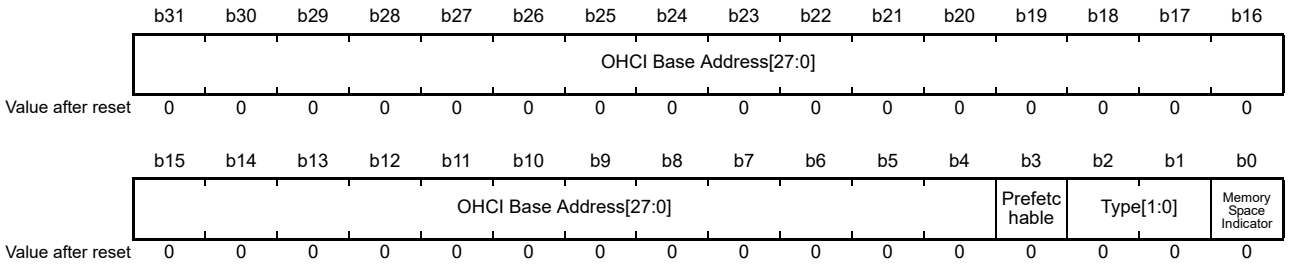
Address(es) A005 000Ch



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Cache Line Size[7:0]	Cache Line Size	These bits notify the system of the cache line size.	R/W
b9, b8	Latency Timer[7:0]	Latency Timer	These bits notify the system of the latency timer. The lowest 2 bits are fixed to 00b.	R
b15 to b10				R/W
b23 to b16	Header Type[7:0]	Header Type	These bits notify the system of the header type. As the header type is Type 0, bits [22:16] are fixed to 00h. As this is a multifunction device, bit 23 is fixed to 1.	R
b31 to b24	BIST[7:0]	Self-test	These bits are used for self testing. As the host logic does not support the self test function, this value is always 00h.	R

23.3.3.5 Offset 10h Register (OHCI Base Address)

Address(es) A005 0010h



Bit	Symbol	Bit Name	Description	R/W
b0	Memory Space Indicator	Memory space indicator	This bit is fixed to 0, which indicates that the OHCI operational registers are mapped to the system memory space.	R
b2, b1	Type[1:0]	Base address type	These bits are fixed to 00b, which indicates that the base address of the OHCI operational registers has a 32-bit width and can be allocated to any location in a 32-bit memory space.	R
b3	Prefetchable	Prefetch setting	As the host logic does not support prefetching in memory read cycles, this bit is fixed to 0.	R
b11 to b4	OHCI Base Address [27:0]	OHCI base address	Bits [31:12] specify the base address of the operational registers. Specify the base address of the operational registers determined by the system during initialization. Bits [11:4] are fixed to 00h, which indicates that the operational registers are allocated to a 4-Kbyte address space.	R
b31 to b12				R/W

23.3.3.6 Offset 2Ch Register (Subsystem Vendor ID, Subsystem ID)

Address(es) A005 002Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Subsystem ID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Subsystem Vendor ID[15:0]															
Value after reset	0	0	0	1	0	0	0	0	0	0	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	Subsystem Vendor ID [15:0]	Subsystem vendor ID	This register indicates the vendor of the device. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R
b31 to b16	Subsystem ID[15:0]	Subsystem ID	This register indicates the device type. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R

23.3.3.7 Offset 30h Register (Expansion ROM Base Address)

Address(es) A005 0030h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Expansion ROM Base Address[21:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Expansion ROM Base Address[21:0]						—	—	—	—	—	—	—	—	—	ROM Decode Enable
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ROM Decode Enable	Expansion ROM decode enable	As decoding of expansion ROM is prohibited, this bit is always read as 0. This bit cannot be written to.	R
b9 to b1	—	Reserved	Don't care	R
b31 to b10	Expansion ROM Base Address[21:0]	Expansion ROM base address	As decoding of expansion ROM is prohibited, these bits are always read as 000000h. These bits cannot be written to.	R

23.3.3.8 Offset 34h Register (Capability Pointer)

Address(es) A005 0034h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	Capability Pointer[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Capability Pointer[7:0]	Capability identifier pointer	These bits hold the pointer to the capability identifier. As the pointer is set to 40h in the host logic, this value is set to 40h.	R
b31 to b8	—	Reserved	Don't care	R

23.3.3.9 Offset 3Ch Register (Interrupt Line, Interrupt Pin, Min gnt, Max Latency)

Address(es) A005 003Ch

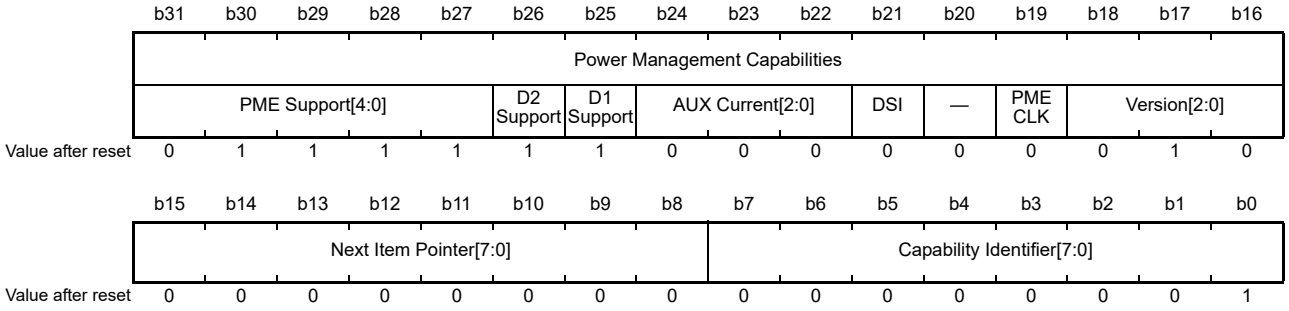
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Max Latency[7:0]								Min Gnt[7:0]							
Value after reset	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Interrupt Pin[7:0]								Interrupt Line[7:0]							
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Interrupt Line [7:0]	Interrupt line	These bits indicate the interrupt line. In the USB host controller, do not change this from the value after reset (00h).	R/W
b15 to b8	Interrupt Pin [7:0]	Interrupt output pin	These bits indicate the pin for outputting interrupts. As INTA is used, this value is fixed to 01h.	R
b23 to b16	Min Gnt[7:0]	Minimum burst transfer time	These bits indicate the minimum burst transfer time. As the minimum time is set to 01h in the host logic, this value is set to 01	R
b31 to b24	Max Latency[7:0]	Maximum frequency of PCI bus acquisition	These bits indicate the maximum frequency of PCI bus acquisition. As the maximum frequency is set to 2Ah in the host logic, this value is set to 2Ah.	R

23.3.3.10 Offset 40h Register (Capability Identifier, Next Item Pointer, Power Management Capabilities)

Address(es) A005 0040h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Capability Identifier [7:0]	PCI power management register ID	These bits indicate the PCI power management register ID. The value is fixed to 01h.	R
b15 to b8	Next Item Pointer[7:0]	Next item non-existence	These bits are fixed to 00h, which indicates that the next item does not exist.	R
b18 to b16	Version[2:0]	Version	These bits are fixed to 010b, which indicates that the host logic conforms to the PCI Power Management Interface Specification Release 1.1.	R
b19	PME CLK	USBPCLK unnecessary	This bit is fixed to 0, which indicates that USBPCLK is not required for PME interrupt generation.	R
b20	—	Reserved	Don't care	R
b21	DSI	Power management initialization unnecessary	This bit is fixed to 0, which indicates that special initialization is not required for power management.	R
b24 to b22	Aux Current[2:0]	Current setting value	These bits indicate the necessary current for 3.3-V auxiliary power supply. As PME interrupt generation from the D3 cold state is not supported, this value is fixed to 000b.	R
b25	D1 Support	PCI power state D1 support	This bit is fixed to 1, which indicates that PCI power state D1 is supported.	R
b26	D2 Support	PCI power state D2 support	This bit is fixed to 1, which indicates that PCI power state D2 is supported.	R
b30 to b27	PME Support[4:0]	PME interrupt support	These bits are fixed to 1111b, which indicates that PME interrupt generation is supported in all PCI power states (D0 to D3).	R
b31		D3 cold state support	This bit indicates whether the D3 cold state is supported. As the D3 cold state is not supported, this value is fixed to 0.	R

23.3.3.11 Offset 44h Register (Power Management Control/Status, PMCSR Bridge Support Extensions)

Address(es) A005 0044h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Data								PMCSR Bridge Support Extensions							
	Data[7:0]								BPCC Enable	B2_B3	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Power Management Control/Status															
	PME Status	Data Scale[1:0]		Data Select[3:0]				PME Enable	—	—	—	—	—	—	Power State[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	Power State[1:0]	PCI power status	These bits indicate the power state of the PCI as follows. b1 b0 0 0: D0 State 0 1: D1 State 1 0: D2 State 1 1: D3 hot State	R/W
b7 to b2	—	Reserved	When writing, write 0.	R/W
b8	PME Enable	PME enable	This bit specifies whether to use PME. Setting this bit to 1 generates a PME interrupt when operation returns from the power management state.	R/W
b12 to b9	Data Select[3:0]	Data selection field	The value of these bits is 0h. This is an optional field in the PCI standard, and the host logic does not support it.	R
b14, b13	Data Scale[1:0]	Data scale field	The value of these bits is 00b. This is an optional field in the PCI standard, and the host logic does not support it.	R
b15	PME Status	PME interrupt status	This bit indicates the PME interrupt state. It is set to 1 when the following condition for PME generation is satisfied. [PME generation condition] Bit 3 (RD) in the HcInterruptStatus register is set to 1 while bit 10 (RWE) in the HcControl register is 1. Writing 1 from the PCI bus clears this bit to 0.	R/W
b21 to b16	—	Reserved	When writing, write 0.	R/W
b22	B2_B3	Bit for bridge	The value of this bit is 0. This is a bit for the bridge, and the host logic does not support it.	R
b23	BPCC Enable	BPCC enable	The value of this bit is 0. This is a bit for the bridge, and the host logic does not support it.	R
b31 to b24	Data[7:0]	Data field	The value of these bits is 00h. This is an optional field in the PCI standard, and the host logic does not support it.	R

23.3.3.12 Offset E0h Register (EXT1)

This register is the same as the EXT1 register located in the EHCI configuration space.
Therefore, this register can be accessed also by using the offset address on the EHCI configuration register side.

Address(es) A005 00E0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	potpgt								Hyper Speed transfer Control #2					—	—	—
Value after reset	0	0	0	0	1	1	1	1	0	0	0	1	0	1	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	Hyper Speed transfer Control #1	PSD	—	—	—	—	ID_Write _Enable	—	—	—	—	—	Port_no	
Value after reset	0	0	1	1	0	0	1	1	0	0	0	1	1	1	0	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	Port_no	USB downstream port number	These bits specify the number of valid USB downstream ports. **Setting Value** / **Valid Ports** 01b / Port1 10b / Port1 and Port2 Other than the above / Reserved In the USB host controller, do not change this from the value after reset (1h).	R/W
b6 to b2	—	Reserved	Do not change this from the value after reset.	R/W
b7	ID_Write _Enable	Write-protect control	This bit write-protects the Subsystem ID, Subsystem Vendor ID, Max Latency, and Min Gnt bits. 0: Write-protected 1: Can be written to.	R/W
b11 to b8	—	Reserved	Do not change this from the value after reset.	R/W
b12	PSD	Periodic schedule disable	0: For transferring data via USB while the USBCMD.Periodic Schedule Enable bit is being 0, always set this bit to 0. 1: No settings required (initial value) Setting this bit is arbitrary when the USBCMD.Periodic Schedule Enable bit is 1.	R/W
b13	Hyper Speed transfer Control #1 (HS Async OUT advance Mode)	Hyper-speed transfer mode setting	This bit specifies the hyper-speed transfer mode for asynchronous OUT transfer. Setting to 1 enables this function (transfer rate improvement).	R/W
b18 to b14	—	Reserved	Do not change this from the value after reset.	R/W
b23 to b19	Hyper Speed transfer Control #2	HS asynchronous FIFO threshold setting field	Do not specify any value other than 02h (HS asynchronous FIFO threshold = 64 bytes).	R/W
b31 to b24	Potpgt	POTPGT setting field	These bits specify the value for bits [31:24] (PPOTPGT) in the OHCI HcRhDescriptorA register. POTPGT is the time that software should wait before accessing a root hub port after power supply to the port is started.	R/W

23.3.3.13 Offset E4h Register (EXT2)

This register is the same as the EXT2 register located in the EHCI configuration space.
Therefore, this register can be accessed also by using the offset address on the EHCI configuration register side.
However, note that bit 0 (EHCI_mask) can be written to only on the OHCI side.

Address(es) A005 00E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	RAM Connect Check Result	RAM Connect Check END Flag	RUN RAM Connect Check
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Hyper Speed transfer Control #3	EHCI_mask
Value after reset	0	1	1	0	1	1	0	0	0	0	0	0	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	EHCI_mask	EHCI host logic mask	This bit enables or disables the EHCI host logic. 0: EHCI host logic is enabled. 1: EHCI host logic is disabled. When this bit is set to 1, register access in the EHCI PCI configuration space and memory space is disabled and the EHCI host logic does not operate.	R/W
b1	Hyper Speed transfer Control #3	Hyper-speed transfer mode setting	This bit specifies the hyper-speed transfer mode for asynchronous IN/OUT transfer. Setting to 1 enables this function (transfer rate improvement).	R/W
b15 to b2	—	Reserved	Do not change this from the value after reset.	R/W
b16	RUN RAM Connect Check	RAM connection check circuit activation	This bit activates the RAM connection check circuit. Set this bit to 1 to start the RAM connection check. This bit is not cleared even after the check is completed. To check again, write 0 to this bit to clear it, and then write 1 again to start the check. When this bit changes from 0 to 1, the connection check circuit is reset and the RAM Connect Check END Flag bit and RAM Connect Check Result bit are cleared.	R/W
b17	RAM Connect Check END Flag	RAM connection check end flag	This bit indicates the end of RAM connection check. 0: Connection check has not been done or ended. 1: Connection check has ended. This bit is set when a specified time (about 2 µs) has passed after the RAM connection check is started by modifying the RUN RAM check bit from 0 to 1.	R
b18	RAM Connect Check Result	RAM connection check result	This bit indicates the result of RAM connection check. 0: Connection check result is NG. 1: Connection check result is OK. This value is valid when the RAM Connect Check END Flag bit is 1. Once the connection check is done, this value is not cleared until the RUN RAM Connect Check bit changes from 0 to 1.	R
b31 to b19	—	Reserved	Do not change this from the value after reset.	R/W

23.3.4 PCI Configuration Registers for EHCI

23.3.4.1 Offset 00h Register (Vendor ID, Device ID)

Address(es) A005 0100h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Device ID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Vendor ID[15:0]															
Value after reset	0	0	0	1	0	0	0	0	0	0	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	Vendor ID [15:0]	Vendor ID	This register indicates the vendor of the device. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R
b31 to b16	Device ID [15:0]	Device ID	This register indicates the device type. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R

23.3.4.2 Offset 04h Register (Command, Status)

Address(es) A005 0104h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Detected Parity Error	Signaled System Error	Received Master Abort	Received Target Abort	Signaled Target Abort	Devsel Timing[1:0]		Data Parity Error Detected	Fast Back to Back Capable	—	Capable 66MHz	Capabilities List	—	—	—	—
Value after reset	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0

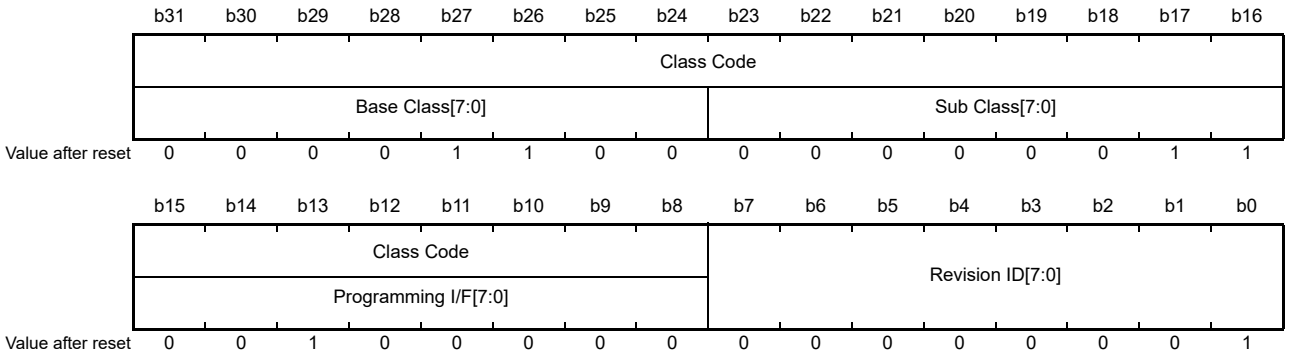
	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	Fast Back to Back Enable	SERR Enable	Wait Cycle Control	Parity Error Response	VGA Palette Snoop	Memory Write and Invalidate Enable	Special Cycle	Bus Master	Memory Space	I/O Space
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	I/O Space	I/O space access enable	This bit enables access to the I/O space. As the host logic does not accept I/O access, this value is always 0.	R
b1	Memory Space	Memory space access enable	This bit enables access to the memory space. This is an enable signal for memory access specified in the PCI standard. Set to 1 when accessing registers. Set this bit to 1 during initial setting of the host logic.	R/W
b2	Bus Master	Bus master enable	This bit enables bus master operation. This is an enable signal for master access to the PCI bus. Set to 1 when accessing the SRAM on the system bus. Set this bit to 1 during initial setting of the host logic.	R/W
b3	Special Cycle	Special cycle enable	This bit enables special cycle operation. As the host logic does not support the special cycle operation, this value is always 0.	R
b4	Memory Write and Invalidate Enable	Memory write and invalidate enable	This bit enables the Memory Write and Invalidate command. In the USB host controller, do not change this from the value after reset (0). 0: Memory write and invalidate command is disabled. 1: Memory write and invalidate command is enabled.	R/W
b5	VGA Palette Snoop	VGA palette snoop enable	This bit enables VGA palette snooping. As the host logic does not support VGA palette snooping, this value is always 0.	R
b6	Parity Error Response	Parity error response enable	This bit enables parity error response. 0: PERR0 is not asserted. 1: PERR0 is asserted. When a parity error is detected, the Detected Parity Error bit is set to 1 even if this bit is cleared to 0.	R/W
b7	Wait Cycle Control	Wait cycle control enable	This bit enables wait cycle control. As the host logic does not support address and data stepping, this value is always 0.	R
b8	SERR Enable	System error response enable	This bit enables system error response. 0: SERR0 is not asserted. 1: SERR0 is asserted. To notify of a system error through the SERR signal, set this bit to 1.	R/W
b9	Fast Back to Back Enable	Fast back to back enable	This bit enables fast back to back transactions. As the host logic does not support fast back to back transactions, this value is always 0.	R
b19 to b10	—	Reserved	When writing, write 0.	R/W
b20	Capabilities List	Power management mode support	This value is fixed to 1, which indicates that the power management mode is supported.	R

Bit	Symbol	Bit Name	Description	R/W
b21	Capable 66MHz	66-MHz operation capable	This bit indicates whether 66-MHz operation is available. As the host logic operates only at 33 MHz, this value is fixed to 0.	R
b22	—	Reserved	When writing, write 0.	R/W
b23	Fast Back to Back Capable	Fast back to back capable	This bit indicates whether fast back to back transactions are supported. As the host logic does not support fast back to back transactions, this value is fixed to 0.	R
b24	Data Parity Error Detected	Parity error detection	This bit is set when a parity error is detected in master operation. Writing 1 from the PCI bus clears this bit. The value is fixed to 0 when the Parity Error Response bit is set as disabled.	R/W
b26, b25	Devsel Timing[1:0]	DEVSEL response speed bit field	These bits indicate the DEVSEL response speed. The value is fixed to 01b (medium-speed response).	R
b27	Signaled Target Abort	Slave/Target abort status	This is a status bit for Slave/Target abort. This bit is set to 1 in slave operation when the host logic has terminated the bus cycle in which the host logic is accessed through Target Abort. Writing 1 from the PCI bus clears this bit.	R/W
b28	Received Target Abort	Master/Target abort status	This is a status bit for Master/Target abort. This bit is set to 1 in master operation when the bus cycle being executed by the host logic is terminated through Target Abort. Writing 1 from the PCI bus clears this bit.	R/W
b29	Received Master Abort	Master/Master abort status	This is a status bit for Master/Master abort. This bit is set to 1 in master operation when the bus cycle being executed by the host logic is terminated through Master Abort. Writing 1 from the PCI bus clears this bit.	R/W
b30	Signaled System Error	SERR status	This is a status bit for SERR. This bit is set when a system error occurs. Writing 1 from the PCI bus clears this bit.	R/W
b31	Detected Parity Error	Parity error status	This is a status bit for parity error. This bit is set when an address or data parity error is detected. Writing 1 from the PCI bus clears this bit.	R/W

23.3.4.3 Offset 08h Register (Revision ID, Class Code)

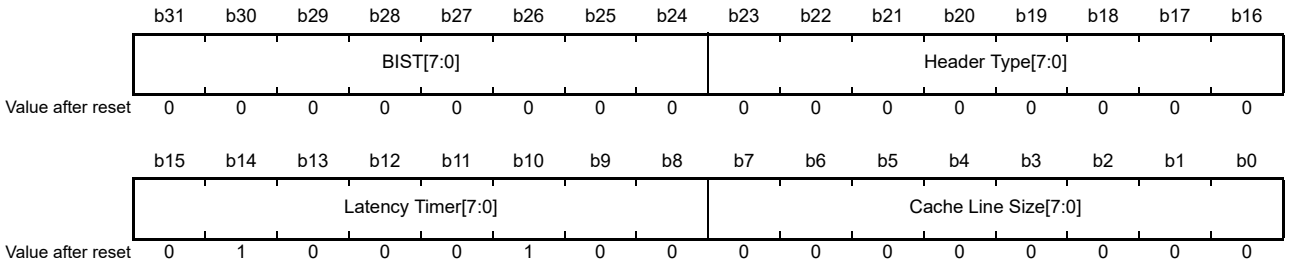
Address(es) A005 0108h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Revision ID [7:0]	Host logic revision	These bits indicate the revision of the host logic. The value is fixed to 01h.	R
b15 to b8	Programming I/F[7:0]	PCI standard program interface bit field	These bits indicate the program interface specified in the PCI standard. The value is 20h, which indicates EHCI.	R
b23 to b16	Sub Class [7:0]	PCI standard subclass	These bits indicate the subclass specified in the PCI standard. The value is 03h, which indicates a USB device.	R
b31 to b24	Base Class [7:0]	PCI standard base class	These bits indicate the base class specified in the PCI standard. The value is 0Ch, which indicates a serial peripheral bus controller.	R

23.3.4.4 Offset 0Ch Register (Cache Line Size, Latency Timer, Header Type, BIST)

Address(es) A005 010Ch



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Cache Line Size[7:0]	Cache line size	These bits notify the system of the cache line size.	R/W
b9, b8	Latency Timer[7:0]	Latency timer	These bits notify the system of the latency timer. The lowest 2 bits are fixed to 00b.	R
b15 to b10				R/W
b23 to b16	Header Type [7:0]	Header type	These bits notify the system of the header type. As the header type is Type 0, bits [22:16] are fixed to 00h. As the multifunction capability is not supported, bit 23 is fixed to 0.	R
b31 to b24	BIST[7:0]	Self testing field	These bits are used for self testing. As the host logic does not support the self test function, this value is always 00h.	R

23.3.4.5 Offset 10h Register (EHCI Base Address)

Address(es) A005 0110h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	EHCI Base Address[27:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	EHCI Base Address[27:0]												Prefetchable	Type[1:0]		Memory Space Indicator
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	Memory Space Indicator	System memory space indicator	This bit is fixed to 0, which indicates that the EHCI operational registers are mapped to the system memory space.	R
b2, b1	Type[1:0]	Base address allocation type	These bits are fixed to 00b, which indicates that the base address of the EHCI operational registers has a 32-bit width and can be allocated to any location in a 32-bit memory space.	R
b3	Prefetchable	Prefetch support	As the host logic does not support prefetching in memory read cycles, this bit is fixed to 0.	R
b7 to b4	EHCI Base Address [27:0]	EHCI base address	Bits [31:8] specify the base address of the operational registers. Specify the base address of the operational registers determined by the system during initialization. Bits [7:4] are fixed to 0h, which indicates that the operational registers are allocated to a 256-byte address space.	R
b31 to b8				R/W

23.3.4.6 Offset 2Ch Register (Subsystem Vendor ID, Subsystem ID)

Address(es) A005 012Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Subsystem ID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Subsystem Vendor ID[15:0]															
Value after reset	0	0	0	1	0	0	0	0	0	0	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	Subsystem Vendor ID [15:0]	Subsystem vendor ID	This register indicates the vendor of the device. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R
b31 to b16	Subsystem ID [15:0]	Subsystem ID	This register indicates the device type. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R

23.3.4.7 Offset 30h Register (Expansion ROM Base Address)

Address(es) A005 0130h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Expansion ROM Base Address[21:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Expansion ROM Base Address[21:0]						—	—	—	—	—	—	—	—	—	ROM Decode Enable
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ROM Decode Enable	ROM decode enable	As decoding of expansion ROM is prohibited, this bit is always read as 0. This bit cannot be written to.	R
b9 to b1	—	Reserved	Don't care	R
b31 to b10	Expansion ROM Base Address[21:0]	Expansion ROM base address	As decoding of expansion ROM is prohibited, these bits are always read as 000000h. These bits cannot be written to.	R

23.3.4.8 Offset 34h Register (Capability Pointer)

Address(es) A005 0134h

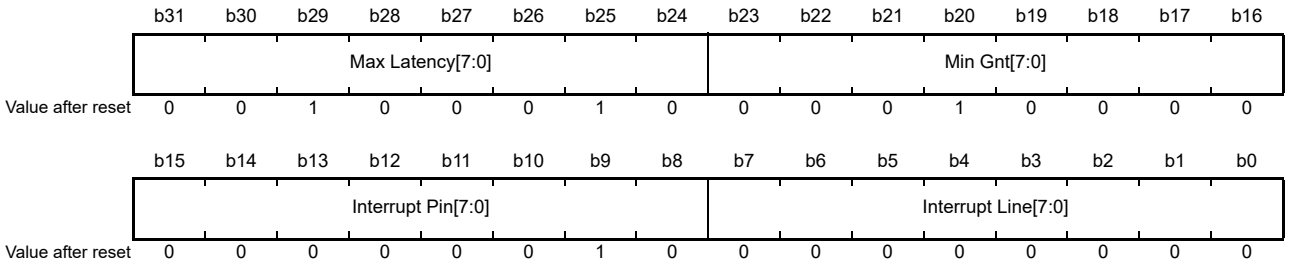
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	Capability Pointer[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Capability Pointer[7:0]	Capability identifier pointer	These bits hold the pointer to the capability identifier. As the pointer is set to 40h in the host logic, this value is set to 40h.	R
b31 to b8	—	Reserved	Don't care	R

23.3.4.9 Offset 3Ch Register (Interrupt Line, Interrupt Pin, Min gnt, Max Latency)

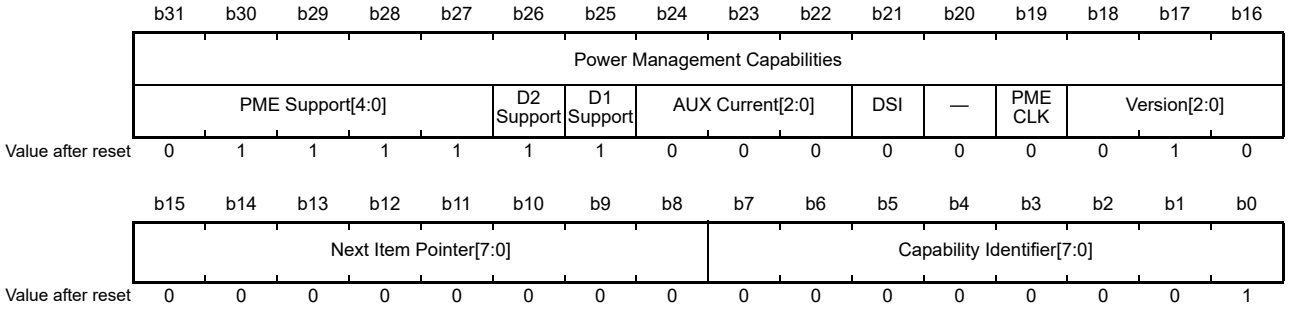
Address(es) A005 013Ch



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Interrupt Line [7:0]	Interrupt line	These bits indicate the interrupt line. In the USB host controller, do not change this from the value after reset (00h).	R/W
b15 to b8	Interrupt Pin [7:0]	Interrupt output pin	These bits indicate the pin for outputting interrupts. As INTB is used, this value is fixed to 02h.	R
b23 to b16	Min Gnt[7:0]	Minimum burst transfer time	These bits indicate the minimum burst transfer time. As the minimum time is set to 10h in the host logic, this value is 10h.	R
b31 to b24	Max Latency [7:0]	PCI bus maximum acquisition frequency	These bits indicate the maximum frequency of PCI bus acquisition. As the maximum frequency is set to 22h in the host logic, this value is set to 22h.	R

23.3.4.10 Offset 40h Register (Capability Identifier, Next Item Pointer, Power Management Capabilities)

Address(es) A005 0140h



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	Capability Identifier [7:0]	Power management register ID	These bits indicate the power management register ID. The value is fixed to 01h.	R
b15 to b8	Next Item Pointer[7:0]	Next item pointer	These bits are fixed to 00h, which indicates that the next item does not exist.	R
b18 to b16	Version[2:0]	PCI version	These bits are fixed to 010b, which indicates that the host logic conforms to the PCI Power Management Interface Specification Release 1.1.	R
b19	PME CLK	USBPCLKPCLK necessity	This bit is fixed to 0, which indicates that USBPCLKPCLK is not necessary for PME interrupt generation.	R
b20	—	Reserved	—	R
b21	DSI	Special initialization necessity	This bit is fixed to 0, which indicates that special initialization is not necessary for power management.	R
b24 to b22	Aux Current [2:0]	Current setting value	These bits indicate the required current for 3.3-V auxiliary power supply. As PME interrupt generation from the D3 cold state is not supported, this value is fixed to 000b.	R
b25	D1 Support	PCI power state D1 support	This bit is fixed to 1, which indicates that PCI power state D1 is supported.	R
b26	D2 Support	PCI power state D2 support	This bit is fixed to 1, which indicates that PCI power state D2 is supported.	R
b30 to b27	PME Support [4:0]	PME interrupt support bit field	These bits are fixed to 1111b, which indicates that PME interrupt generation is supported in all PCI power states (D0 to D3).	R
b31		D3 cold state support	This bit indicates whether the D3 cold state is supported. As the D3 cold state is not supported, this value is fixed to 0.	R

23.3.4.11 Offset 44h Register (Power Management Control/Status, PMCSR Bridge Support Extensions)

Address(es) A005 0144h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Data								PMCSR Bridge Support Extensions							
	Data[7:0]								BPCC Enable	B2_B3	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Power Management Control/Status															
	PME Status	Data Scale[1:0]		Data Select[3:0]				PME Enable	—	—	—	—	—	—	Power State[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	Power State [1:0]	PCI power status	These bits indicate the power state of the PCI as follows. b1 b0 0 0: D0 State 0 1: D1 State 1 0: D2 State 1 1: D3 hot State	R/W
b7 to b2	—	Reserved	When writing, write 0.	R/W
b8	PME Enable	PME enable	This bit specifies whether to use external pin PME. Setting this bit to 1 generates a PME interrupt when operation returns from the power management state.	R/W
b12 to b9	Data Select [3:0]	Data selection field	The value of these bits is 0h. This is an optional field in the PCI standard, and the host logic does not support it.	R
b14, b13	Data Scale [1:0]	Data scale field	The value of these bits is 00b. This is an optional field in the PCI standard, and the host logic does not support it.	R
b15	PME Status	PME interrupt status	This bit indicates the PME interrupt state. It is set to 1 when the PME generation condition is satisfied. Writing 1 from the PCI bus clears this bit to 0.	R/W
b21 to b16	—	Reserved	When writing, write 0.	R/W
b22	B2_B3	Bridge	The value of this bit is 0. This is a bit for the bridge and the host logic does not support it.	R
b23	BPCC Enable	BPCC enable	The value of this bit is 0. This is a bit for the bridge and the host logic does not support it.	R
b31 to b24	Data[7:0]	PCI standard option field	The value of these bits is 00h. This is an optional field in the PCI standard, and the host logic does not support it.	R

23.3.4.12 Offset 60h Register (SBRN, FLADJ, PORTWAKECAP)

Address(es) A005 0160h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PORTWAKECAP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FLADJ[7:0]								SBRN[7:0]							
Value after reset	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	SBRN[7:0]	Serial bus release number	These bits indicate the serial bus release number. The value is fixed to 20h.	R
b13 to b8	FLADJ[7:0]	1 micro frame length	These bits adjust the length of a micro-frame in 16-HS bit time units. The value after reset is 20h (60000d HS bit time).	R/W
b15, b14				R
b31 to b16	PORTWAKECAP[15:0]	Wakeup event mask field	From among the connected devices, these bits specify a mask for which port to be used in response to a wakeup event. This setting does not affect the actual operation of the host logic. The USB host controller only has one port so this register never needs be used.	R/W

23.3.4.13 Offset E0h Register (EXT1)

This register is the same as the EXT1 register located in the OHCI configuration space. See the description of the OHCI configuration register.

23.3.4.14 Offset E4h Register (EXT2)

This register is the same as the EXT2 register located in the OHCI configuration space. See the description of the OHCI configuration register.
Note that bit 0 (EHCI_mask) cannot be accessed from the EHCI side.

23.3.5 PCI Configuration Register for AHB-PCI Bridge

23.3.5.1 Offset 00h Register (Vendor ID, Device ID)

Address(es) A005 0000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DEVICE_ID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	VENDOR_ID[15:0]															
Value after reset	0	0	0	1	0	0	0	0	0	0	1	1	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	VENDOR_ID [15:0]	Vendor ID	This register indicates the vendor of the device. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host. It is always read as 1033h.	R
b31 to b16	DEVICE_ID [15:0]	Device ID	This register indicates the device type. This is used to select a driver that operates the device as specified in the PCI standard. This register does not need to be used in the embedded host.	R

23.3.5.2 Offset 04h Register (Command, Status)

Address(es) A005 0004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DETPERR	SIGSERR	REMABORT	RETABORT	SIGTABORT	DEVTIM[1:0]		MDPERR	FBTBCAP	—	CAP66M	CAPLIST	—	—	—	—
Value after reset	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	FBTBEN	SERREN	STEPCTR	PERREN	VGAPSNP	MWINVEN	SPECIALC	MASTEREN	MEMEN	IOEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	IOEN	I/O space access enable	This bit enables access to the I/O space. The value is fixed to 0.	R
b1	MEMEN	PCI slave operation setting	This bit specifies the PCI slave operation. In the USB host controller, set this bit to 1 during the initial settings. 0: Memory cycles cannot be accepted (value after reset) 1: Memory cycles can be accepted.	R/W
b2	MASTEREN	PCI master operation setting	This bit specifies the PCI master operation. In the USB host controller, set this bit to 1 during the initial settings. 0: Master operation is disabled (value after reset). 1: Master operation is enabled.	R/W
b3	SPECIALC	Special cycle enable	This bit enables the special cycle operation. The value is fixed to 0.	R
b4	MWINVEN	Memory write and invalidate enable	This bit enables the Memory Write and Invalidate command. The value is fixed to 0.	R
b5	VGAPSNP	VGA palette snoop enable	This bit enables VGA palette snooping. The value is fixed to 0.	R
b6	PERREN	Parity error detection operation setting	This bit specifies the operation when a parity error is detected. In the USB host controller, set this bit to 1 during the initial settings. 0: Nothing is to be done. (value after reset) 1: PERR# is asserted.	R/W
b7	STEPCTR	Address stepping control	This bit controls address stepping. The value is fixed to 0. (Address stepping is not done.)	R
b8	SERREN	System error detection operation setting	This bit specifies the operation when a system error is detected. In the USB host controller, set this bit to 1 during the initial settings. 0: Nothing is to be done. (value after reset) 1: SERR# is asserted.	R/W
b9	FBTBEN	Fast back to back enable	This bit enables fast back to back transactions. The value is fixed to 0.	R
b19 to b10	—	Reserved	When writing, write 0.	R/W
b20	CAPLIST	Capabilities list support	This bit indicates whether the capabilities list is supported. The value is fixed to 0. (The capabilities list is not supported.)	R
b21	CAP66M	66-MHz operation support	This bit indicates whether 66-MHz operation is supported. The value is fixed to 0. (66-MHz operation is not supported.)	R
b22	—	Reserved	When writing, write 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b23	FBTBCAP	Fast back to back capability	This bit indicates whether fast back to back transactions are supported. The value is fixed to 0. (Fast back to back transactions are not supported.)	R
b24	MDPERR	Parity error detection flag	This bit is set when a parity error is detected in master operation. Writing 1 clears this bit. 0: Parity error has not been detected. 1: Parity error has been detected.	R/W
b26, b25	DEVTIM[1:0]	DEVSEL response speed	These bits indicate the DEVSEL response speed. The value is fixed to 01b (Medium Mode).	R
b27	SIGTABORT	Slave Target Abort status	This is a status bit for Slave Target Abort. This bit is set when Target Abort is transmitted. Writing 1 clears this bit. 0: Target Abort has not been transmitted. 1: Target Abort has been transmitted.	R/W
b28	RETABORT	Master Target Abort status	This is a status bit for Master Target Abort. This bit is set when Target Abort is received. Writing 1 clears this bit. 0: Target Abort has not been received. 1: Target Abort has been received.	R/W
b29	REMABORT	Master Abort status	This is a status bit for Master Abort. This bit is set when Master Abort is received. Writing 1 clears this bit. 0: Master Abort has not been received. 1: Master Abort has been received.	R/W
b30	SIGSERR	SERR status	This is a status bit for SERR. This bit is set when a system error occurs. Writing 1 clears this bit. 0: SERR# has not been asserted. 1: SERR# has been asserted.	R/W
b31	DETPERR	Parity error status	This is a status bit for parity error. This bit is set when an address or data parity error is detected. Writing 1 clears this bit. 0: Parity error has not been detected. 1: Parity error has been detected.	R/W

23.3.5.3 Offset 08h Register (Revision ID, Class Code)

Address(es) A005 0008h

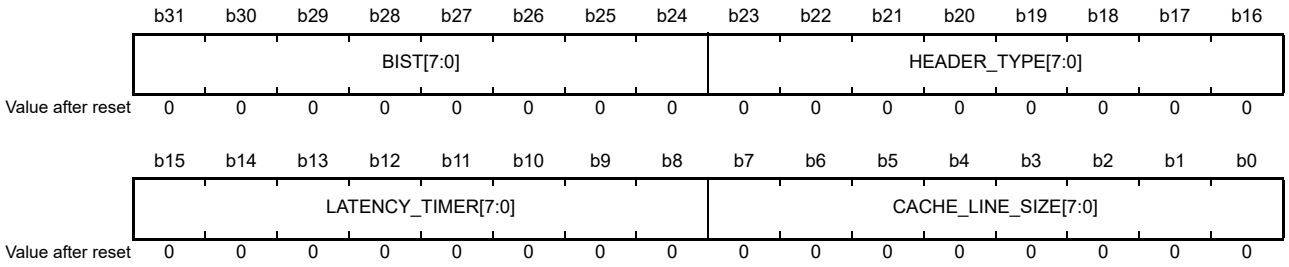
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CLASS_CODE[23:0]															
Value after reset	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CLASS_CODE[23:0]								REVISION_ID[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	REVISION_ID[7:0]	Revision ID	The value of these bits is 01h.	R
b31 to b8	CLASS_CODE[23:0]	CLASS CODE	The value of these bits is 060000h.	R

23.3.5.4 Offset 0Ch Register (Cache Line Size, Latency Timer, Header Type, BIST)

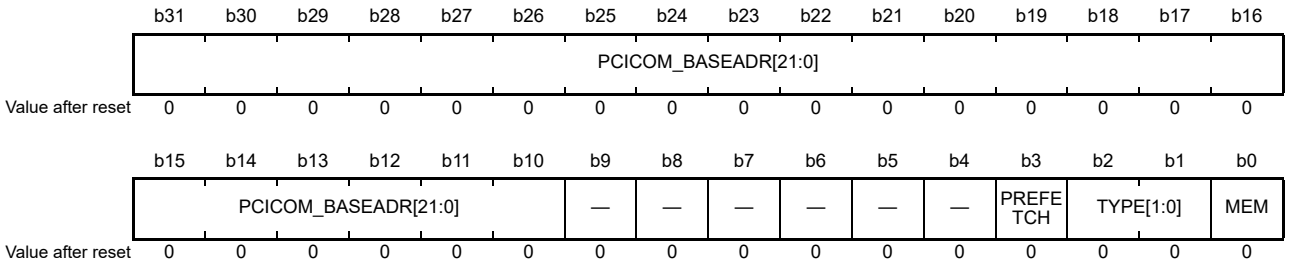
Address(es) A005 000Ch



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CACHE_LINE_SIZE[7:0]	CACHE LINE SIZE	The value of these bits is 00h (cache is not supported).	R
b15 to b8	LATENCY_TIMER[7:0]	Latency timer	These bits notify the system of the latency timer. The USB host controller does not use the latency timer; do not change this from the value after reset (00h).	R/W
b23 to b16	HEADER_TYPE[7:0]	HEADER TYPE	The value of these bits is 00h (single function device).	R
b31 to b24	BIST[7:0]	BIST	The value of these bits is 00h (BIST is not implemented).	R

23.3.5.5 Offset 10h Register (AHB-PCI Bridge Base Address)

Address(es) A005 0010h



Bit	Symbol	Bit Name	Description	R/W
b0	MEM	Base address specifying memory space	This bit indicates that the bits specified by the base address are in the memory space. The value is fixed to 0.	R
b2, b1	TYPE[1:0]	Base address type	These bits indicate the base address type. The value is 00b. (The base address can be allocated to any location in a 4-Gbyte space.)	R
b3	PREFETCH	Data prefetch capability	This bit indicates whether data can be prefetched. The value is fixed to 0. (Data cannot be prefetched.)	R
b9 to b4	—	Reserved	When writing, write 0.	R/W
b31 to b10	PCICOM_ BASEADR [21:0]	AHB-PCI bridge PCI communication register area base address	These bits specify the base address of the AHB-PCI bridge PCI communication register area. A 1-Kbyte area is necessary and the 24 high-order bits specify the base address.	R/W

23.3.5.6 Offset 14h Register (PCI-AHB WIN1 Base Address)

Address(es) A005 0014h

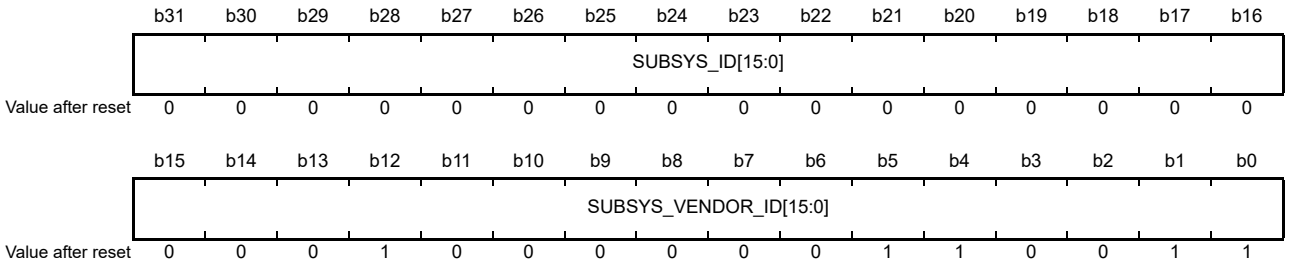
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PCI_WIN1_BASEADR [3:0]				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	PREFETCH	TYPE[1:0]		MEM
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MEM	Base address specifying memory space	This bit indicates that the field specified by the base address is in the memory space. The value is fixed to 0.	R
b2, b1	TYPE[1:0]	Base address type	These bits indicate the base address type. The value is 00b. (The base address can be allocated to any location in a 4-Gbyte space.)	R
b3	PREFETCH	Data prefetch capability	This bit indicates whether data can be prefetched. The value is fixed to 1. (Data can be prefetched.)	R
b27 to b4	—	Reserved	When writing, write 0.	R/W
b31 to b28	PCI_WIN1_ BASEADR[3:0]	PCI-AHB Window1 base address	These bits specify the base address of PCI-AHB Window 1. For the PCI-AHB Window 1 space, a 1-Gbyte area can be accessed through the setting of bits [11:10] (PCI_AHB_WIN1_SIZE) in the USBCTR register. **PCI-AHB Window 1 Space** / **AHB_BASEADR[31:28]** 1 Gbyte: The 2 higher-order bits [31:30] are the base address. Other bits are masked by 0. For details of register setting, see section 23.4.1, Register Access.	R/W

23.3.5.7 Offset 2Ch Register (Subsystem Vendor ID, Subsystem ID)

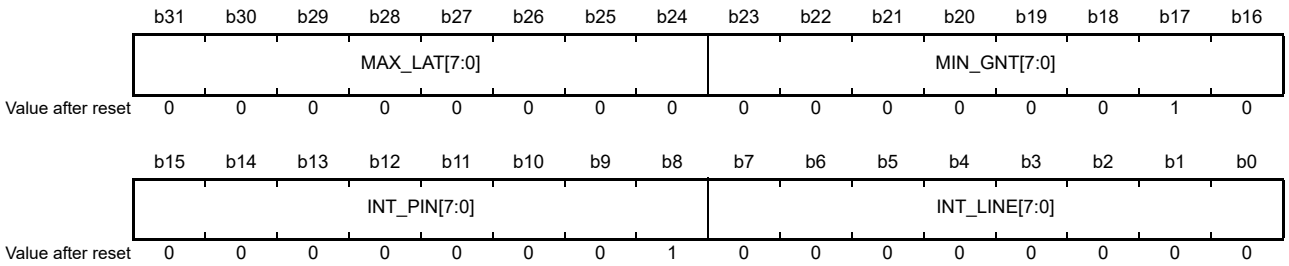
Address(es) A005 002Ch



Bit	Symbol	Bit Name	Description	R/W
b15 to b0	SUBSYS_ VENDOR_ID [15:0]	Subsystem vendor ID	The value of these bits is 1033h.	R
b31 to b16	SUBSYS_ID [15:0]	Subsystem ID	The value of these bits is 0000h.	R

23.3.5.8 Offset 3Ch Register (Interrupt Line, Interrupt Pin, Min gnt, Max Latency)

Address(es) A005 003Ch



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	INT_LINE[7:0]	Interrupt line	These bits specify the interrupt line. In the USB host controller, do not change this from the value after reset (00h).	R
b15 to b8	INT_PIN[7:0]	Interrupt pin field	The value of these bits is 01h. (INTA# is used).	R
b23 to b16	MIN_GNT[7:0]	Latency timer request field	The value of these bits is 02h. (Requested latency timer: 16 burst).	R
b31 to b24	MAX_LAT[7:0]	Bus use frequency request field	The value of these bits is 00h. (No value is specified for the bus use frequency.)	R

23.3.6 AHB-PCI Bridge PCI Communication Registers

23.3.6.1 PCIAHB_WIN1_CTR Register

This register is used to make settings for access from the host logic to AHB.

Address(es) A005 0800h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	AHB_BASEADR[3:0]				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PREFETCH[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	PREFETCH [1:0]	AHB bus prefetch setting bit field	These bits specify whether to prefetch data in the AHB bus when the host logic issues a read request. In the USB host controller, set these bits to 11b during the initial settings and do not change the value. b1 b0 0 0: Prefetch is disabled. 0 1: Prefetch is enabled (up to 4 bursts). 1 0: Prefetch is enabled (up to 8 bursts). 1 1: Prefetch is enabled (up to 16 bursts).	R/W
b27 to b2	—	Reserved	When writing, write 0.	R/W
b31 to b28	AHB_BASEADR [3:0]	AHB bus base address	These bits specify the base address of the AHB bus for access from the host logic to PCI-AHB Window 1. For the PCI-AHB Window 1 space, a 1-Gbyte area can be made accessible through the setting of bits [11:10] (PCI_AHB_WIN1_SIZE) in the USBCTR register. Do not modify this value from the value after reset (0000b). **PCI-AHB Window 1 Space** / **AHB_BASEADR[31:28]** 1 Gbyte / The 2 high-order bits (bits [31:30]) specify the base address. For details of register setting, see section 23.4.1, Register Access.	R/W

23.3.6.2 AHBPCI_WIN1_CTR Register

This register is used to make necessary settings for access to the PCI configuration space.

Address(es) A005 0810h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PCIWIN1_BASEADR[20:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	PCIWIN1_BASEADR[20:0]					—	—	—	—	—	—	—	PCICMD[2:0]			—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	When writing, write 0.	R/W
b3 to b1	PCICMD[2:0]	PCI bus cycle type	These bits specify the type of PCI bus cycle. In the USB host controller, set these bits to 101b during the initial settings and do not change the value. b3 b1 000: Interrupt Acknowledge / Special Cycle 001: IO Read / IO Write 011: Memory Read / Memory Write 101: Configuration Read / Configuration Write 110: Memory Read Multiple / Memory Write 111: Memory Read Line / Memory Write Setting is prohibited for other than the above.	R/W
b10 to b4	—	Reserved	When writing, write 0.	R/W
b31 to b11	PCIWIN1_BASEADR [20:0]	PCI bus base address	These bits specify the base address of the PCI bus for access from AHB to AHB-PCI Window 1. This register should be set when the PCI configuration space for the host logic or for the AHB-PCI bridge is accessed. For details of settings, see section 23.4.1.1, Access to PCI Configuration Registers.	R/W

23.3.6.3 AHBPCI_WIN2_CTR Register

This register is used to make necessary settings for access to the OHCI operational register area.

Address(es) A005 0814h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	PCIWIN2_BASEADR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	BURST_EN	—	PCICMD[2:0]			—
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	When writing, write 0.	R/W
b3 to b1	PCICMD[2:0]	PCI bus cycle type	These bits specify the type of PCI bus cycle. In the USB host controller, set these bits to 011b during the initial settings and do not change the value. b3 b1 001: IO Read / IO Write 011: Memory Read / Memory Write 110: Memory Read Multiple / Memory Write 111: Memory Read Line / Memory Write Setting is prohibited for other than the above.	R/W
b4	—	Reserved	When writing, write 0.	R/W
b5	BURST_EN	PCI bus burst transfer enable	This bit enables burst transfer to the PCI bus. In the USB host controller, clear to 0 during the initial settings and do not change the value. 0: Burst transfer is disabled. 1: Burst transfer is enabled.	R/W
b15 to b6	—	Reserved	When writing, write 0.	R/W
b31 to b16	PCIWIN2_BASEADR[15:0]	PCI bus base address	These bits specify the base address of the PCI bus for access from AHB to AHB-PCI Window 2. This register is used for access to the OHCI operational register area. For details of settings, see section 23.4.1.1, Access to PCI Configuration Registers.	R/W

23.3.6.4 PCI_INT_ENABLE Register

This register enables or disables each interrupt source indicated in the PCI_INT_STATUS register. When an interrupt source is disabled, the interrupt signal is not asserted even when the interrupt source is generated and the corresponding bit in the PCI_INT_STATUS register is set to 1.

Address(es) A005 0820h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	USBH_PMEEN	—	USBH_INTBEN	USBH_INTAEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PCIAHB_WIN2_INTEN	PCIAHB_WIN1_INTEN	—	—	—	—	—	—	RESERR_INTEN	SIGSERR_INTEN	PERR_INTEN	REMABORT_INTEN	RETABORT_INTEN	SIGTABORT_INTEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SIGTABORT_INTEN	SIGTABORT interrupt enable	This bit enables or disables the interrupt source indicated in bit 0 (SIGTABORT_INT) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b1	RETABORT_INTEN	RETABORT interrupt enable	This bit enables or disables the interrupt source indicated in bit 1 (RETABORT_INT) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b2	REMABORT_INTEN	REMABORT interrupt enable	This bit enables or disables the interrupt source indicated in bit 2 (REMABORT_INT) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b3	PERR_INTEN	PERR interrupt enable	This bit enables or disables the interrupt source indicated in bit 3 (PERR_INT) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b4	SIGSERR_INTEN	SIGSERR interrupt enable	This bit enables or disables the interrupt source indicated in bit 4 (SIGSERR_INT) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b5	RESERR_INTEN	RESERR interrupt enable	This bit enables or disables the interrupt source indicated in bit 5 (RESERR_INT) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b11 to b6	—	Reserved	When writing, write 0.	R/W
b12	PCIAHB_WIN1_INTEN	PCIAHB_WIN1 interrupt enable	This bit enables or disables the interrupt source indicated in bit 12 (PCIAHB_WIN1_INT) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b13	PCIAHB_WIN2_INTEN	PCIAHB_WIN2 interrupt enable	This bit enables or disables the interrupt source indicated in bit 13 (PCIAHB_WIN2_INT) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b15, b14	—	Reserved	When writing, write 0.	R/W
b16	USBH_INTAEN	USBH interrupt enable	This bit enables or disables the interrupt source indicated in bit 16 (USBH_INTA) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b17	USBH_ INTBEN	USBH interrupt enable	This bit enables or disables the interrupt source indicated in bit 17 (USBH_INTB) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b18	—	Reserved	When writing, write 0.	R/W
b19	USBH_ PMEEN	USBH_PME enable	This bit enables or disables the interrupt source indicated in bit 19 (USBH_PME) in PCI_INT_STATUS. 0: Disabled 1: Enabled	R/W
b31 to b20	—	Reserved	When writing, write 0.	R/W

23.3.6.5 PCI_INT_STATUS Register

This register indicates the state of interrupt sources in the AHB-PCI bridge and the state of interrupt signals from the host logic.

Address(es) A005 0824h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	USBH_PME	—	USBH_INTB	USBH_INTA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	PCIAHB_WIN2_INT	PCIAHB_WIN1_INT	—	—	—	—	—	—	RESERR_INT	SIGSERR_INT	PERR_INT	REMABORT_INT	RETABORT_INT	SIGTABORT_INT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SIGTABORT_INT	Target Abort notification	This bit indicates that Target Abort has been sent in PCI target operation. Writing 1 clears this bit. 0: Target Abort has not been sent. 1: Target Abort has been sent.	R/W
b1	RETABORT_INT	Target Abort reception	This bit indicates that Target Abort has been received in PCI master operation. Writing 1 clears this bit. 0: Target Abort has not been received. 1: Target Abort has been received.	R/W
b2	REMABORT_INT	MasterAbort reception	This bit indicates that Master Abort has been received in PCI master operation. Writing 1 clears this bit. 0: Master Abort has not been received. 1: Master Abort has been received.	R/W
b3	PERR_INT	PERR# Input/Output interrupt source status	This bit indicates the state of the interrupt source caused by PERR# input/output. Writing 1 clears this bit. 0: PERR# has not been asserted. 1: PERR# has been asserted.	R/W
b4	SIGSERR_INT	SERR# Output interrupt source status	This bit indicates the state of the interrupt source caused by SERR# output. Writing 1 clears this bit. 0: SERR# has not been asserted. 1: SERR# has been asserted.	R/W
b5	RESERR_INT	SERR# Input interrupt source status	This bit indicates the state of the interrupt source caused by SERR# input. Writing 1 clears this bit. 0: SERR# assertion has not been detected. 1: SERR# assertion has been detected.	R/W
b11 to b6	—	Reserved	When writing, write 0.	R/W
b12	PCIAHB_WIN1_INT	AHB bus error occurrence flag	This bit indicates that an AHB bus error has occurred in PCIAHB Window 1. Writing 1 clears this bit. 0: AHB bus error has not occurred. 1: AHB bus error has occurred.	R/W
b13	PCIAHB_WIN2_INT	AHB bus error occurrence flag	This bit indicates that an AHB bus error has occurred in PCIAHB Window 2. Writing 1 clears this bit. 0: AHB bus error has not occurred. 1: AHB bus error has occurred.	R/W
b15, b14	—	Reserved	When writing, write 0.	R/W

Bit	Symbol	Bit Name	Description	R/W
b16	USBH_INTA	Host logic INTA# interrupt status	This bit indicates the state of the INTA# interrupt from the host logic. The interrupt can be cleared from the host logic. 0: INTA interrupt has not occurred. 1: INTA interrupt has occurred.	R
b17	USBH_INTB	Host logic INTB# interrupt status	This bit indicates the state of the INTB# interrupt from the host logic. The interrupt can be cleared from the host logic. 0: INTB interrupt has not occurred. 1: INTB interrupt has occurred.	R
b18	—	Reserved	When writing, write 0.	R/W
b19	USBH_PME	Host logic PME# interrupt status	This bit indicates the state of the PME# interrupt from the host logic. The interrupt can be cleared from the host logic. 0: PME interrupt has not occurred. 1: PME interrupt has occurred.	R
b31 to b20	—	Reserved	When writing, write 0.	R/W

23.3.6.6 AHB_BUS_CTR Register

This register specifies the AHB master and slave functions of the host logic.

Address(es) A005 0830h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SMODE_READY_CTR	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	MMODE_HBUSREQ	—	—	—	—	MMODE_WR_INCR	MMODE_BYTE_BURST	MMODE_HTRANS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MMODE_HTRANS	HTRANS signal operation mode setting	This bit specifies the operating mode for the HTRANS signal in AHB master operation. In the USB host controller, set this bit to 1 during the initial settings and do not change the value. 0: For divided cycles, NONSEQ is output continuously. 1: For divided cycles, IDLE is inserted and the bus is requested again through HBUSREQ.	R/W
b1	MMODE_BYTE_BURST	Burst transfer control	This bit controls burst transfer for 16-bit or 8-bit transfer in AHB master operation. In the USB host controller, set this bit to 1 during the initial settings and do not change the value. 0: Burst transfer is enabled for 16-bit or 8-bit transfer. 1: Burst transfer is disabled for 16-bit or 8-bit transfer.	R/W
b2	MMODE_WR_INCR	Condition setting of using undefined-length burst transfer	This bit specifies the condition of using an undefined-length burst for write transfer in AHB master operation. In the USB host controller, set this bit to 1 during the initial settings and do not change the value. 0: INCR is always used except when the transfer count is 4, 8, or 16. 1: INCR4, INCR8, and INCR16 are generally used, and INCR is used only when the remaining data is 2 to 3 beats.	R/W
b6 to b3	—	Reserved	When writing, write 0.	R/W
b7	MMODE_HBUSREQ	HBUSREQ deassertion timing setting	This bit specifies the timing of HBUSREQ deassertion in AHB master operation. In the USB host controller, set this bit to 1 during the initial settings and do not change the value. 0: Deasserted in the last address phase of the cycle. 1: Deasserted with the first timing of HGRANT = 1 and HREADY = 1.	R/W
b16 to b8	—	Reserved	When writing, write 0.	R/W
b17	SMODE_READY_CTR	Wait operation control	This bit controls the wait operation in AHB slave operation. Set this bit to 0 in a system that uses RETRY or SPLIT. Do not modify this value except for initial setting. 0: Wait is controlled with HRESP = RETRY. 1: Wait is controlled with HREADY = 0.	R/W
b31 to b18	—	Reserved	When writing, write 0.	R/W

23.3.6.7 USBCTR Register

This register is used to make settings of the host logic.

Address(es) A005 0834h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PCI_AHB_WIN1_SIZE[1:0]		PCI_AHB_WIN2_EN	—	—	—	—	—	—	—	PCICLK_MASK	USBH_RST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	USBH_RST	Reset signal control	This bit controls the reset signal supplied to the host logic. The host logic can be accessed a maximum of 3 USBPCLK clock cycles after the reset is canceled. 0: Host logic reset is canceled. 1: Host logic reset is issued.	R/W
b1	PCICLK_MASK	PCI clock supply mask	This bit stops the PCI clock supply in the host logic. Note that the host logic cannot be accessed when this bit is set to 1. 0: PCI clock is supplied. 1: PCI clock is stopped.	R/W
b8 to b2	—	Reserved	When writing, write 0.	R/W
b9	PCI_AHB_WIN2_EN	PCI-AHB Window2 enable	This bit enables operation of PCI-AHB Window 2. For details, see section 23.4.1, Register Access. 0: PCI-AHB Window 2 is not available. Settings other than the above are prohibited. Operation cannot be guaranteed if the setting of this bit is modified.	R/W
b11, b10	PCI_AHB_WIN1_SIZE[1:0]	PCI-AHB Window1 area	These bits control the PCI-AHB Window 1 area. For details, see section 23.4.1, Register Access. Do not modify this value except for initial setting. b11 b10 10: 1 Gbyte	R/W
b31 to b12	—	Reserved	When writing, write 0.	R/W

23.3.6.8 PCI_ARBITER_CTR Register

This register is used to make settings of the PCI bus arbitration function.

Address(es) A005 0840h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	PCIBP_MODE	—	—	—	—	—	—	—	—	—	—	PCIREQ1	PCIREQ0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	PCIREQ0	PCI bus request signal mask 0	This bit enables or disables the PCI bus request signal from this unit. In the USB host controller, set this bit to 1 during the initial settings and do not change the value. 0: Request signal is disabled. 1: Request signal is enabled.	R/W
b1	PCIREQ1	PCI bus request signal mask 1	This bit enables or disables the PCI bus request signal from the host logic. In the USB host controller, set this bit to 1 during the initial settings and do not change the value. 0: Request signal is disabled. 1: Request signal is enabled.	R/W
b11 to b2	—	Reserved	When writing, always write 0.	R/W
b12	PCIBP_MODE	Master setting	This bit specifies the master while the PCI bus is in the parked state. In the USB host controller, set this bit to 1 during the initial settings and do not change the value. 0: This unit is the master in the bus-parked state. 1: The master that made the last access is the master in the bus-parked state.	R/W
b31 to b13	—	Reserved	Do not change this from the value after reset.	R/W

23.3.6.9 PCI_UNIT_REV Register

This register indicates the version of the AHB-PCI bridge macro.

Address(es) A005 0848h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	Major Revision ID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	Minor Revision ID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	Minor Revision ID [15:0]	Minor revision ID	These bits indicate the minor revision ID of this unit.	R
b31 to b16	Major Revision ID [15:0]	Major revision ID	These bits indicate the major revision ID of this unit.	R

23.4 Register Access

23.4.1 Register Access

Registers are accessed through the internal PCI bus. To access them from the AHB bus correctly, appropriately specify the mapping between the memory space for the AHB bus and that for the internal PCI bus in the USB host controller (see section 23.8.1.1, Sample of Initial Settings). Note that the PCI space is divided into two: the PCI memory space where data transfer is actually done, and the PCI configuration space where PCI bus transfer settings and the settings of the base addresses in the PCI memory space are stored.

Access from the AHB to the host logic and master access from the host logic to the AHB are done through the window areas in the AHB-PCI bridge. Figure 23.3 and Table 23.5 show the functions and relationship of register and window areas.

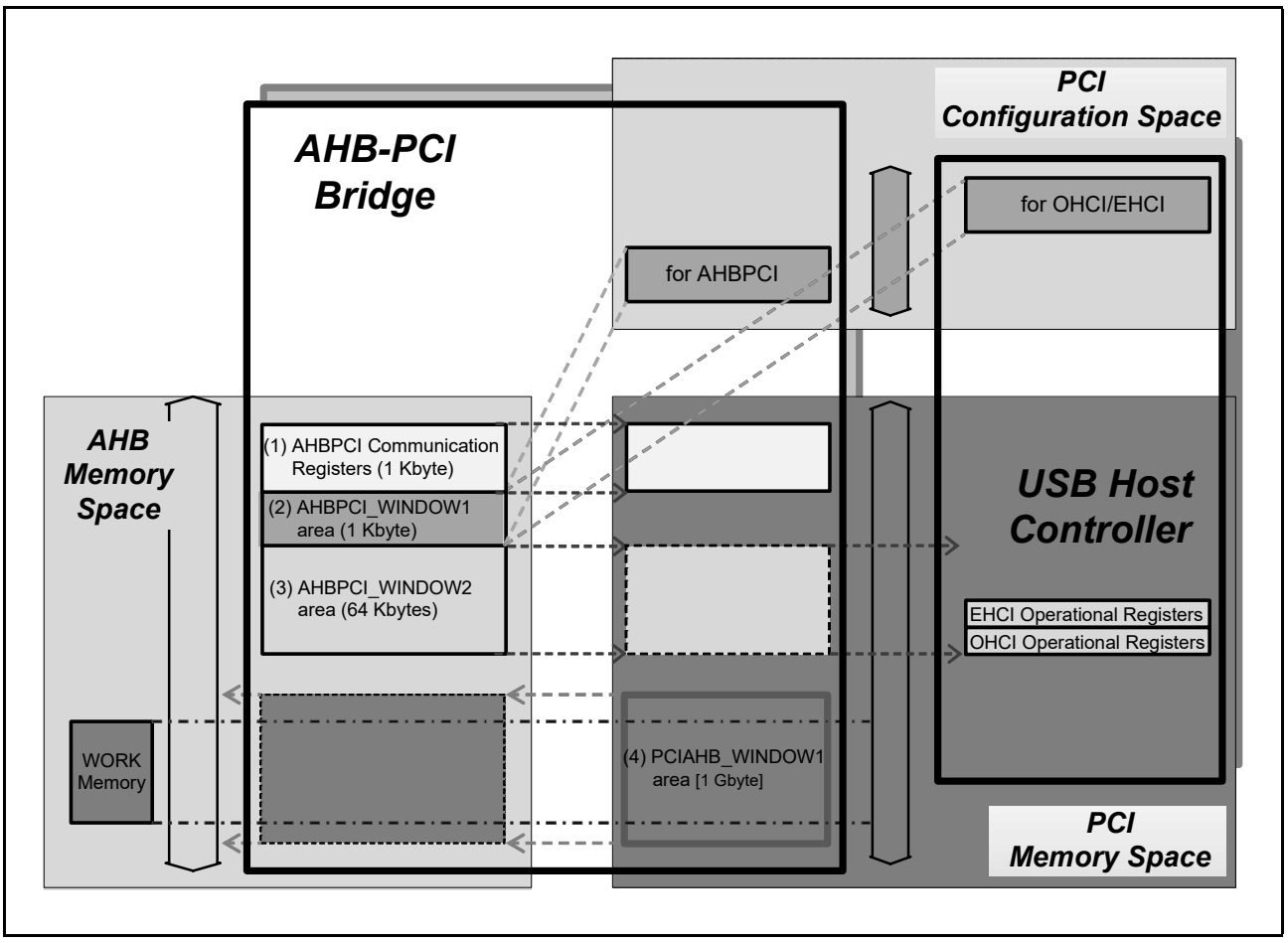


Figure 23.3 Image of AHB and PCI Space Mapping

Table 23.5 Descriptions of Areas

AHB Area Name	Size	Description
(1) AHBPCI communication registers	1 Kbyte	Various AHB settings are stored and the base address of each window area is specified. This area is also allocated in the PCI memory space; make sure that it does not overlap with other areas.
(2) AHBPCI_WINDOW1 area	1 Kbyte	The PCI configuration registers are accessed through this area. Whether to access an OHCI/EHCI configuration register or a register in the AHB-PCI bridge is switched through the AHBPCI_WIN1_CTR register setting.
(3) AHBPCI_WINDOW2 area	64 Kbytes	The OHCI/EHCI operational registers are accessed through this area.
(4) PCIAHB_WINDOW1 area	1 Gbyte	The host logic accesses the work memory on the AHB bus through this area. The size can be selected through the USBCTR register.

Make sure that the AHBPCI communication registers and the AHB-PCI window 2 area (OHCI/EHCI operational registers) do not overlap with the PCIAHB window 1 area in the PCI memory space.
An easy way to do this is to allocate the areas to the same addresses in both the AHB memory space and the PCI memory space.
However, when the above areas overlap due to the memory map for the AHB bus, use the PCI configuration registers (OHCI/EHCI/AHB-PCI Base Address registers) to avoid overlap with the PCI-AHB window 1 area. Figure 23.4 shows an image of mapping in such cases.

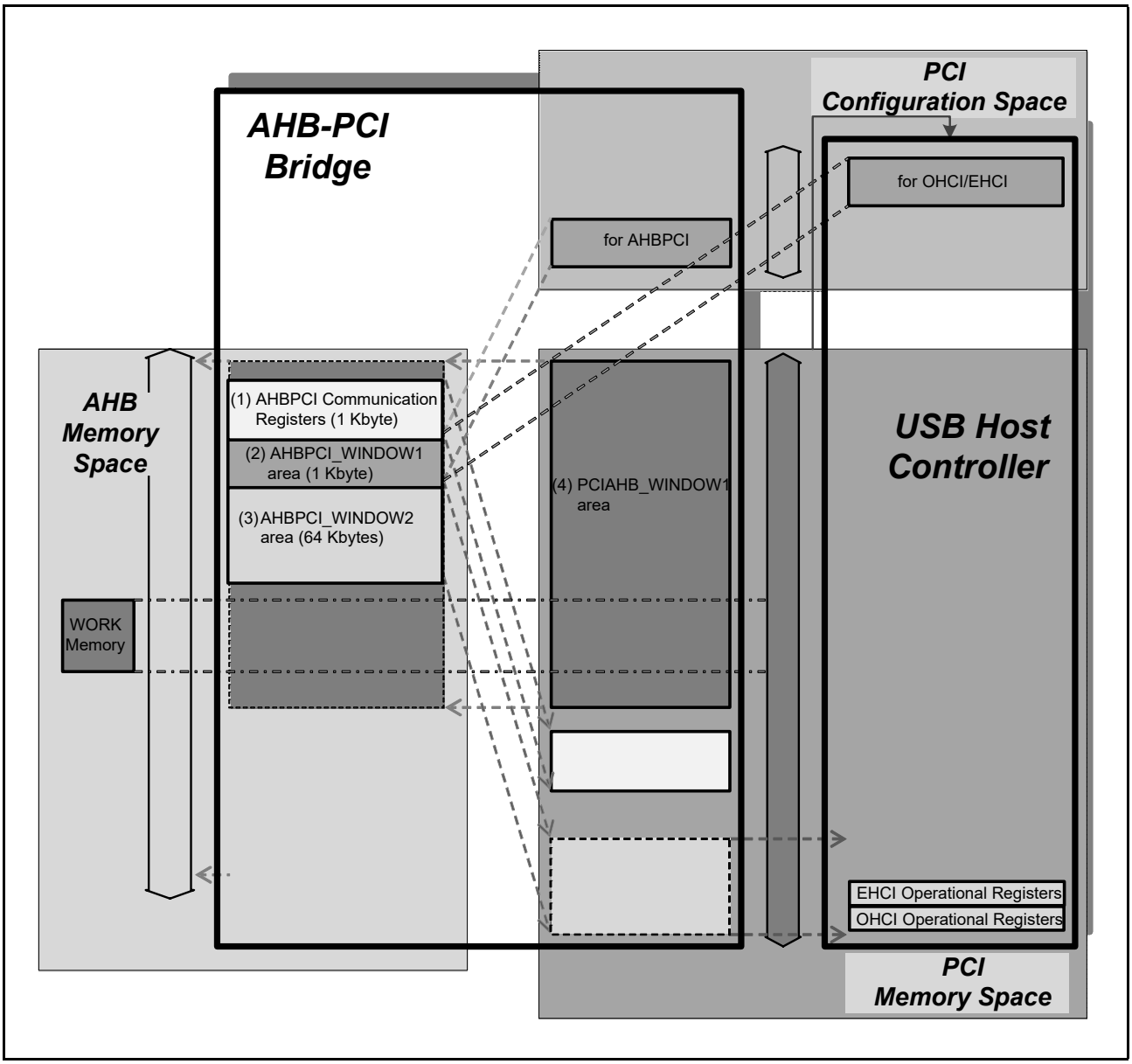


Figure 23.4 Image of AHB and PCI Space Mapping (when Areas Overlap)

Figure 23.5 and Table 23.6 show the registers used for AHB and PCI space mapping and the relationship between the settings in the registers and the mapping in the spaces.

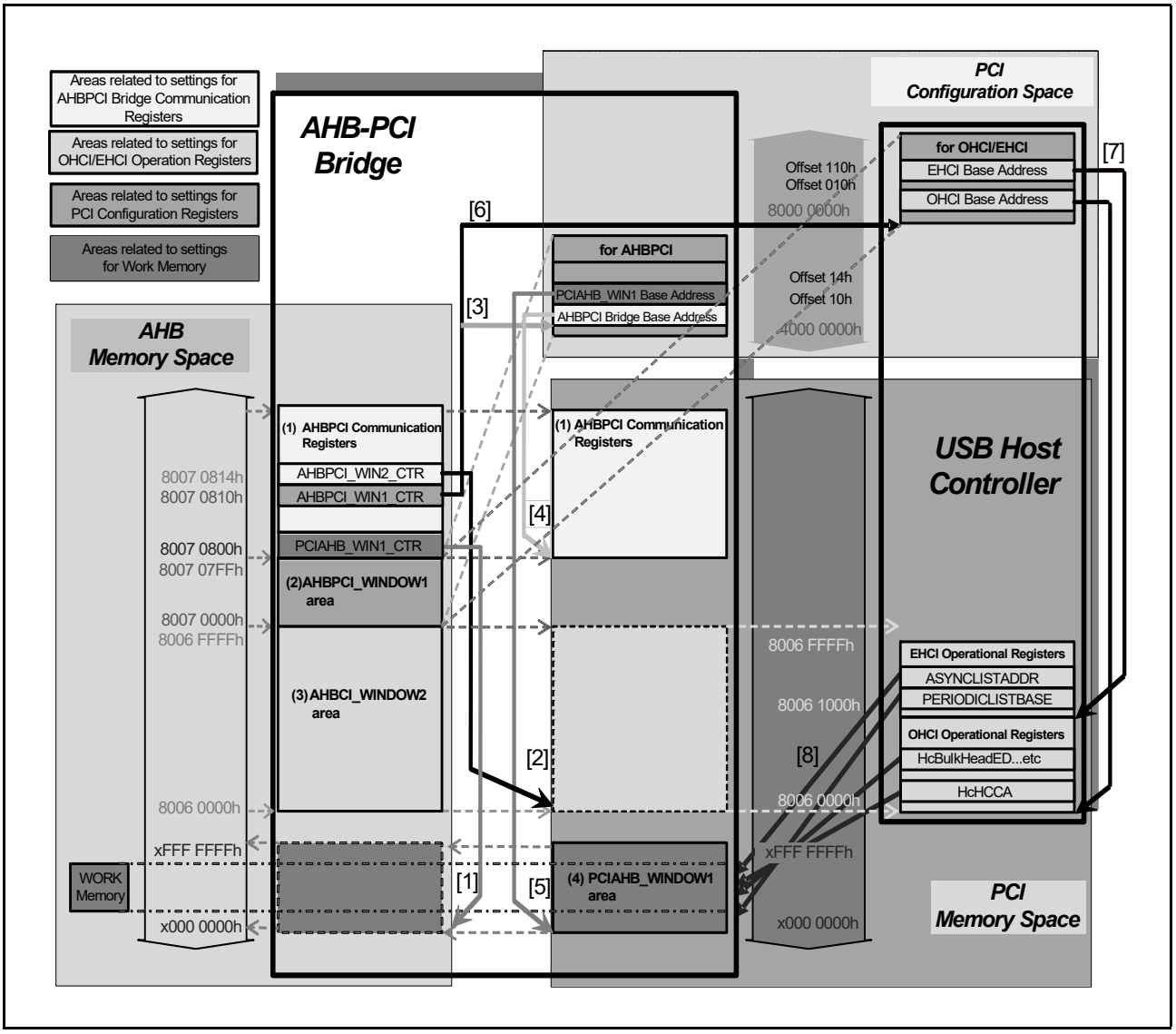


Figure 23.5 Relationship between the Address Setting in the Registers and AHB and PCI Space Mapping

Table 23.6 Descriptions of Address Settings in Registers

Register	Description
[1] PCIAHB_WIN1_CTR	When the host logic performs master access to the PCI-AHB window 1 area, the AHB bus address is converted to the base address specified in this register. In general usage, specify an area including the work memory to be used.
[2] AHBPCI_WIN2_CTR	When the AHB-PCI window 2 area is accessed, the PCI bus address is converted to the base address specified in this register. In general usage, this register can be set to the same value as that for the AHB-PCI window 2 area. However, if this area overlaps the area specified in [1] (the area including work memory), specify an appropriate address to avoid overlap.
[3] AHBPCI_WIN1_CTR	When the base address is set to 4000_0000h in this register, the PCI configuration registers for AHB-PCI are accessed.
[4] AHBPCI Bridge Base Address	This register specifies the base address of the AHB-PCI bridge in the PCI space. This register is not accessed from the PCI bus, but specify an appropriate address to avoid overlap with other areas.
[5] PCIAHB_WIN1 Base Address	This register specifies the base address of the PCI-AHB window 1 area in the PCI space. In general usage, specify the same base address as [1].
[6] AHBPCI_WIN1_CTR	When the base address is set to 8000_0000h in this register, the PCI configuration registers for OHCI/EHCI are accessed.
[7] OHCI/EHCI Base Address	This register specifies the base address of the OHCI/EHCI operational registers in the PCI space. In general usage, the OHCI base address is specified as the same value as specified in [2]. The EHCI base address is set to a value obtained by adding an offset of 1000h to the base address specified in [2].
[8] Various OHCI/EHCI Operational Registers	After settings of [1] to [7] are completed, the host logic can access data (such as descriptors) expanded in the work RAM on the AHB bus through the PCI. The following registers specify the addresses of data stored in work RAM. • OHCI/EHCI operational registers – HcHCCA register – HcPeriodicCurrentED register – HcControlHeadED register – HcControlCurrentED register – HcBulkHeadED register – HcBulkCurrentED register – HcDoneHead register • EHCI operational registers – PERIODICLISTBASE register – ASYNCLISTADDR register

23.4.1.1 Access to PCI Configuration Registers

The registers in the PCI configuration spaces are accessed through the AHB-PCI window 1 area (addresses 10000h to 107FFh: 2-Kbyte space). Before access, make appropriate settings in the AHBPCI_WIN1_CTR register. The following shows the settings of the AHBPCI_WIN1_CTR register to access the PCI configuration spaces for OHCI/EHCI and AHB-PCI bridge.

Table 23.7 AHBPCI_WIN1_CTR Register Setting

	AHBPCI_WIN1_CTR Register Setting	
Area to be Accessed	**PCIWIN1_BASEADR[31:11]**	**PCICMD[2:0]**
OHCI/EHCI PCI Configuration Space	Only bit 31 is set to 1.	101b
AHB-PCI Bridge PCI Configuration Space	Only bit 30 is set to 1.	

23.4.1.2 Access to OHCI/EHCI Operational Registers

When OHCI/EHCI operational registers are accessed, it is necessary to make settings in the OHCI/EHCI PCI configuration register and the AHBPCI_WIN2_CTR register as well as PCI space address mapping. The following shows the necessary settings.

Table 23.8 Necessary Settings for Access to OHCI/EHCI Operational Registers

Locations of Settings	**Settings**
OHCI/EHCI PCI configuration apace Offset 04h (Command, Status) Bit1 (Memory space)	1 (Access to memory space is enabled)
AHBPCI_WIN2_CTR register bit[3:1] PCICMD[2:0]	011b (Memory read or memory write)

23.5 Clock Lines

23.5.1 Externally Supplied Clocks

The following three clock signals are supplied from within the chip to the USB host controller of the EC-1.

Table 23.9 List of Externally Supplied Clocks

Clock Signal	Function	Frequency
PCLKA	AHB clock AHB clock for power management	150 MHz
USBMCLK	Reference clock	50 MHz
USBPCLK	PCI clock inside the USB host controller	30 MHz

23.5.2 Clock Distribution Diagram

Figure 23.6 shows the clock distribution diagram of this system.

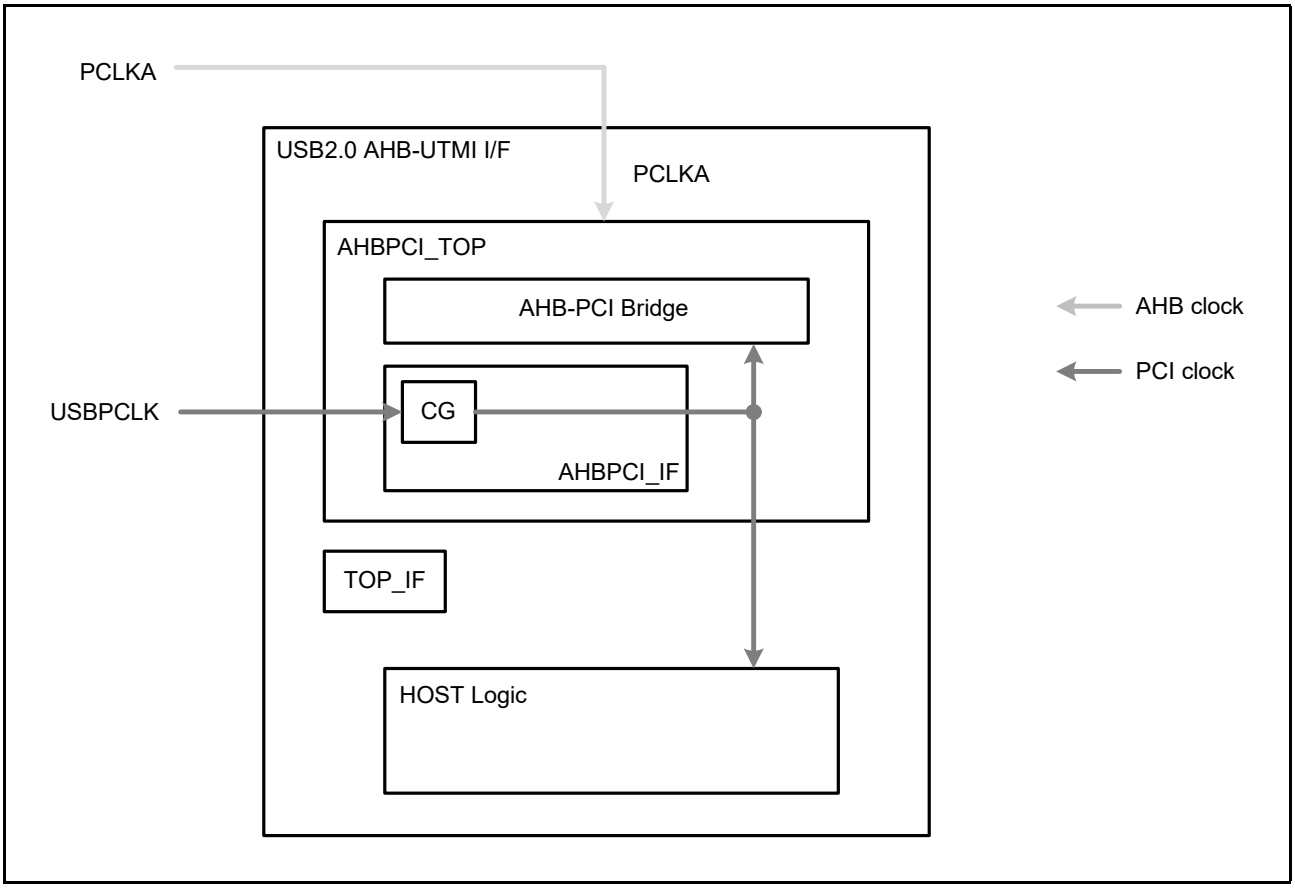


Figure 23.6 Clock Distribution Diagram

23.6 Interrupts

23.6.1 Interrupt Control Registers

23.6.1.1 U2H_INT Control Registers

U2H_INT is an interrupt signal generated by the AHB-PCI bridge. Registers in the AHB-PCI bridge are used to check and clear the interrupt state and to enable interrupts.

Table 23.10 U2H_INT Control Registers

Operation	Control Register
Check and clear interrupt state	PCI_INT_STATUS register
Enable interrupts	PCI_INT_ENABLE register

23.6.1.2 U2H_OHCI_INT Control Registers

U2H_OHCI_INT is the INTA interrupt signal from the host logic. Each interrupt is controlled basically through registers in the host logic, but to assert the interrupt signal, the interrupt enable bit in the AHB-PCI bridge should be set.

Table 23.11 U2H_OHCI_INT Control Registers

Operation	Control Register
Check and clear interrupt state	HcInterruptStatus register
Enable interrupts	HcInterruptEnable register HcInterruptDisable register PCI_INT_ENABLE register (bit 16 (USBH_INTAEN))

23.6.1.3 U2H_EHCI_INT Control Registers

U2H_EHCI_INT is the INTB interrupt signal from the host logic. Each interrupt is controlled basically through registers in the host logic, but to assert the interrupt signal, the interrupt enable bit in the AHB-PCI bridge should be set.

Table 23.12 U2H_EHCI_INT Control Registers

Operation	Control Register
Check and clear interrupt state	USBSTS register
Enable interrupts	USBINTR register PCI_INT_ENABLE register (bit 17 (USBH_INTBEN))

23.6.1.4 U2H_PME_INT Control Registers

U2H_PME_INT is the PME interrupt signal from the host logic. Each interrupt is controlled basically through registers in the host logic, but to assert the interrupt signal, the interrupt enable bit in the AHB-PCI bridge should be set.

Table 23.13 U2H_PME_INT Control Registers

Operation	Control Register
Check and clear interrupt state	PCI Configuration register for OHCI/EHCI offset 44h
Enable interrupts	PCI Configuration register for OHCI/EHCI offset 44h PCI_INT_ENABLE register (bit 19 (USBH_PMEEN))

23.6.2 U2H_BIND_INT

The U2H_BIND_INT interrupt signal is generated as the logical OR of four interrupt source signals (U2H_INT, U2H_OHCI_INT, U2H_EHCI_INT, and U2H_PME_INT).
The states of the U2H_PME_INT, U2H_OHCI_INT, and U2H_EHCI_INT signals are reflected in the PCI_INT_STATUS register in the AHB-PCI bridge. Read the register to check which source was responsible for a generated interrupt.

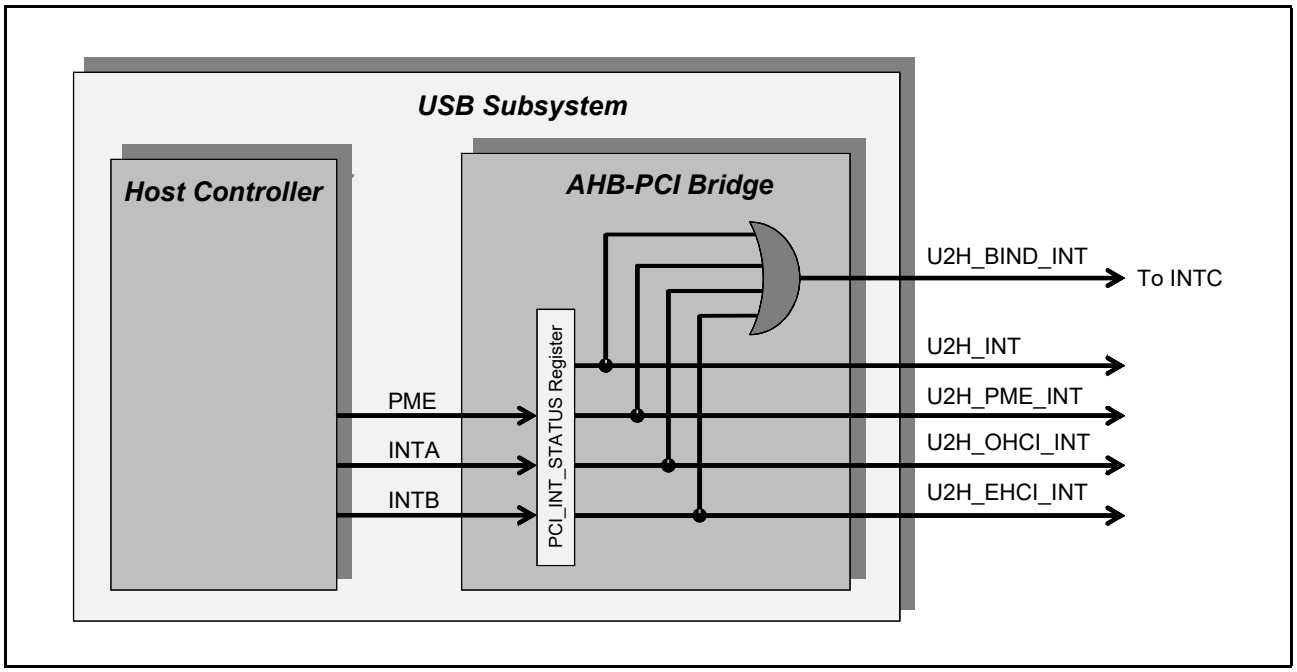


Figure 23.7 Image of Interrupt Signal Integration

23.6.3 Time Required to Clear Interrupt Signals

An interrupt generated in the USB host controller may be falsely detected multiple times because, depending on the access state on the internal bus, it may take a long time to actually clear the interrupt after the register for clearing the interrupt source is written to (see Figure 23.8 Time Required to Clear an Interrupt) due to the posted write operation of the AHB bridge.
A measure should be taken to prevent false detection before the next interrupt is detected after the register access for clearing the interrupt is completed.
For example, after accessing the register for clearing the interrupt source (1), access any register in the host logic (2). Register access (2) waits (SHREADY/MHREADY = 0 or RETRY response) until (1) is completed, and the interrupt is cleared without fail when register access (2) is completed.

(1) Host Logic
It may take time to clear a U2H_OHCI_INT, U2H_EHCI_INT, or U2H_PME_INT interrupt after the register for clearing the interrupt is accessed.
This usually takes approximately 300 ns when PCLKA is 150 MHz, but in the worst case when master transfer is in progress on the internal PCI bus, this takes 36 USBPCLK cycles + 3 PCLKA cycles + two 12-MHz clock cycles; this is approximately 1.4 µs when USBPCLK is 30 MHz.

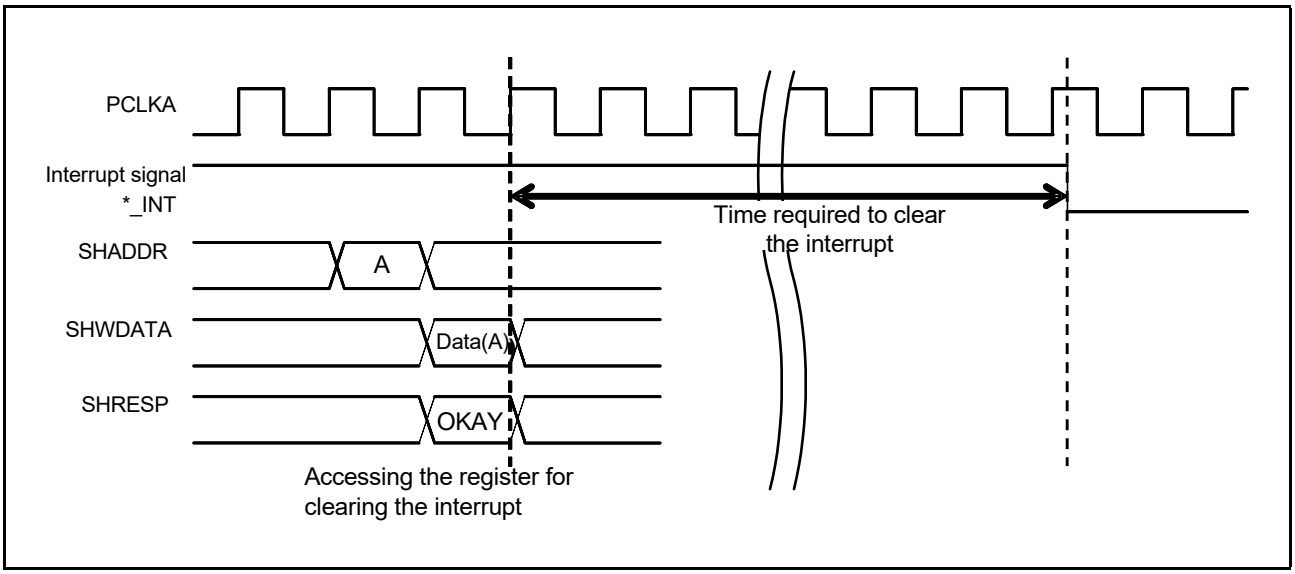


Figure 23.8 Time Required to Clear an Interrupt

23.7 Overcurrent Control and VBUS Control

23.7.1 Overcurrent Control

This section describes the operation of the USB_OVRCUR and USB_VBUSEN signals used for overcurrent detection at the USB port and VBUS control.

23.7.1.1 USB_OVRCUR and USB_VBUSEN Signal Functions

Table 23.14 shows the meaning of the USB_OVRCUR and USB_VBUSEN signals.

Table 23.14 USB_OVRCUR and USB_VBUSEN

Pin	I/O	Level	Meaning
USB_OVRCUR	Input	1	No overcurrent has been detected.
		0	Overcurrent has been detected.
USB_VBUSEN	Output	1	Power supply to VBUS is on.
		0	Power supply to VBUS is off.

23.7.1.2 Conditions for Asserting and Deasserting USB_VBUSEN Output Signal

Figure 23.9 is a timing chart for USB_OVRCUR and USB_VBUSEN assertion and deassertion.

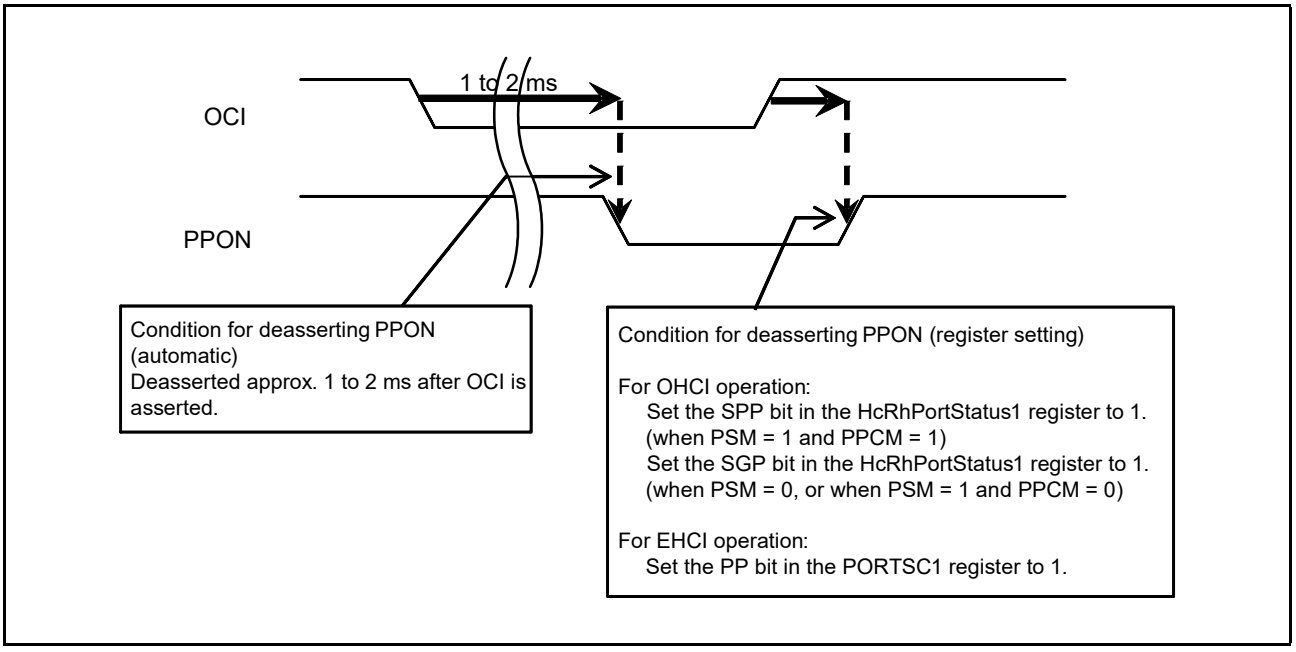


Figure 23.9 Timing Chart for USB_OVRCUR and USB_VBUSEN Assertion and Deassertion

After USB_OVRCUR is deasserted, USB_VBUSEN is not asserted automatically. USB_VBUSEN is asserted when software sets the port power bit after USB_OVRCUR is deasserted.

23.7.2 VBUS Control

When the USB port is not used, VBUS can be stopped to reduce power consumption by connecting the USB_OVRCUR and USB_VBUSEN pins to the high-side switch (although this may not be possible depending on the connections with peripheral circuits).
The relationship between USB_VBUSEN and VBUS is shown below.

Table 23.15 Relationship between USB_VBUSEN and VBUS

USB_VBUSEN	VBUS
0	Stopped
1	Operated

The USB_VBUSEN operation when USB_OVRCUR is asserted depends on the settings in the PCI configuration register and OHCI operational registers as shown below.

Table 23.16 Register Settings and USB_VBUSEN Operation

OHCI Operational Registers				
HcRhDescriptorA Register			HcRhDescriptorB Register	USB_VBUSEN Output Pin Operation
NOCP Bit	NPS Bit	PSM Bit	PPCM[1] Bit	
1	—	—	—	Fixed to 1
—	1	—	—	Fixed to 1
0	0	0	—	Deasserted (0) when USB_OVRCUR is asserted (0)
		1	0	
			1	

Note: When NPS = 1, the host logic detects an overcurrent but does not deassert USB_VBUSEN.

23.7.3 Initial Settings of PPON1

ON and OFF of PPON1 (PortPower) is controlled by the port control register of OHCI/EHCI operational registers listed below in general usage after reset without setting the NOCP and NPS bits in the HcRhDescriptorA register. PPON1 is turned ON when at least one of the following bits (SPP, SGP, and PP) is ON.
In general, the following bits (SPP, SGP, and PP) are cleared to 0 when an overcurrent is detected.

Register		Bit	Symbol
OHCI operation	HcRhPortStatus1 register	8	SPP
	HcRhPortStatus register	16	SGP*1
EHCI operation	PORTSC1 register	12	PP

Note 1. This is not generally used with the single-port version.
For PPON control by this bit, the bit must be in the state of PSM = 0, or PSM = 1 and PPCM = 0.

PPON1 is turned on by setting the NOCP or NPS bit to 1 regardless of the Port Power bit of the above ports.
The initial settings of the PPON1 control bits (SPP, SGP, PP, NOCP, and NPS) mentioned above are all 0 (OFF). Setting these bits to 1 (ON) during initial setting, the procedure in the following flow is required.

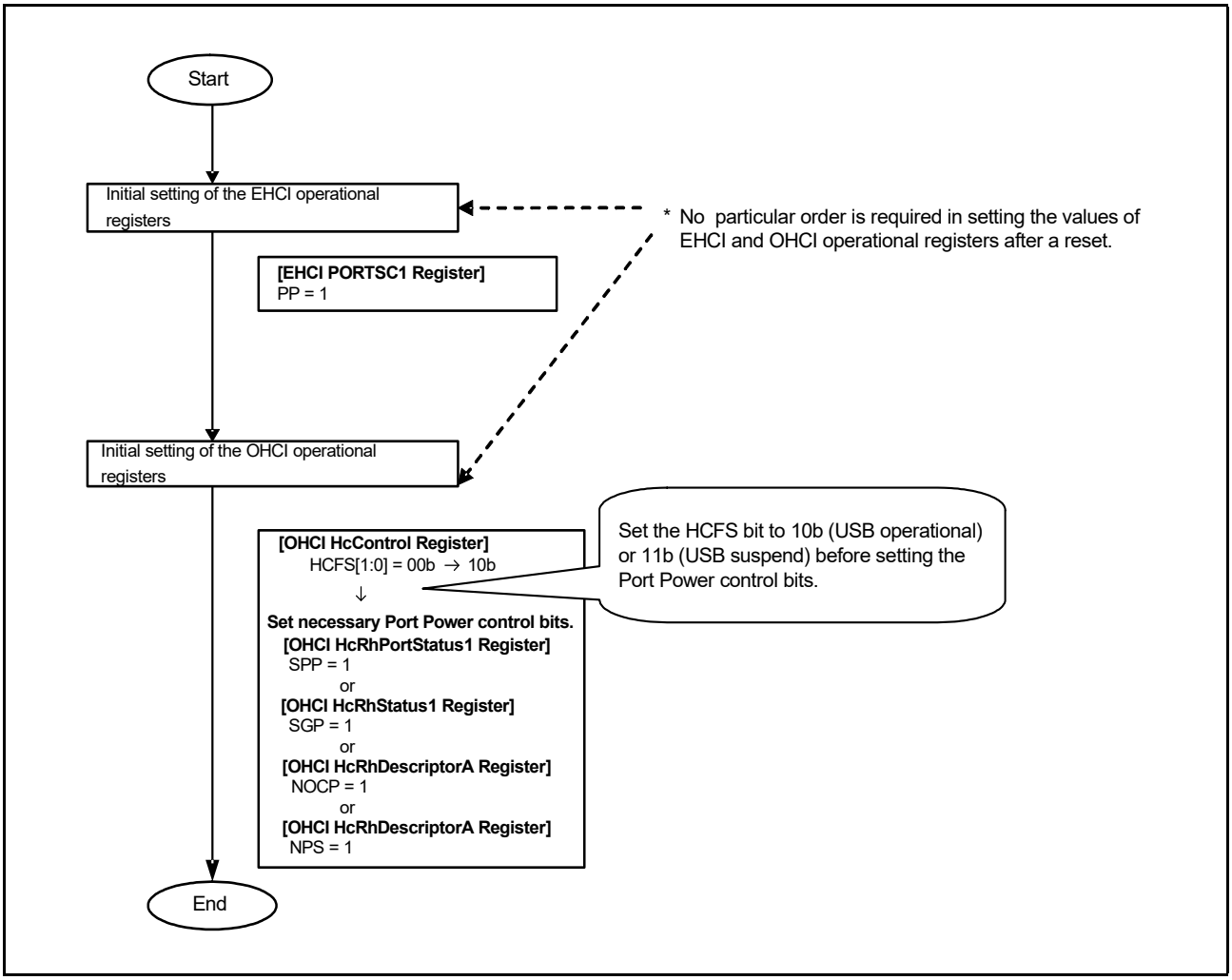


Figure 23.10 Flow of PPON1 Initial Settings

23.7.4 Procedure for USB_VBUSEN Control Regarding Overcurrent Detection

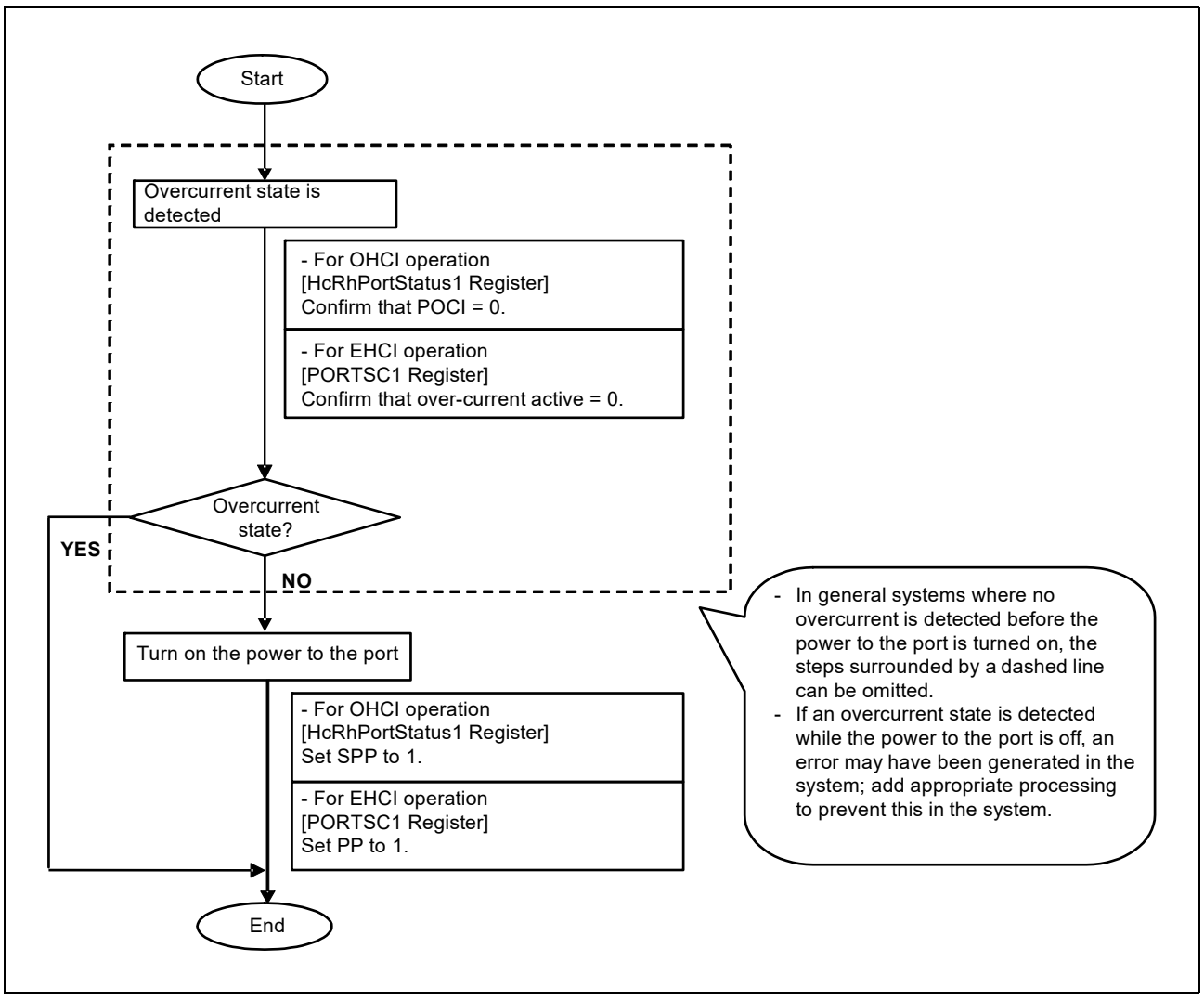


Figure 23.11 Flow for USB_VBUSEN Control Regarding Overcurrent Detection

23.7.5 Procedure for USB_VBUSEN Setting

The following shows the procedure for USB_VBUSEN setting in a system where USB_OVRCUR may be active (0) at system start-up.

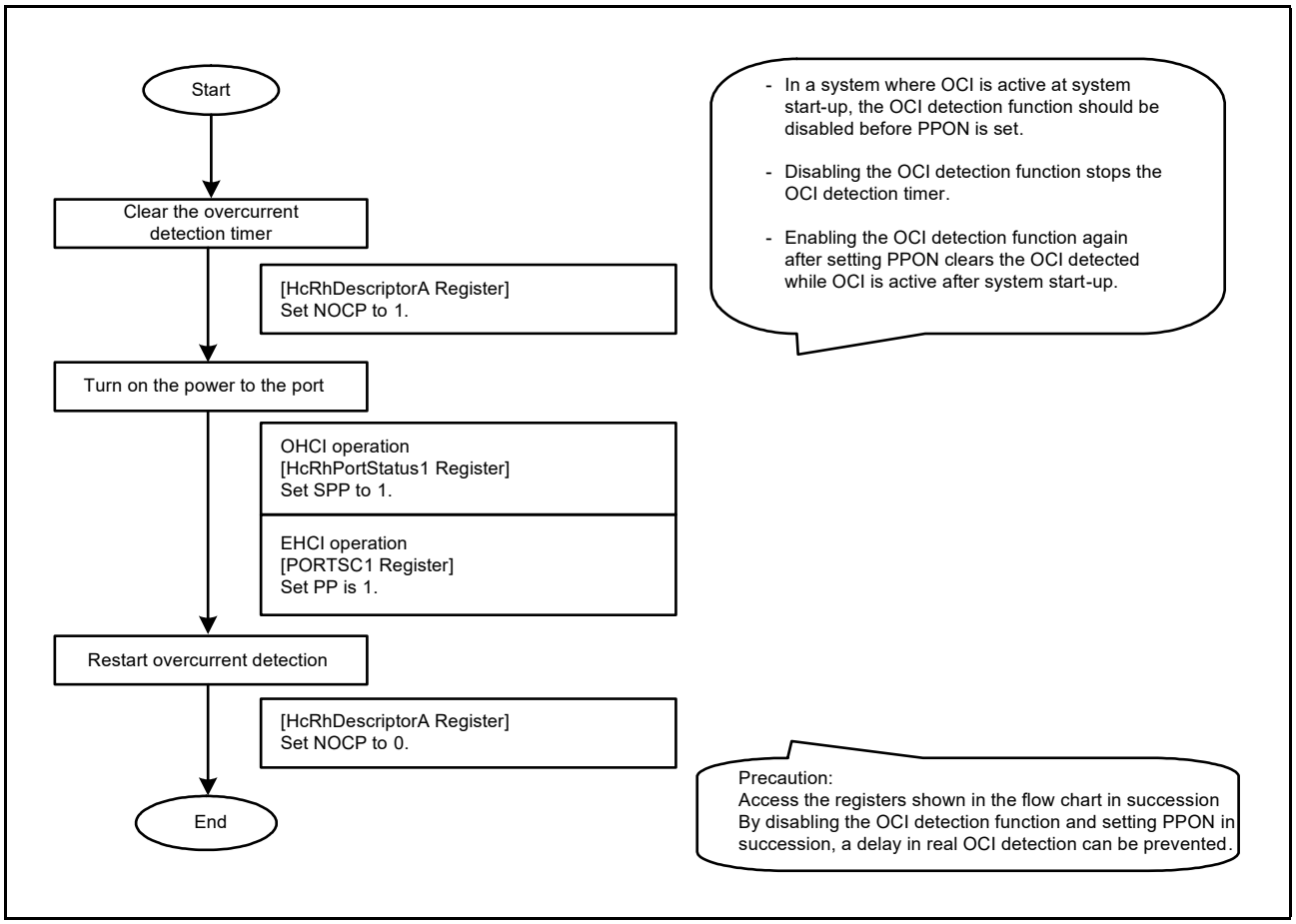


Figure 23.12 Flow for USB_VBUSEN Setting

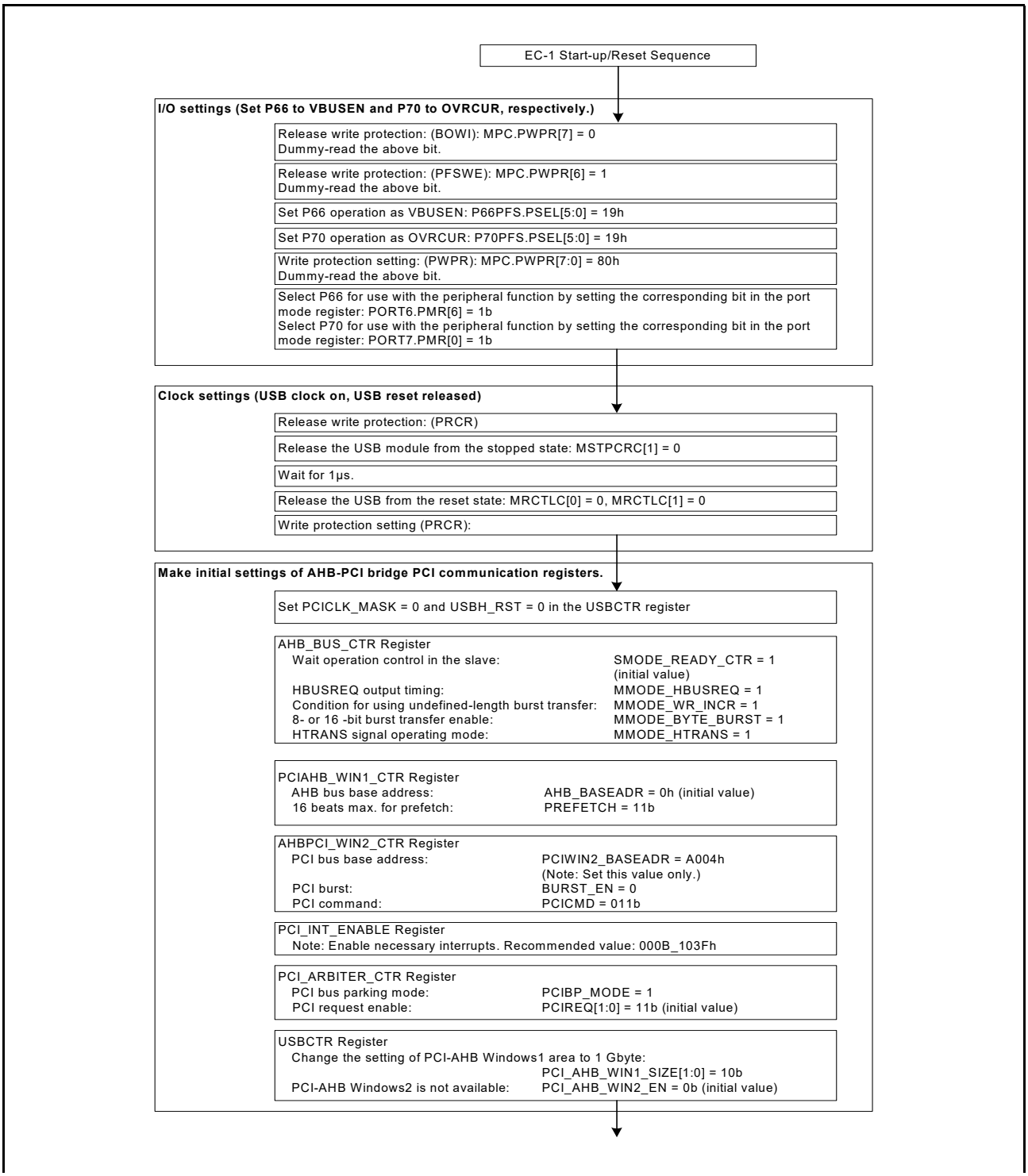
23.8 Operating Procedures

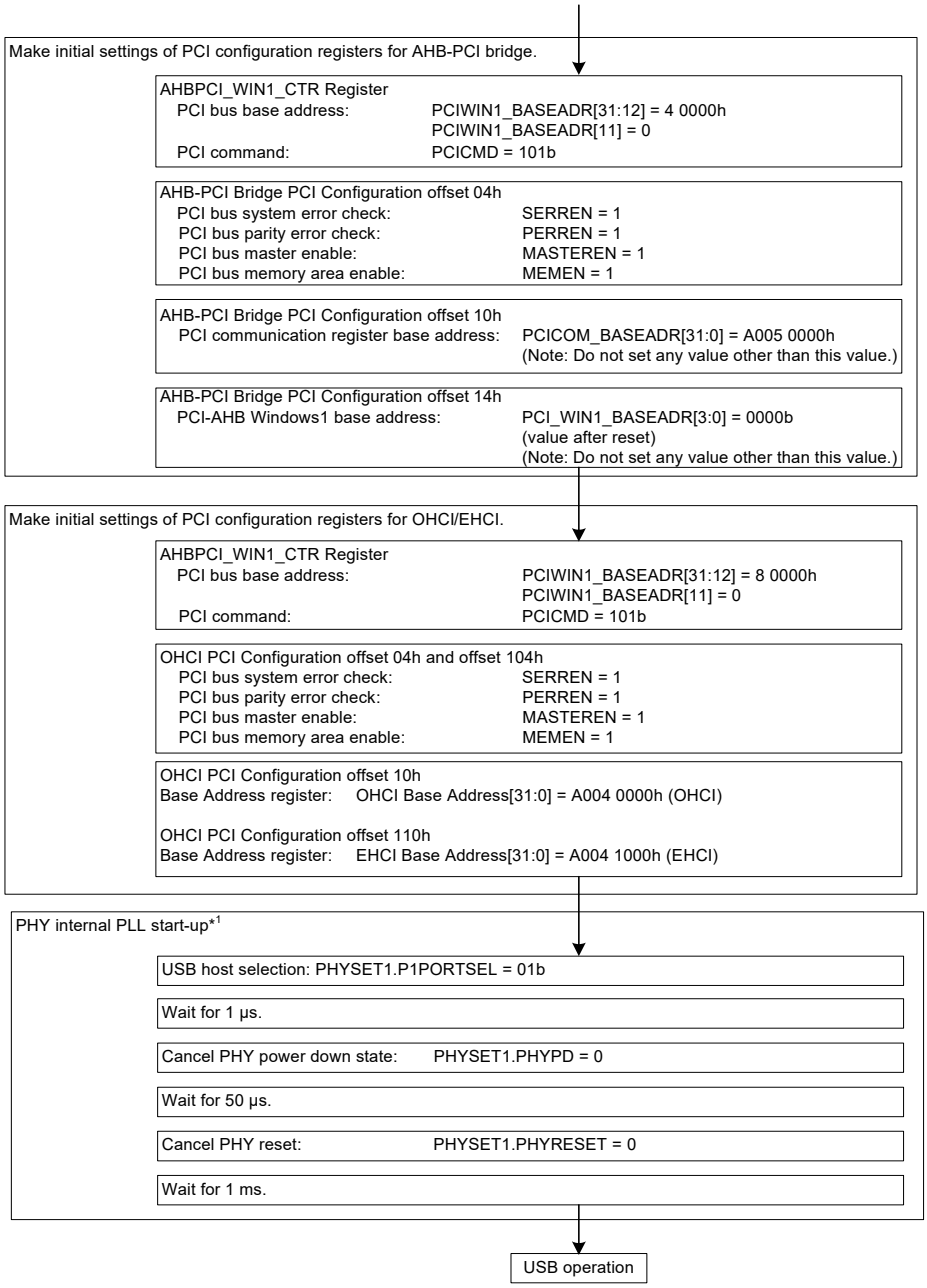
23.8.1 Initial Setting Sequence

23.8.1.1 Sample of Initial Settings

This section shows sample initial settings to implement the following functions.

- Access to OHCI/EHCI operational registers through AHB-PCI window 2 registers.
- Data transfer from the host logic to the AHB bus.





Note 1. A halt of the operation of the PHY internal PLL by setting PHY Power-Down (the PHYSET1.PHYPD bit is 1) and PHY reset (PHYSET1.PHYRESET bit is 1) is prohibited once the PHY internal PLL is activated.

Figure 23.13 Initial Setting Sequence

23.8.2 USB Host Transfer Procedure

For the USB host transfer procedure, see the following OHCI and EHCI specifications.

- Open Host Controller Interface Specification for USB Rev 1.0a
- Enhanced Host Controller Interface Specification for Universal Serial Bus Revision 1.0

This section gives only supplementary information regarding DMA stop.

23.8.2.1 Stopping DMA Transfer

The AHB-PCI Bridge core does not have a function for starting or stopping DMA transfer. The PCI bus cycle started by the host logic working as the master is output to the AHB bus as a DMA transfer without change.
DMA transfer is done in the following two cases.

(1) The current frame number is written to memory.
(2) The descriptors and data expanded in memory are read or written to for list processing.

Note: The frame number is automatically written at frame cycle intervals when the USB state is operational.

To stop DMA transfer, shift the USB state to Suspend or Reset.
To suspend only the list processing in (2), clear the bit for enabling list processing (BLE, CLE, IE, or PLE bit in the HcControl register); the list processing stops at the next frame.

24. USB 2.0 HS Function Module (USBf)

24.1 Overview

The USB module of this LSI is a dual-role device that has USB 2.0 host and function modules. However, it does not have a facility for detecting the ID and so does not support On-the-Go (OTG) functionality.
This LSI handles a single USB port for either host controller or function controller operation. The port connection path select input signal bits (PHYSET1.P1PORTSEL[1:0]) are used to switch between host controller and function controller operation.
Since operation as a host controller or a function controller are exclusive of each other, dynamic switching between the types of operation is not possible.

This section describes function controller operation.

The USBf module supports high-speed transfer and full-speed transfer defined by the USB 2.0 specification.

This LSI supports all transfer types that are defined in the USB specification. It incorporates an 8-Kbyte buffer memory for data transfer and can use up to 10 pipes. Pipes 1 to 9 can be assigned an arbitrary endpoint number according to the user system. The controller is equipped with a dedicated DMA interface as a local bus interface which is independent of the CPU bus interface, making it suitable for systems requiring high-speed, large-capacity data transfer.

Table 24.1 Specifications of the USB Module (1 / 2)

Item	Description
For High-Speed USB	• Built-in USB function controller
For all types of USB transfer	• Support for all types of USB transfer, including isochronous • Control transfer • Bulk transfer • Interrupt transfer (high-bandwidth transfers not supported) • Isochronous transfer (high-bandwidth transfers not supported)
Bus interface	• Two DMA interfaces available - The interface with the DMAC is selectable and DMA transfer is independent of the CPU bus interface • High-speed data transfer for access to the internal FIFOs at 60 Mbytes/second (when the bus width is 32 bits)
Pipe configuration	• 8 Kbytes of buffer memory for USB communications • Up to 10 pipes can be selected (including the default control pipe) • Programmable pipe configuration • Any endpoint number can be assigned to pipes 1 to 9 • Transfer conditions that can be set for each pipe are listed below. Pipe 0: Control transfer, single buffer fixed at 64 bytes Pipes 1 and 2: Bulk transfer or isochronous transfer can be selected, continuous transfer mode, programmable buffer size (specifiable as up to 2 Kbytes, double buffer is also specifiable) Pipes 3 to 5: Bulk transfer, continuous transfer mode, programmable buffer size (specifiable as up to 2 Kbytes, double buffer is also specifiable) Pipes 6 to 9: Interrupt transfer, single buffer fixed at 64 bytes
Features of function controller operation	• High-speed (480 Mbps) and full-speed transfer (12 Mbps) are supported • Automatic recognition of high-speed operation or full-speed operation based on automatic response to the reset handshake • Control transfer stage monitoring • Monitoring of device state • Automatic response to SET_ADDRESS requests • NAK response interrupt function (NRDY) • SOF interpolation

Table 24.1 Specifications of the USB Module (2 / 2)

Item	Description
Other functions	• Judging the end of transfer on the basis of a transaction count • Adjustable timing for BRDY interrupt event notification (BFRE) • Automatic clearing of the buffer memory after the data for the pipe specified in the DxFIFO port has been read (DCLRM) • NAK setting for response PID generated by end of transfer (SHTNAK) • Only self-powered mode is supported (bus-powered mode is not supported).

Figure 24.1 is a block diagram of the USB module.

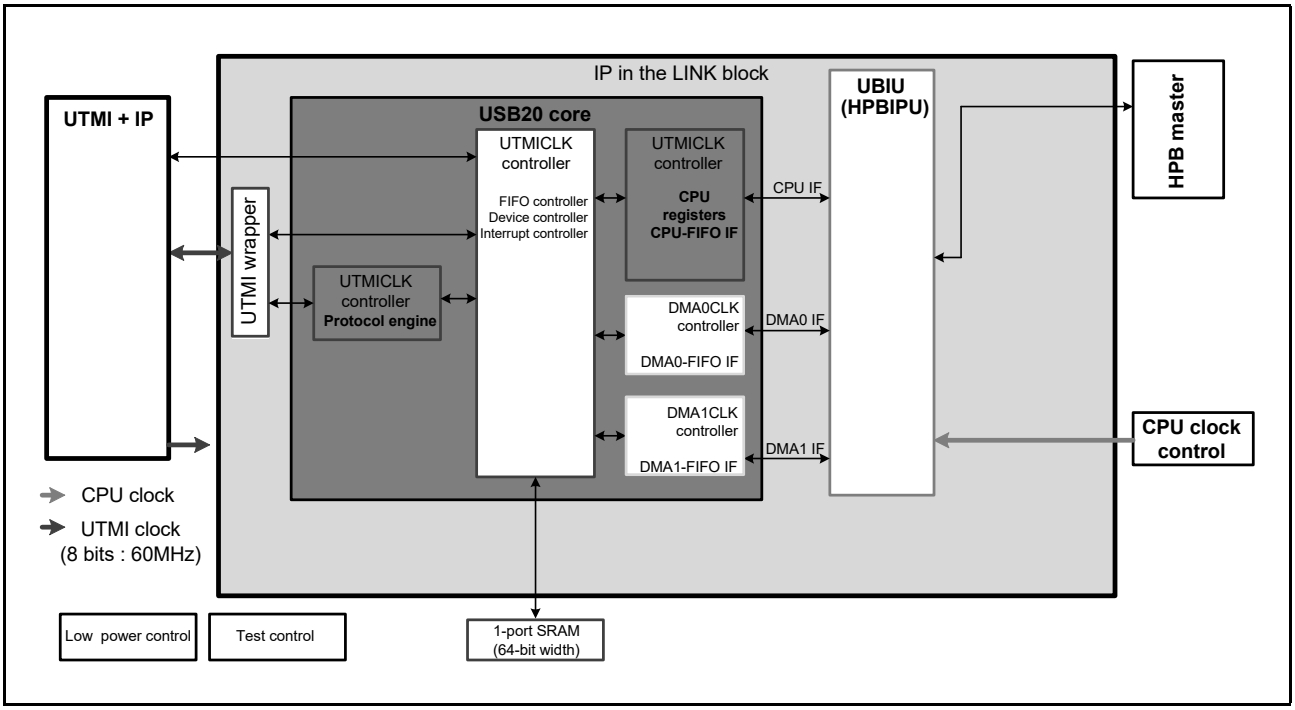


Figure 24.1 Block Diagram of the USB Module

Data transfer between this controller and the host controller connected to the USB bus uses buffer memory assigned to each pipe. For communication between this controller and the host controller, this controller converts the data stored in buffer memory into USB data packets and outputs them to the USB bus serially, and also inputs data packets from the USB bus and stores them in buffer memory.

24.1.1 Functional Overview

24.1.1.1 Identification of the USB Transfer Speed

The hardware can automatically identify the USB transfer speed.

24.1.1.2 Bus interface

(1) How to access the FIFO buffer memory

The following two types of access are available for the FIFO buffer memory for USB data transfers. To read from or write to the FIFO buffer memory, access (read or write) the FIFO ports from the CPU (DMAC).

1. CPU access
   Specify a FIFO port address and write data to the FIFO buffer memory or read data from the FIFO buffer memory.
2. DMA access
   Specify a FIFO port address through the DMAC in the CPU or the dedicated DMAC and write data to the FIFO buffer memory or read data from the FIFO buffer memory.

USB data transfer is done in little endian. The byte endian swap function is available for FIFO port access; for 16-bit or 32-bit access, the endian can be switched through register settings.

24.1.1.3 USB Event

This controller sends an interrupt to the user system to notify an event in USB operation. This controller asserts the UCL_Dx_DREQ (interrupt source [43] or [44]) signal to notify that the pipe selected for the DMA interface has become ready for access.

Sending interrupts for notification can be enabled or disabled separately for each interrupt type and source through software settings.

24.1.1.4 USB Data Transfer

This controller performs all types of USB data transfer: control transfer, bulk transfer, interrupt transfer, and isochronous transfer. The following numbers of pipes are available for each transfer type.
One pipe dedicated to control transfer
Four pipes dedicated to interrupt transfer
Three pipes dedicated to bulk transfer
Two pipes selectively used for bulk transfer or isochronous transfer

Each pipe should be set up as necessary for the desired USB transfer in accordance with the user system; for example, transfer type, endpoint number, or maximum packet size.

This controller has an 8-byte buffer memory. For pipes dedicated for bulk transfer or selective pipes for bulk transfer or isochronous transfer, buffer memory assignment, buffer operating mode setting, or other necessary settings should be made in accordance with the user system. By setting the buffer operating mode, such as double buffer structure or continuous data packet transfer mode, fast data transfer is achieved with fewer interrupts.

CPU and DMA controller access to the buffer memory is made through three FIFO port registers.

24.1.1.5 Functions for Access from Direct Memory Access Controller (DMAC)

This controller provides two channels of the DMA interface having the following functions.
(1) Notification of the end of transfer using a transfer end signal
(2) Automatic clearing of the FIFO buffer when a zero-length packet is received
(3) Judging the end of transfer using the transaction counter

24.2 Register Descriptions

24.2.1 System Configuration Control

24.2.1.1 System Configuration Control Register 0 (SYSCFG0)

Address(es): A006 0000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	HSE	—	DRPD	DPRPU	—	—	—	USBE
Value after hardware reset:	x	x	x	x	x	x	x	0	0	0	1	0	x	x	x	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b0	USBE	USB Block Operation Enable	Enables or disables operation of the USB block. 0: USB block operation is disabled. 1: USB block operation is enabled.	R/W	R
b3 to b1	—	Reserved	When writing to these bits, write 0.	R/W	—
b4	DPRPU	D+ Line Resistor Control	Enables or disables pulling up the D+ line in function controller operation. 0: Pulling up the line is disabled. 1: Pulling up the line is enabled.	R/W	R
b5	DRPD	D+/D- Line Resistor Control	Always set 0 at initialization.	R/W	R
b6	—	Reserved	When writing to this bit, write 0.	R/W	—
b7	HSE	High-Speed Operation Enable	Enables or disables high-speed operation. 0: High-speed operation is disabled (full-speed). 1: High-speed operation is enabled (the controller detects the communication speed)	R/W	R
b15 to b8	—	Reserved	When writing to these bits, write 0.	R/W	—

Note: Writing to this register is possible even while the UTMI clock (output of the clock signal from the PLL in the USB) is stopped. Note however that any values that are set while the UTMI clock is stopped are reflected after oscillation of the UTMI clock is restarted.

USB Block Operation Enable Bit (USBE)

The setting of the USBE bit enables or disables operation of the USB block.
When the USBE bit is modified from 1 to 0, this controller initializes the bits listed in Table 24.2.

Table 24.2 Registers Initialized by Writing USBE = 0

Register Name	Bit Name
SYSSTS0	LNST
DVSTCTR0	RHST
INTSTS0	DVSQ
USBADDR	USBADDR
USBREQ	bRequest bmRequestType
USBVAL	wValue
USBINDX	wIndex
USBLENG	wLength

The value of this bit must be changed when SUSPM = 1 and after the UTMI clock oscillation is started.

D+/D- Line Resistor Control (DRPD and DPRPU)

Settings related to USB data bus resistance are given in Table 24.3, USB Data Bus Resistance Control. Use the DRPD and DPRPU bits to select USB data bus resistance.

Table 24.3 USB Data Bus Resistance Control

Settings		USB Data Bus Resistance Control		
DRPD	DPRPU	D- Line	D+ Line	Remarks
0	0	Open	Open	
0	1	Open	Pull-up	Make these settings when the module is used as a function controller.
1	0	Pull-down	Pull-down	Setting prohibited
1	1	Pull-down	Pull-up	Setting prohibited

- D+ pull-up resistance control for function controller operation (DPRPU)

When this bit set to 1, this controller pulls up the D+ line to 3.3V and can notify the USB host that the function module is attached.
The controller cancels pulling-up of the D+ line by modifying this bit from 1 to 0, and the state for the USB host can be shown as detached.

Settings in these bits should be made set at the same time as that of PHYSET1.PHYVBUSIN.

Hi-Speed Operation Enable Bit (HSE)

Setting this bit to 1 enables high-speed operation. Setting the HSE bit to 1 lets this controller perform high-speed or full-speed operation based on the results of reset handshake.

Setting the HSE bit to 0 lets this controller perform full-speed operation.
On the other hand, setting the HSE bit to 1 lets this controller perform the reset handshake protocol and then perform high-speed or full-speed operation automatically according to the results.

This bit can be modified when DPRPU = 0.

24.2.1.2 System Configuration Control Register 1 (SYSCFG1)

Address(es): A006 0002h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	BWAIT[5:0]					
Value after hardware reset:	x	x	0	0	0	0	0	0	x	x	0	0	1	1	1	1
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b5 to b0	BWAIT[5:0]	CPU Bus Access Wait Specification	Specify the number of wait cycles for access to this controller. b5 b0 000000: 0 wait cycles (2 access cycles) : : 000010: 2 wait cycles (4 access cycles) : : 000100: 4 wait cycles (6 access cycles) : : 001111: 15 wait cycles (17 access cycles) (value after a reset) : : 111111: 63 wait cycles (65 access cycles)	R/W	R
b15 to b6	—	Reserved	When writing to these bits, write 0.	R/W	—

CPU Bus Access Wait Specification Bit (BWAIT)

For continuous access to the registers at addresses of this controller beginning at A006 0004h, at least 67 ns must be secured.

To satisfy this restriction, control wait cycles using the frequency of CPU clock. The value after a reset is the maximum value (17 clock cycles), so select an appropriate value no greater than this.

This setting is the same as that for waiting in access to a FIFO port register. The maximum speeds of access to the FIFO ports are as follows:
MBW = 10b (32-bit width): Max 60 Mbytes/sec
MBW = 01b (16-bit width): Max 30 Mbytes/sec
MBW = 00b (8-bit width): Max 15 Mbytes/sec

24.2.1.3 System Configuration Status Register (SYSSTS0)

Address(es): A006 0004h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LNST[1:0]	
Value after hardware reset:	x	x	x	x	x	x	x	x	x	0	0	x	x	x	x	x
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b1, b0	LNST[1:0]	USB Data Line Status Monitor	Indicate the state of the USB line.	R	W
b15 to b2	—	Reserved	These bits are read as an undefined value.	R	—

Line Status Monitor Bits (LNST)

The states of USB data bus line of this controller is shown in Table 24.4. The controller monitors the states of USB data bus lines (D+ and D- lines) in the LNST bits of the SYSSTS0 register.
Reference to the LNST bits must be made following attach processing (DPRPU = 1) after USBE has been set to 1.

Table 24.4 States of USB Data Bus Line

LNST[1]	LNST[0]	Full-Speed Operations	High-Speed Operations	Chirp Operations*1
0	0	SE0	Squelch*2	Squelch*2
0	1	J state	Unsquelch*3	Chirp J*4
1	0	K state	Invalid	Chirp K*5
1	1	SE1	Invalid	Invalid

Note 1. Chirp: Reset handshake protocol being executed in high-speed operations enabled state (HSE = “1”)
Note 2. Squelch: SE0 or idle state
Note 3. Unsquelch: High-speed J state or high-speed K state
Note 4. Chirp J: Chirp J-state
Note 5. Chirp K: Chirp K-state

24.2.2 USB Signal Control

24.2.2.1 Device State Control Register 0 (DVSTCTR0)

Address(es): A006 0008h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	WKUP	—	—	—	—	—	RHST[2:0]		
Value after hardware reset:	x	x	x	x	x	0	0	0	0	0	0	0	x	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	0	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b2 to b0	RHST[2:0]	Reset Handshake	Indicate the state of reset handshake.	R	W
b7 to b3	—	Reserved	When writing to these bits, write 0.	R/W	—
b8	WKUP	Wakeup Output	Enables or disables the remote wakeup (resume signal output). Only 1 can be written to this bit. 0: Remote wakeup signal is not output. 1: Remote wakeup signal is output.	R/W	R/W
b15 to b9	—	Reserved	When writing to these bits, write 0.	R/W	—

Reset Handshake Status Bits (RHST)

These bits indicate the result of reset handshake. Table 24.5 lists the results of reset handshake.

Table 24.5 State of Reset Handshake

Bus State	RHST Bit Value
Powered or no connection	000b
Reset handshake in progress	100b
Full-speed connection	010b
High-speed connection	011b

When the HSE bit is set to 1, the RHST bits indicate 100b when this controller detects a USB bus reset. Then, these bits indicate 011b when this controller outputs Chirp-K and detects Chirp-JK from the USB host three times. If the connection speed is not fixed to high speed within 2.5 ms after Chirp-K output, these bits indicate 010b.

When the HSE bit is set to 0, these bits indicate 010b when the controller detects a USB bus reset.

A DVST interrupt is generated when the value of the RHST bits is 010b or 011b after the USB bus reset is detected by the controller.

Remote Wakeup (Resume Signal Output) Enable/Disable Bit (WKUP)

With the WKUP bit set to 1, this controller outputs the remote wakeup signal to the USB bus.
This controller controls the output time of a remote wakeup signal. When this bit is set to 1 by software, the controller clears this bit to 0 after outputting the 10-ms K-state.
According to the USB Specifications, the USB bus idle state must be kept for 5 ms or longer before a remote wakeup signal is sent. If this controller writes 1 to this bit right after detection of the suspended state, the K-state will be output after 2 ms.

Do not write 1 to this bit unless the device is in the suspended state (bits DVSQ = 1xxb) and the USB host enables the remote wakeup signal.
When this bit is set to 1, the internal clock must not be stopped even in the suspended state (write 1 to this bit while the SUSPM bit = 1).

24.2.3 Test Mode

24.2.3.1 USB Test Mode Register (TESTMODE)

Address(es): A006 000Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	UTST[3:0]			
Value after hardware reset:	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b3 to b0	UTST[3:0]	Test Mode	* See the detailed description below.	R/W	R
b15 to b4	—	Reserved	When writing to these bits, write 0.	R/W	—

Test Mode Bits (UTST)

Writing values to these bits allows this controller to output a USB test signal during high-speed operations. Table 24.6 shows the test mode operations of the controller.

Table 24.6 Test Mode Operations

Test Mode	UTST Settings
Normal operation	0000b
Test_J	0001b
Test_K	0010b
Test_SE0_NAK	0011b
Test_Packet	0100b
Test_Force_Enable	—
Reserved	0101b to 0111b

Write to these bits according to the SetFeature request from the USB host during high-speed communication.
This controller does not enter the suspended state while these bits are 0001h to 0100h.
To perform normal USB communications after the test mode is set, apply a hardware reset.

24.2.4 DMA-FIFO Bus Access Control

The D0FBCFG and D1FBCFG registers control bus access to DMA0-FIFO and DMA1-FIFO, respectively.

24.2.4.1 DMA0-FIFO Bus Configuration Register (D0FBCFG)
DMA1-FIFO Bus Configuration Register (D1FBCFG)

Address(es): D0FBCFG: A006 0010h
D1FBCFG: A006 0012h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DFACC[1:0]		—	—	—	—	—	—	—	TENDE	—	—	—	—
Value after hardware reset:	x	x	0	0	x	x	x	x	x	x	x	0	x	x	x	x
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b3 to b0	—	Reserved	When writing to these bits, write 0.	R/W	—
b4	TENDE	TENDx_N Signal Enable	Enables or disables the TENDx_N signal for input. 0: TENDx_N signal is disabled. 1: TENDx_N signal is enabled. Set this bit to 0 when DFACC = 01b or 10b.	R/W	R
b11 to b5	—	Reserved	When writing to these bits, write 0.	R/W	—
b13, b12	DFACC[1:0]	DMAx-FIFO Access Mode	Specify the access mode of the selected FIFO port. b13 b12 00: Cycle stealing mode (value after a reset) 01: 16-byte continuous access mode 10: 32-byte continuous access mode 11: Invalid	R/W	R
b15, b14	—	Reserved	When writing to these bits, write 0.	R/W	—

TENDx_N Input Signal Enable Bit (TENDE)

This bit selects enabling or disabling of the TEND input for write access to a FIFO buffer in DMA mode.
Set this bit to 0 when DFACC = 01b or 10b.

DMA Transfer FIFO Access Mode Select Bits (DFACC)

These bits specify the transfer mode for DMA transfer.

(a) When cycle stealing mode is set, use the DxFIFO port for access to the FIFO buffer.

(b) When 16- or 32-byte continuous access mode is set, use the DxFIFO continuous transfer port for access to the FIFO buffer. The MBW bits of DxFIFOSEL can only be set to 10b (32-bit width).

24.2.5 FIFO Ports

24.2.5.1 CFIFO Port Register (CFIFO)
D0FIFO Port Register (D0FIFO)
D1FIFO Port Register (D1FIFO)

Address(es): CFIFO: A006 0014h
D0FIFO: A006 0018h
D1FIFO: A006 001Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	FIFOPORT[31:0]															
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FIFOPORT[31:0]															
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Symbol	Bit Name	Description	S/W	H/W
b31 to b0	FIFOPORT [31:0]	FIFO Port	Accessing these bits allows reading the received data from the FIFO buffer or writing the transmit data to the FIFO buffer.	R/W	R/W

- FIFO port control

The transmission/reception buffer of this controller is made up of a FIFO structure (FIFO buffer). Use the FIFO port register for access to the FIFO buffer. There are three FIFO ports: CFIFO, D0FIFO, and D1FIFO ports. Also, there is a DxFIFO continuous transfer port for continuous transfer. Each FIFO port consists of a port register (CFIFO, D0FIFO, or D1FIFO) that handles reading of data from the FIFO buffer memory and writing of data to the FIFO buffer memory, a port select register (CFIFOSEL, D0FIFOSEL, or D1FIFOSEL) that is used to select the pipe assigned to the FIFO port, and a port control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR).

The FIFO ports have the following restrictions.

(1) Access to an FIFO buffer for DCP should be through the CFIFO port.
(2) Access to an FIFO buffer by DMA transfer when DFACC = 00b (cycle stealing mode) should be through the DxFIFO port.
(3) Access to an FIFO buffer by DMA transfer when DFACC = 01b or 10b (16-byte or 32-byte continuous access mode) should be through the DxFIFO continuous transfer port.
(4) The DxFIFO ports can also be accessed by the CPU.
(5) When using functions specific to the FIFO port, the pipe number (selected pipe) specified by the CURPIPE bits cannot be changed (when the DMA transfer function is used, etc.).
(6) Registers containing the FIFO port do not affect the other FIFO ports.
(7) Do not assign the same pipe to separate FIFO ports.
(8) In the FIFO buffer state, there are two types of access rights: one assigned to the CPU, and the other to SIE. Access from the CPU is not possible when SIE has the rights to access the FIFO buffer.

FIFO Port Bits (CFIFO, D0FIFO, and D1FIFO)

This controller accesses the FIFO buffer assigned to the pipe number written to the CURPIPE bits of the corresponding select registers (CFIFOSEL, D0FIFOSEL, and D1FIFOSEL) through access to any of these registers by software.

This register can only be accessed when the FRDY bit of each control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR) indicates 1 (or while this controller asserts UCL_Dx_DREQ (interrupt sources [43] and [44]) output signals).
The valid bits of this register depend on the settings of the corresponding MBW and BIGEND bits.
Table 24.7 to Table 24.9 list the valid bits.

Table 24.7 Endian Operation in 32-Bit Access (MBW = 10)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	N+3 address	N+2 address	N+1 address	N+0 address
1	N+0 address	N+1 address	N+2 address	N+3 address

Table 24.8 Endian Operation in 16-Bit Access (MBW = 01)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	N+1 address	N+0 address	Writing: Invalid Reading: Prohibited*1	
1	Writing: Invalid Reading: Prohibited*1		N+0 address	N+1 address

Table 24.9 Endian Operation in 8-Bit Access (MBW = 00)

BIGEND	b31 to b24	b23 to b16	b15 to b8	b7 to b0
0	N+0 address	Writing: Invalid Reading: Prohibited*1		
1	Writing: Invalid Reading: Invalid*1			N+0 address

Note 1. Reading data from the invalid bits in a word or byte unit is prohibited.

24.2.5.2 CFIFO Port Select Register (CFIFOSEL)

Address(es): A006 0020h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RCNT	REW	—	—	MBW[1:0]		—	BIGEND	—	—	ISEL	—	CURPIPE[3:0]			
Value after hardware reset:	0	0	x	x	0	0	x	0	x	x	0	x	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b3 to b0	CURPIPE [3:0]	FIFO Port Access Pipe Specification	Specify the pipe number to access the CFIFO port. b3 b0 0000: DCP 0001: Pipe 1 0010: Pipe 2 1000: Pipe 8 1001: Pipe 9	R/W	R
b4	—	Reserved	When writing to this bit, write 0.	R/W	—
b5	ISEL	FIFO Port Access Direction when DCP is Selected	Specifies access direction of the FIFO port when DCP is selected in CURPIPE bits. 0: Reading from the buffer memory 1: Writing to the buffer memory	R/W	R
b7, b6	—	Reserved	When writing to these bits, write 0.	R/W	—
b8	BIGEND	FIFO Port Endian Control	Specifies the CFIFO port byte endian. 0: Little endian 1: Big endian	R/W	R
b9	—	Reserved	When writing to this bit, write 0.	R/W	—
b11, b10	MBW[1:0]	CFIFO Port Access Bit Width	Specify the bit width for accessing the CFIFO port. b11b10 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited	R/W	R
b13, b12	—	Reserved	When writing to these bits, write 0.	R/W	—
b14	REW	Buffer Pointer Rewind	Specifies 1 when rewinding the buffer pointer. This bit is read as 0. 0: The buffer pointer is not rewound. 1: The buffer pointer is rewound.	R/W	R
b15	RCNT	Read Count Mode	Specifies the read mode for the value in the DTLN bits in CFIFOCTR. 0: The DTLN bits are cleared when all of the received data has been read from the CFIFO. 1: The DTLN bits are decremented each time the received data is read from the CFIFO.	R/W	R

FIFO Port Access Pipe Specification Bits (CURPIPE)

Write the pipe number for the data to be read or written through the CFIFO port.

When modifying this bit field, write this bit first and then read this bit. Only when the written value matches the read one, proceed to the next process.

Do not specify the same pipe in the CURPIPE bits of the CFIFOSEL, D0FIFOSEL, and D1FIFOSEL registers.

Even if the setting of these bits is modified during access to the FIFO buffer, the state of the FIFO buffer is retained, with continued access proceeding after these bits are re-set to the value before the modification.

FIFO Port Access Direction Bit when DCP is Selected (ISEL)

To change this bit when the specified pipe is DCP, first write the data to this bit and then read it. Proceed to the next process after checking if the written value matches with the read value.

When the settings of this bit are modified during access to the FIFO buffer, access up to then is saved. Access to the buffer can be continued after rewriting the settings.

Settings in this bit should be made at the same as that of the CURPIPE bits.

FIFO Port Endian Control Bit (BIGEND)

In this bit, set the byte endian for the CFIFO port.
For details, see the description of FIFO Port Bits (CFIFO, D0FIFO, and D1FIFO).

CFIFO Port Access Bit Width Bits (MBW)

In this bit field, set the bit width for accessing the CFIFO port.

When the pipe specified in the CURPIPE bits is in the receiving direction, if reading is started after setting this bit field, do not modify the value of the MBW bits until all data is read.
When the specified pipe is in the receiving direction, modify the value of the CURPIPE bit once and then set the CURPIPE and MBW bits at the same time.
For the procedure for changing the CURPIPE bit, see the description for the same.

When the specified pipe is in the transmitting direction, to switch the bit width from 8 bits to 16 or 32 bits, or from 16 bits to 32 bits is not allowed while writing to the buffer memory is in progress.
With the 16-bit or 32-bit width setting, writing to odd bytes is possible by exercising byte access control.

Buffer Pointer Rewind (REW)

When the selected pipe is in the receiving direction, setting the REW bit to 1 while the FIFO buffer is being read allows re-reading the FIFO buffer from the first data (in double-buffer operation, re-reading the currently-read FIFO buffer plane from the first data is allowed).

Do not set REW to 1 simultaneously with modifying the CURPIPE bits. Before setting REW to 1, be sure to check that FRDY is 1.

To re-write to the FIFO buffer again from the first data for the pipe in the transmitting direction, use the BCLR bit.

Read Count Mode (RCNT)

When 0 is written to this bit, if all received data of the FIFO buffer assigned to the pipe specified in the CURPIPE bits (selected pipe) is read (when the data is read on one side of a double buffer), this controller clears the DTLN bits in the CFIFOCTR register to 0.

When 1 is written to this bit, the controller decrements the value of the DTLN bits in the CFIFOCTR register whenever the received data is read from the FIFO buffer assigned to the specified pipe.

24.2.5.3 D0FIFO Port Select Register (D0FIFOSEL)
D1FIFO Port Select Register (D1FIFOSEL)

Address(es): D0FIFOSEL: A006 0028h
D1FIFOSEL: A006 002Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RCNT	REW	DCLRM	DREQE	MBW[1:0]		—	BIGEND	—	—	—	—	CURPIPE[3:0]			
Value after hardware reset:	0	0	0	0	0	0	x	0	x	x	x	x	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b3 to b0	CURPIPE [3:0]	FIFO Port Access Pipe Specification	b3 b0 0000: Not specified 0001: Pipe 1 0010: Pipe 2 1000: Pipe 8 1001: Pipe 9	R/W	R
b7 to b4	—	Reserved	When writing to these bits, write 0.	R/W	—
b8	BIGEND	FIFO Port Endian Control	Specifies the CFIFO port byte endian. 0: Little endian 1: Big endian	R/W	R
b9	—	Reserved	When writing to this bit, write 0.	R/W	—
b11, b10	MBW[1:0]	FIFO Port Access Bit Width	Specify the bit width for accessing the FIFO port. b11b10 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited	R/W	R
b12	DREQE	UCL_Dx_DREQ Signal Output Enable	Enables or disables the output of the UCL_Dx_DREQ (interrupt sources [43] and [44]) signals. 0: Output is disabled 1: Output is enabled	R/W	R
b13	DCLRM	Auto Buffer Memory Clear Mode after Specified Pipe Data is Read	Enables or disables automatic clearing of the buffer memory after data has been read out using the selected pipe. 0: Auto buffer clear mode is disabled 1: Auto buffer clear mode is enabled	R/W	R
b14	REW	Buffer Pointer Rewind	Specifies 1 when rewinding the buffer pointer. This bit is read as 0. 0: The buffer pointer is not rewound 1: The buffer pointer is rewound	R/W	R
b15	RCNT	Read Count Mode	Specifies the read mode for the value in the DTLN bits in Dx_FIFOCTR. 0: The DTLN bits are cleared when all of the received data has been read. 1: The DTLN bits are decremented each time the received data is read.	R/W	R

FIFO Port Access Pipe Specification Bits (CURPIPE)

Write the pipe number for the data to be read or written through the DxFIFO port.

When modifying this bit field, first write a value to this bit field and then read it. Check if the write value matches the read value and then proceed to the next process.

Do not specify the same pipe in the CURPIPE bits of the CFIFOSEL, D0FIFOSEL, and D1FIFOSEL registers.

Even if the setting of these bits is modified during access to the FIFO buffer, the state of the FIFO buffer is retained, with continued access proceeding after these bits are re-set to the value before the modification.

DxFIFO Port Byte Endian Control Bit (BIGEND)

Write the byte endian for the DxFIFO port in this bit.
For details, see the description of FIFO Port Bits (CFIFO, D0FIFO, and D1FIFO).

DxFIFO Port Access Bit Width (MBW)

Write the bit width for accessing the DxFIFO port in this bit.

When the pipe specified in the CURPIPE bits is in the receiving direction, if reading is started after setting this bit field, do not modify the value of the MBW bits until all data is read.
When the specified pipe is in the receiving direction, set the CURPIPE and MBW bits at the same time.
For the procedure for changing the CURPIPE bit, see the description for the same.

For details, see the description of FIFO Port Access Pipe Specification Bits (CURPIPE).

UCL_Dx_DREQ Output Disable/Enable Bit (DREQE)

Use this bit to enable or disable the output of the UCL_Dx_DREQ signals.

When enabling the output of the UCL_Dx_DREQ signals, set this bit to 1 after setting the CURPIPE bits.

When modifying the setting of the CURPIPE bits, do so after setting this bit to 0.

Auto FIFO Buffer Clear Disable/Enable Bit (DCLRM)

After reading data in the specified pipe, disable or enable automatic clearing of the FIFO buffer. When 1 is written to this bit, this controller sets the BCLR bit to 1 for a FIFO buffer if a zero-length packet is received when the FIFO buffer assigned to the specified pipe is empty, or when a short packet is received and the data is completely read while BFRE = 1 .

When using this controller with the BRDYM bit set to 1, be sure to set this bit to 0.

Buffer Pointer Rewind (REW)

When the selected pipe is in the receiving direction, setting the REW bit to 1 while the FIFO buffer is being read allows re-reading the FIFO buffer from the first data (in double-buffer operation, re-reading the currently-read FIFO buffer plane from the first data is allowed).

Do not set the REW bit to 1 simultaneously with modifying the CURPIPE bits. Before setting the REW bit to 1, be sure to check that the FRDY bit is 1.
To re-write to the FIFO buffer again from the first data for the pipe in the transmitting direction, use the BCLR bit.

Read Count Mode (RCNT)

When 0 is written to this bit, if all received data of the FIFO buffer assigned to the pipe specified in the CURPIPE bits (selected pipe) is read (when the data is read on one side of a double buffer), this controller clears the DTLN bits in the DxFIFOCTR register to 0.

When 1 is written to this bit, the controller decrements the value of the DTLN bits in the DxFIFOCTR register whenever the received data is read from the FIFO buffer assigned to the specified pipe.

24.2.5.4 CFIFO Port Control Register (CFIFOCTR)
D0FIFO Port Control Register (D0FIFOCTR)
D1FIFO Port Control Register (D1FIFOCTR)

Address(es): CFIFOCTR: A006 0022h
D0FIFOCTR: A006 002Ah
D1FIFOCTR: A006 002Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BVAL	BCLR	FRDY	—	DTLN[11:0]											
Value after hardware reset:	0	0	0	x	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b11 to b0	DTLN[11:0]	Receive Data Length	Indicate the length of the receive data.	R	W
b12	—	Reserved	When writing to this bit, write 0.	R/W	—
b13	FRDY	FIFO Port Ready	Indicates whether the FIFO port can be accessed. 0: FIFO port is not accessible. 1: FIFO port is accessible.	R	W
b14	BCLR	CPU Buffer Clear	Specifies 1 to clear the FIFO buffer on the CPU side for the selected pipe. Only 1 can be written to this bit. This bit is read as 0. 0: Invalid 1: Clears the CPU buffer memory on the CPU side.	R/W	R
b15	BVAL	Buffer Memory Valid Flag	Specifies 1 when writing has ended in the CPU-side FIFO buffer for the pipe specified in CURPIPE (selected pipe). Only 1 can be written to this bit. 0: Invalid 1: Writing ended	R/W	R/W

Receive Data Length Bits (DTLN)

The DTLN bits indicate the length of the receive data. While the FIFO buffer is being read, the DTLN bits indicate different values depending on the RCNT bit value as described below.

(1) When RCNT = 0:
The length of received data is set in these bits, and the value is retained until all received data has been read from a single FIFO buffer plane.
While BFRE is 1, these bits retain the length of the receive data until BCLR is set to 1 even after all data has been read.

(2) When RCNT = 1:
This controller decrements the value indicated by these bits each time data is read from the FIFO buffer.
(The value is decremented by 1 when MBW is 00, by 2 when MBW is 01b, and by 4 when MBW is 10b.)

This controller sets these bits to 0 when all data has been read from one FIFO buffer plane. However, in double-buffer operation, if data has been received in one FIFO buffer plane before all data has been read from the other plane, this controller sets these bits to indicate the length of the receive data in the latter plane when all data has been read from the former plane.

When reading these bits during FIFO buffer reading while RCNT = 1, note that these bits are updated within 150 ns after a read cycle for the FIFO port.

FIFO Port Ready Bit (FRDY)

This bit indicates whether the FIFO port can be accessed from the CPU (DMAC).

In the following cases, the controller sets FRDY to 1, but data cannot be read via the FIFO port because there is no data to be read. In these cases, set BCLR to 1, clear the FIFO buffer, and then enable transmission and reception of the next data.

(1) A zero-length packet is received when the FIFO buffer assigned to the selected pipe is empty.
(2) A short packet is received and the data is completely read while BFRE is 1.

CPU Buffer Clear Bit (BCLR)

When this bit is set to 1, this controller clears the FIFO buffer on the CPU for the selected pipe.

When double-buffer operation is set for the FIFO buffer assigned to the selected pipe, this controller clears only one plane of the FIFO buffer even when both planes are read-enabled.

When the selected pipe is the DCP, setting BCLR to 1 allows this controller to clear the FIFO buffer regardless of whether the FIFO buffer is on the CPU side or SIE side. When clearing the buffer on the SIE side, set the PID bits for the DCP to NAK before setting BCLR to 1.

When the selected pipe is not the DCP, writing 1 to this bit should be done while FRDY indicates 1. When checking the FRDY bit after executing BCLR, allow an interval of at least 80 ns after executing BCLR before referencing FRDY.

Buffer Memory Valid Flag (BVAL)

When the pipe specified in the CURPIPE bits (selected pipe) is for transmission, write 1 to this bit in the following cases. This controller sets the FIFO buffer from the CPU side to the SIE side, enabling transmission.

(1) To transmit the short packet, set this bit to 1 after data has been written.
(2) To transmit a zero-length packet, set this bit to 1 before writing data to FIFO.
(3) Set this bit to 1 after the number of data bytes has been written for the pipe in continuous transfer mode, where the number is a natural integral multiple of the maximum packet size and less than the buffer size.

When the data of the maximum packet size has been written for the pipe in non-continuous transfer mode, this controller sets this bit to 1 and switches the FIFO buffer from the CPU side to the SIE side, enabling transmission.

When 1 is written to the BVAL and BCLR bits at the same time when the specified pipe is in the transmitting direction, the controller clears the old data and enables the transmission of a zero-length packet.

Writing 1 to this bit should be done while FRDY indicates 1. When checking the FRDY bit after executing BVAL, allow an interval of at least 80 ns after executing BVAL before referencing FRDY.

When the specified pipe is in the receiving direction, do not write 1 to this bit.

24.2.6 Enabling Interrupts

24.2.6.1 Interrupt Enable Register 0 (INTENB0)

Address(es): A006 0030h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	—	—	—	—	—	—	—	—
Value after hardware reset:	0	0	0	0	0	0	0	0	x	x	x	x	x	x	x	x
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b7 to b0	—	Reserved	When writing to these bits, write 0.	R/W	—
b8	BRDYE	Buffer Ready Interrupt Enable	Enables or disables the USB interrupt output when the BRDY interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b9	NRDYE	Buffer Not Ready Response Interrupt Enable	Enables or disables the USB interrupt output when the NRDY interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b10	BEMPE	Buffer Empty Interrupt Enable	Enables or disables the USB interrupt output when the BEMP interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b11	CTRE	Control Transfer Stage Transition Interrupt Enable	Enables or disables the USB interrupt output when the CTRT interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b12	DVSE	Device State Transition Interrupt Enable	Enables or disables the USB interrupt output when the DVST interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b13	SOFE	Frame Number Update Interrupt Enable	Enables or disables the USB interrupt output when the SOF interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b14	RSME	Resume Interrupt Enable	Enables or disables the USB interrupt output when the RESM interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b15	VBSE	VBUS Interrupt Enable	Enables or disables the USB interrupt output when the VBINT interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R

24.2.6.2 BRDY Interrupt Enable Register (BRDYENB)

Address(es): A006 0036h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PIPEBRDYE[9:0]									
Value after hardware reset:	x	x	x	x	x	x	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b9 to b0	PIPEBRDYE [9:0]	BRDY Interrupt Enable for Each Pipe	Enable or disable the BRDY interrupt for each pipe. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b15 to b10	—	Reserved	When writing to these bits, write 0.	R/W	—

Note: The bit number corresponds to the pipe number.

BRDY Interrupt Enable Bit for Each Pipe (PIPEBRDYE)

On detecting the BRDY interrupt for the pipe corresponding to the bit in this register which has been set to 1 by software, this controller sets 1 to the corresponding PIPEBRDY bit in BRDYSTS and the BRDY bit in INTSTS0, and asserts the interrupt signal through the INT_N (interrupt source [42]) pin.

While at least one PIPEBRDY bit in BRDYSTS indicates 1, this controller asserts the interrupt signal through the INT_N pin when the corresponding interrupt enable bit of this register is changed from 0 to 1 by software.

24.2.6.3 NRDY Interrupt Enable Register (NRDYENB)

Address(es): A006 0038h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PIPENRDYE[9:0]									
Value after hardware reset:	x	x	x	x	x	x	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b9 to b0	PIPENRDYE [9:0]	NRDY Interrupt Enable for Each Pipe	Enable or disable the NRDY interrupt for each pipe. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b15 to b10	—	Reserved	When writing to these bits, write 0.	R/W	—

Note: The bit number corresponds to the pipe number.

NRDY Interrupt Enable Bit for Each Pipe (PIPENRDYE)

On detecting the NRDY interrupt for the pipe corresponding to the bit in this register which has been set to 1 by software, this controller sets 1 to the corresponding PIPENRDY bit in NRDYSTS and the NRDY bit in INTSTS0, and asserts the interrupt signal through the INT_N pin.

While at least one PIPENRDY bit in NRDYSTS indicates 1, this controller asserts the interrupt signal through the INT_N pin when the corresponding interrupt enable bit of this register is changed from 0 to 1 by software.

24.2.6.4 BEMP Interrupt Enable Register (BEMPENB)

Address(es): A006 003Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PIPEBEMPE[9:0]									
Value after hardware reset:	x	x	x	x	x	x	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b9 to b0	PIPEBEMPE [9:0]	BEMP Interrupt Enable for Each Pipe	Enable or disable the BEMP interrupt for each pipe. 0: Interrupt output disabled 1: Interrupt output enabled	R/W	R
b15 to b10	—	Reserved	When writing to these bits, write 0.	R/W	—

Note: The bit number corresponds to the pipe number.

BEMP Interrupt Enable Bit for Each Pipe (PIPEBEMPE)

On detecting the BEMP interrupt for the pipe corresponding to the bit in this register which has been set to 1 by software, this controller sets 1 to the corresponding PIPEBEMP bit in BEMPSTS and the BEMP bit in INTSTS0, and asserts the interrupt signal through the INT_N pin.

While at least one PIPEBEMP bit in BEMPSTS indicates 1, this controller asserts the interrupt signal through the INT_N pin when the corresponding interrupt enable bit of this register is changed from 0 to 1 by software.

24.2.7 SOF Control Register

24.2.7.1 SOF Pin Configuration Register (SOFCFG)

Address(es): A006 003Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	BRDYM	INTL	EDGESTS	—	—	—	—
Value after hardware reset:	x	x	x	x	x	x	x	0	x	0	0	0	0	0	x	x
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b3 to b0	—	Reserved	When writing to these bits, write 0.	R/W	—
b4	EDGESTS	Interrupt Edge Processing Status	Indicates the state of processing in response to interrupts when edge-sensing has been selected. 0: Processing in response to an edge-sensed interrupt is not in progress. 1: Processing in response to an edge-sensed interrupt is in progress	R	W
b5	INTL	Interrupt Output Sense Select	Selects sense mode for USB interrupt output. 0: Edge sense 1: Level sense	R/W	R
b6	BRDYM	PIPEBRDY Interrupt Status Clear Timing	Specifies the timing for clearing the PIPEBRDY interrupt state. 0: Software clears the state. 1: Hardware clears the state by reading from the FIFO buffer or by writing to the FIFO buffer This bit can be set only in the initial setting (before communications). The setting cannot be changed once communication starts.	R/W	R
b15 to b7	—	Reserved	When writing to these bits, write 0.	R/W	—

Note 1. When setting the BRDYM bit to 1, set the INTL bit to 1 (level sense).
Note 2. With the INTL bit set to 0, to stop the system clock (SUSPM = 0) after clearing the interrupt state, write 0 to the SUSPM bit after checking that the EDGESTS bit is set to 0.

24.2.8 Interrupt Status

24.2.8.1 Interrupt Status Register 0 (INTSTS0)

Address(es): A006 0040h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ[2:0]			VALID	CTSQ[2:0]		
Value after hardware reset:	0	0	0	0	0	0	0	0	x	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	1	x	x	x	x	x	0	0	1	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b2 to b0	CTSQ[2:0]	Control Transfer Stage	Indicate the control transfer stage. b2 b0 000: Idle or setup stage 001: Control read data stage 010: Control read status stage 011: Control write data stage 100: Control write status stage 101: Control write (no data) status stage 110: Control transfer sequence error 111: Reserved	R	W
b3	VALID	USB Request Reception	Indicates whether reception of the USB request has been detected. When writing to this bit, only 0 can be written. 0: Not detected 1: Setup packet reception	R/W	W
b6 to b4	DVSQ[2:0]	Device State	Indicate the device state. b6 b4 000: Powered state 001: Default state 010: Address state 011: Configured state 1xx: Suspended state	R	W
b7	VBSTS	VBUS Input Status	Indicates the input state of the VBUS pin. 0: VBUS pin is at the low level 1: VBUS pin is at the high level	R	W
b8	BRDY	BRDY Interrupt Status	Indicates the BRDY interrupt state. 0: BRDY interrupts not generated 1: BRDY interrupts generated	R	W
b9	NRDY	NRDY Interrupt Status	Indicates the NRDY interrupt state. 0: NRDY interrupts not generated 1: NRDY interrupts generated	R	W
b10	BEMP	BEMP Interrupt Status	Indicates the BEMP interrupt state. 0: BEMP interrupts not generated 1: BEMP interrupts generated	R	W
b11	CTRT	Control Transfer Stage Transition Interrupt Status	Indicates the control transfer stage transition interrupt state. When writing to this bit, only 0 can be written. 0: Control transfer stage transition interrupts not generated 1: Control transfer stage transition interrupts generated	R/W	W
b12	DVST	Device State Transition Interrupt Status	Indicates the device state transition interrupt state. When writing to this bit, only 0 can be written. 0: Device state transition interrupts not generated 1: Device state transition interrupts generated	R/W	W

Bit	Symbol	Bit Name	Description	S/W	H/W
b13	SOFR	Frame Number Update Interrupt Status	Indicates the frame number update interrupt state. When writing to this bit, only 0 can be written. 0: SOF interrupts not generated 1: SOF interrupts generated	R/W	W
b14	RESM	Resume Interrupt Status	Indicates the resume detection interrupt state. When writing to this bit, only 0 can be written. 0: Resume interrupts not generated 1: Resume interrupts generated	R/W	W
b15	VBINT	VBUS Change Detect Interrupt Status	Indicates the VBUS change detection interrupt state. When writing to this bit, only 0 can be written. 0: VBUS interrupts not generated 1: VBUS interrupts generated	R/W	W

Note 1. To clear the state indicated by the VBINT, RESM, SOFR, DVST, or CTRT bits, write 0 only to the bit to be cleared, and write 1 to the other bits. Do not write 0 to the status bit set to 0.
Note 2. The controller detects the change in state indicated by the VBINT and RESM bits of this register, even while the clock is being stopped (SUSPM = 0), and conveys the interrupt if the corresponding interrupt is enabled. When the clock is enabled, clear the state using software.

Buffer Ready Interrupt Status Bit (BRDY)

This controller sets a BRDY bit to 1 when, among the PIPEBRDY bits for which the corresponding PIPEBRDYE bits in BRDYENB have been set to 1, at least one PIPEBRDY bit in BRDYSTS is set to 1 (that is, when this controller detects the BRDY interrupt state in at least one pipe among the pipes for which software has enabled the output of BRDY interrupts).

For the conditions for the setting of PIPEBRDY bits, refer to the description of the BRDYSTS register.

When a BRDY bit has been set to 1, the controller clears the bit in response to software writing 0 to all PIPEBRDY bits corresponding to PIPEBRDYE bits which have been set to 1.
A BRDY bit will not be cleared to 0 in response to software only writing 0 to individual bits that have been set to 1.

Buffer Not Ready Interrupt Status Bit (NRDY)

This controller sets an NRDY bit to 1 when, among the PIPEBRDY bits for which the corresponding PIPENRDYE bits in NRDYENB have been set to 1, at least one PIPENRDY bit in BNRDYSTS is set to 1 (that is, when this controller detects the NRDY interrupt state in at least one pipe among the pipes for which software has enabled the output of NRDY interrupts).

For the conditions for the setting of PIPENRDY bits, refer to the description of the NRDYSTS register.

When an NRDY bit has been set to 1, the controller clears the bit in response to software writing 0 to all PIPENRDY bits corresponding to PIPENRDYE bits which have been set to 1.
An NRDY bit will not be cleared to 0 in response to software only writing 0 to individual bits that have been set to 1.

Buffer Empty Interrupt Status Bit (BEMP)

This controller sets a BEMP bit to 1 when, among the PIPEBRDY bits for which the corresponding PIPEBEMPE bits in BEMPENB have been set to 1, at least one PIPEBEMP bit in BEMPSTS is set to 1 (that is, when this controller detects the BEMP interrupt state in at least one pipe among the pipes for which software has enabled the output of BEMP interrupts).

For the conditions for the setting of PIPEBEMP bits, refer to the description of the BEMPSTS register.

When a BEMP bit has been set to 1, the controller clears the bit in response to software writing 0 to all PIPEBEMP bits corresponding to PIPEBEMPE bits which have been set to 1.
A BEMP bit will not be cleared to 0 in response to software only writing 0 to individual bits that have been set to 1.

Control Transfer Stage Transition Interrupt Status Bit (CTRT)

If this controller detects the stage transition of control transfer, it updates the CTSQ value and sets 1 to this bit.

When this interrupt occurs, clear the state before the controller detects the next control transfer stage transition.

Device State Transition Interrupt Status Bit (DVST)

If this controller detects a change in the device state, it updates the DVSQ value and sets 1 to this bit.

When this interrupt occurs, clear the state before the controller detects the next device state transition.

Frame Number Update Interrupt Status Bit (SOFR)

The conditions when the controller sets 1 in this bit are below.
When updating the frame number, this controller sets 1 to this bit (this interrupt is detected every 1 ms).
The controller detects the SOFR interrupt by internal interpolation even if the SOF packet from the USB host is corrupted.

Resume Interrupt Status Bit (RESM)

If this controller is in the suspended state (DVSQ = 1xxb) and the DP pin falling edge is detected, 1 is set to this bit.

VBUS Change Interrupt Status Bit (VBINT)

When this controller detects a change in the VBUS pin input value (from high to low and from low to high), 1 is written to this bit. The controller writes the input value of the VBUS pin to the VBSTS bit. When the VBINT interrupt occurs, use the software to execute a consistency check several times during reading the VBSTS bit, and remove the chattering effect.

24.2.8.2 BRDY Interrupt Status Register (BRDYSTS)

Address(es): A006 0046h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PIPEBRDY[9:0]									
Value after hardware reset:	x	x	x	x	x	x	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b9 to b0	PIPEBRDY [9:0]	BRDY Interrupt Status for Each Pipe	Indicate the BRDY interrupt state for each pipe. When writing to these bits, only 0 can be written. 0: Interrupts not generated 1: Interrupts generated	R/W	W
b15 to b10	—	Reserved	When writing to these bits, write 0.	R/W	—

Note 1. The bit number corresponds to the pipe number.
Note 2. When BRDYM = 0, to clear the state of each bit of this register, write 0 only to the bit to be cleared and 1 to other bits.
Note 3. When BRDYM = 0, clearing these interrupt status bits should be done before accessing the FIFO.

BRDY Interrupt Status Bit for Each Pipe (PIPEBRDY)

When the BRDY interrupt is detected for a pipe by this controller, the controller sets 1 in the corresponding PIPEBRDY bit of the BRDYSTS register. Here, when 1 is written to the corresponding bit of BRDYENB register by using the software, the controller sets 1 to the BRDY bit of the INTSTS0 register.

The conditions for generating and clearing the BRDY interrupt depend on the settings of the SOFCFG.BRDYM bit and PIPECFG.BFRE bit for each pipe as described below.

- When BRDYM = 0 and BFRE = 0

With these settings, the BRDY interrupt indicates that the FIFO port is accessible.
On any of the following conditions, this controller generates the internal BRDY interrupt request trigger and sets 1 to the PIPEBRDY bit corresponding to the selected pipe.

(1) For the pipe in the transmitting direction
  (a) When the software has modified the DIR bit from 0 to 1
  (b) When the controller ends the packet transmission of the selected pipe in the condition where writing is not possible from the CPU to the FIFO buffer that has been assigned to the selected pipe (when the BSTS bit is read as 0).
  In continuous transmission/reception mode, the request trigger is generated when data of one plane of the FIFO buffer is complete.
  (c) In double-buffer operation, one FIFO buffer being empty on completion of the writing of data to the other FIFO buffer.
  The request trigger is not generated until completion of writing data to the currently-written FIFO buffer plane even if transmission to the other FIFO buffer is completed.
  (d) When the hardware flushes the buffer of the pipe for isochronous transfers.

(e) The FIFO buffer making the transition from the write-disabled to the write-enabled state in response to writing of 1 to the ACLRM bit.
The request trigger is not generated for the DCP (that is, during data transmission for control transfers).

(2) For the pipe in the receiving direction

(a) When packet reception is completed successfully thus enabling the FIFO buffer to be read when read-access from the CPU to the FIFO buffer for the selected pipe is disabled (when the BSTS bit is read as 0).
The request trigger is not generated for the transaction in which DATA-PID disagreement occurs.
When a short packet is received, the request trigger is generated even if the FIFO buffer has available space.
When the transaction counter is used, the request trigger is generated on receiving the specified number of packets.
In this case, the request trigger is generated even if the FIFO buffer has available space.

(b) In double-buffer operation, one FIFO buffer being ready for reading on completion of the reading of data from the other FIFO buffer.
The request trigger is not generated until completion of reading data from the currently-read FIFO buffer plane even if reception by the other FIFO buffer is completed.
The BRDY interrupt is not generated in the status stage of control transfers.
The PIPEBRDY interrupt state of the selected pipe can be cleared to 0 by writing 0 to the corresponding PIPEBRDY interrupt status bit in the BRDYSTS register. In this case, 1s should be written to the PIPEBRDY interrupt status bits for the other pipes.
Be sure to clear the BRDY state before accessing the FIFO buffer.

- When BRDYM = 0 and BFRE = 1

With these settings, this controller generates the BRDY interrupt on completion of the reading of all data for a single transfer using the pipe in the receiving direction, and sets 1 to the PIPEBRDY bit corresponding to the selected pipe.

On any of the following conditions, this controller determines that the last data for a single transfer has been received.

(1) When a short packet including a zero-length packet is received.

(2) When the transaction counter register (TRNCNT bits) is used and the number of packets specified by the TRNCNT bits is completely received.

When the pertinent data is completely read out after any of the above determination conditions has been satisfied, this controller determines that all data for a single transfer has been completely read out.

When a zero-length packet is received when the FIFO buffer is empty, the controller determines that all data for a single transfer has been completely read out upon the FRDY and the DTLN bits of the FIFO port control register are set to 1 and 0, respectively.
In this case, write 1 to the BCLR bit of the corresponding FIFOCTR register by software to start the next transfer.

With these settings, this controller does not detect the BRDY interrupt for the pipe in the transmitting direction.

The PIPEBRDY interrupt state of the selected pipe can be cleared to 0 by writing 0 to the corresponding PIPEBRDY interrupt status bit. In this case, 1s should be written to the PIPEBRDY interrupt status bits for the other pipes.

In this mode, the BFRE bit setting should not be modified until all data for a single transfer has been processed.
When it is necessary to modify the BFRE bit before completion of processing, all the FIFO buffers for the selected pipe should be cleared using the ACLRM bit.

- When BRDYM = 1 and BFRE = 0

With these settings, the PIPEBRDY values are linked to the BSTS bit settings for each pipe. In other words, the BRDY interrupt status bits (PIPEBRDY) are set to 1 or 0 by this controller depending on the FIFO buffer state.

(1) For the pipe in the transmitting direction
The setting of a BRDY interrupt status bit is 1 while the port register of the corresponding FIFO buffer is ready for written data and 0 when it is not able to accept written data.
However, the BRDY interrupt is not generated in response to setting of the bit to 1 if writing to the DCP in the transmitting direction can proceed.

(2) For the pipe in the receiving direction
The setting of a BRDY interrupt status bit is 1 while the port register of the FIFO buffer is ready for reading and 0 when all data have been read.

When a zero-length packet is received when the FIFO buffer is empty, the corresponding bit is set to 1 and the BRDY interrupt continues to be effective until 1 is written to BCLR.

With this setting, the PIPEBRDY bit cannot be cleared to 0.

When BRDYM is set to 1, all of the BFRE bits (for all pipes) should be cleared to 0.
When BRDYM is set to 1, the INTL bit should be set to 1 (level control).

24.2.8.3 NRDY Interrupt Status Register (NRDYSTS)

Address(es): A006 0048h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PIPENRDY[9:0]									
Value after hardware reset:	x	x	x	x	x	x	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b9 to b0	PIPENRDY [9:0]	NRDY Interrupt Status for Each Pipe	Indicate the NRDY interrupt state for each pipe. When writing to these bits, only 0 can be written. 0: Interrupts not generated 1: Interrupts generated	R/W	W
b15 to b10	—	Reserved	When writing to these bits, write 0.	R/W	—

Note 1. The bit number corresponds to the pipe number.
Note 2. To clear the state indicated by each bit of this register, write 0 only to the bit to be cleared and 1 to the other bits.

NRDY Interrupt Status Bit for Each Pipe (PIPENRDY)

On generating the internal NRDY interrupt request for the pipe whose PID bits are set to BUF, this controller sets the corresponding PIPENRDY bit in NRDYSTS in the register to 1. If the corresponding bit in the NRDYENB register is set to 1, this controller sets the NRDY bit in INTSTS0 in the register to 1.
The conditions on which this controller generates the internal NRDY interrupt request for a given pipe are described below.
However, the internal NRDY interrupt request is not generated during status stage execution of the control transfer.

(1) When the pipe is in the transmitting direction
  (a) When an IN Token is received while there is no transmission data in the FIFO buffer and the corresponding PIPE PID bit is set to BUF (01):
  In this case, this controller generates an NRDY interrupt request at the reception of the IN token, setting the PIPENRDY bit to 1.
  For the pipe for the isochronous transfers in which an interrupt is generated, this controller transmits a zero-length packet, setting the OVRN bit to 1.

(2) For the pipe in the receiving direction

(a) When the PID bits for the corresponding pipe are set to BUF (01) and an OUT token is received while there is no open space in the FIFO buffer:
For the pipe for the isochronous transfers in which an interrupt is generated, this controller generates an NRDY interrupt request at the reception of the OUT token, setting the PIPENRDY bit to 1 and OVRN bit to 1.
For the pipe for the transfers other than isochronous transfers in which an interrupt is generated, this controller generates an NRDY interrupt request when an NAK handshake is transferred after the data following the OUT token was received, setting the PIPENRDY bit to 1.
However, during re-transmission (due to DATA-PID disagreement), the NRDY interrupt request is not generated. In addition, if an error occurs in the DATA packet, the NRDY interrupt request is not generated.

(b) When the PID bits for the corresponding pipe are set to BUF (01) and a PING token is received while there is no open space in the FIFO buffer:
In this case, this controller generates an NRDY interrupt request at the reception of the PING token, setting the PIPENRDY bit to 1.

(c) In an isochronous transfer pipe, when the PID bits are set to BUF (01) and data is not received successfully within the interval frame:
In this case, this controller generates an NRDY interrupt request at the reception of an SOF, and sets the PIPENRDY bit to 1.

24.2.8.4 BEMP Interrupt Status Register (BEMPSTS)

Address(es): A006 004Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	PIPEBEMP[9:0]									
Value after hardware reset:	x	x	x	x	x	x	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b9 to b0	PIPEBEMP [9:0]	BEMP Interrupt Status for Each Pipe	Indicate the BEMP interrupt state for each pipe. When writing to these bits, only 0 can be written. 0: Interrupts not generated 1: Interrupts generated	R/W	W
b15 to b10	—	Reserved	When writing to these bits, write 0.	R/W	—

Note 1. The bit number corresponds to the pipe number.
Note 2. To clear the state indicated by each bit of this register, write 0 only to the bit to be cleared and 1 to the other bits.

BEMP Interrupt Status Bit for Each Pipe (PIPEBEMP)

On detecting the BEMP interrupt for the pipe whose PID bits are set to BUF by software, this controller sets the corresponding PIPEBEMP bit in BEMPSTS to 1. If the corresponding bit in BEMPENB is set to 1 by software, this controller sets the BEMP bit in INTSTS0 to 1.

The following describes the conditions on which this controller generates the internal BEMP interrupt request.

(1) For the pipe in the transmitting direction, when the FIFO buffer of the corresponding pipe is empty on completion of transmission (including zero-length packet transmission).
In single-buffer operation, the internal BEMP interrupt request is generated simultaneously with the BRDY interrupt for the pipe other than DCP.
However, the internal BEMP interrupt request is not generated on any of the following conditions.
  (a) When writing data to the FIFO buffer on the CPU (DMAC) side is started by software on completion of transmitting data of one plane in double-buffer operation.
  (b) When the buffer is cleared (emptied) by setting the ACLRM or BCLR bit to 1.
  (c) When IN transfer (zero-length packet transmission) is performed during the control transfer status stage.

(2) For the pipe in the receiving direction
When the successfully-received data packet size exceeds the specified maximum packet size.
In this case, this controller generates the BEMP interrupt request, setting the corresponding PIPEBEMP bit to 1, and discards the received data and modifies the setting of the PID bits of the corresponding pipe to STALL (11).
However, the internal BEMP interrupt request is not generated on any of the following conditions.
  (a) When a CRC error or bit stuffing error is detected in the received data.
  (b) When a setup transaction is being performed.

Writing 0 to the PIPEBEMP bit clears the state; writing 1 to the PIPEBEMP bit has no effect.

24.2.9 Frame Number Registers

24.2.9.1 Frame Number Register (FRMNUM)

Address(es): A006 004Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OVRN	CRCE	—	—	—	FRNM[10:0]										
Value after hardware reset:	0	0	x	x	x	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b10 to b0	FRNM[10:0]	Frame Number	Indicate the latest frame number.	R	W
b13 to b11	—	Reserved	When writing to these bits, write 0.	R/W	—
b14	CRCE	CRC Error Detection Status	Indicates whether a CRC error has been detected in the pipe during isochronous transfer. When writing to this bit, only 0 can be written. 0: No error 1: An error occurred.	R/W	W
b15	OVRN	Overrun/Underrun Detection Status	Indicates whether an overrun/underrun error has been detected in the pipe during isochronous transfer. When writing to this bit, only 0 can be written. 0: No error 1: An error occurred.	R/W	W

Note 1. The OVRN bit is for use in debugging. When designing a system, control the timing so that neither overrun nor underrun occurs.

Frame Number Bits (FRNM)

This controller sets these bits to indicate the latest frame number, which is updated every time an SOF packet is issued or received (every 1 ms).

When reading these bits by software, repeat reading until the same value is read twice.

CRC Error Detection Status Bit (CRCE)

In an isochronous transfer pipe, this controller sets 1 to this bit when a CRC error or bit stuffing error has been detected.

This bit can be cleared by writing 0 to it by software. In this case, write 80h when OVRN is not to be cleared at the same time.

This controller generates an internal NRDY interrupt request when it detects a CRC error. For details, see NRDY Interrupt Enable Bit for Each Pipe (PIPENRDYE).

Overrun/Underrun Detection Status Bit (OVRN)

For the isochronous transfer pipe, this controller sets 1 to this bit when an overrun/underrun is detected.
When an overrun/underrun is detected, the controller issues an internal NRDY request. Refer to NRDY Interrupt Enable Bit for Each Pipe (PIPENRDYE) for details.

This bit can be cleared writing 0 to it by software. In this case, write 40h when CRCE is not to be cleared at the same time.

This controller sets this bit to 1 on any of the following conditions.

(1) For the isochronous transfer pipe in the transmitting direction, the IN token is received before all the transmit data has been written to the FIFO buffer.
(2) For the isochronous transfer pipe in the receiving direction, the OUT token is received when no FIFO buffer planes are empty.

24.2.9.2 µFrame Number Register (UFRMNUM)

Address(es): A006 004Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UFRNM[2:0]		
Value after hardware reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b2 to b0	UFRNM[2:0]	Microframe Number	Indicate the µframe number.	R	W
b15 to b3	—	Reserved	These bits are read as an undefined value.	R	—

Microframe Number Bits (UFRNM)

This controller sets these bits to indicate the µframe number during high-speed operation. During operation other than high-speed operation, this controller sets these bits to 00h.

When reading these bits, repeat reading until the same value is read twice.

24.2.10 USB Address

24.2.10.1 USB Address Register (USBADDR)

Address(es): A006 0050h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	USBADDR[6:0]						
Value after hardware reset:	x	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b6 to b0	USBADDR [6:0]	USB Address	Indicate the USB address assigned by the host.	R	R/W
b15 to b7	—	Reserved	These bits are read as an undefined value.	R	—

USB Address Bits (USBADDR)

These bits indicate the USB address assigned by the host when the SetAddress request is successfully processed.
If a USB bus reset is detected by the controller, 00h is set to this bit.

24.2.11 USB Request Registers

The USB request registers are used to store the setup requests for control transfers. The values of the USB request that have been received are stored.

24.2.11.1 USB Request Type Register (USBREQ)

Address(es): A006 0054h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	bRequest[7:0]								bmRequestType[7:0]							
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b7 to b0	bmRequestType [7:0]	Request Type	USB request bmRequestType value	R	W
b15 to b8	bRequest[7:0]	Request	USB request bRequest value	R	W

USB Request Type Bits (bmRequestType)

These bits indicate the USB request data value received by this controller in the setup transaction. Writing to these bits by software has no effect.

USB Request Bits (bRequest)

These bits indicate the USB request data value received by this controller in the setup transaction. Writing to these bits by software has no effect.

24.2.11.2 USB Request Value Register (USBVAL)

Address(es): A006 0056h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	wValue[15:0]															
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b15 to b0	wValue[15:0]	Value	USB request wValue value	R	W

Value Bits (wValue)

These bits are used to read the value of the USB request wValue. Bits 7 to 0 are lower-order byte.

These bits indicate the USB request wValue value received by this controller in the setup transaction. Writing to these bits by software has no effect.

24.2.11.3 USB Request Index Register (USBINDX)

Address(es): A006 0058h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	wIndex[15:0]															
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b15 to b0	wIndex[15:0]	Index	USB request wIndex value	R	W

Index Bits (wIndex)

These bits are used to read the value of the USB request wIndex. Bits 7 to 0 are lower-order byte.

These bits indicate the USB request wIndex value received by this controller in the setup transaction. Writing to these bits by software has no effect.

24.2.11.4 USB Request Length Register (USBLENG)

Address(es): A006 005Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	wLength[15:0]															
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b15 to b0	wLength [15:0]	Length	USB request wLength value	R	W

Length Bit (wLength)

These bits are used to read the value of the USB request wLength. Bits 7 to 0 are lower-order byte.

These bits indicate the USB request wLength value received in setup transaction by the controller. Writing to these bits by software has no effect.

24.2.12 DCP Configuration

24.2.12.1 DCP Maximum Packet Size Register (DCPMAXP)

Address(es): A006 005Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	MXPS[6:0]						
Value after hardware reset:	0	0	0	0	x	x	x	x	x	1	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b6 to b0	MXPS[6:0]	Maximum Packet Size	Specify the maximum data payload (maximum packet size) for the DCP.	R	R/W
b15 to b7	—	Reserved	These bits are read as an undefined value.	R	—

Maximum Packet Size Bits (MXPS)

Set the maximum data payload (maximum packet size) for the DCP in these bits. 40h (64 bytes) is the value after a reset.

The MXPS bits should be set to the value based on the USB specification.
The MXPS bits should be set while PID = NAK and before setting the CURPIPE bits.
Before modifying these bits after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

While MXPS is 0, do not write to the FIFO buffer or do not set PID to BUF.

24.2.12.2 DCP Control Register (DCPCTR)

Address(es): A006 0060h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BSTS	—	—	—	—	—	—	SQCLR	SQSET	SQMON	PBUSY	—	—	CCPL	PID[1:0]	
Value after hardware reset:	0	0	0	0	0	x	x	0	0	1	0	0	x	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b1, b0	PID[1:0]	Response PID	These bits control the response type of this controller during control transfer. b1 b0 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response	R/W	R/W
b2	CCPL	Control Transfer End Enable	Setting this bit to 1 enables the status stage of the control transfer to be completed. 0: End of the control transfer disabled 1: End of the control transfer enabled	R/W	R/W
b4, b3	—	Reserved	When writing to these bits, write 0.	R/W	—
b5	PBUSY	PIPE Busy	Indicates whether the selected pipe is currently being used by the USB bus. 0: The pipe is not being used by the USB bus. 1: The pipe is being used by the USB bus.	R	W
b6	SQMON	Sequence Toggle Bit Monitor	Indicates the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: DATA0 1: DATA1	R	W
b7	SQSET	Toggle Bit Set	Specifies DATA1 as the expected value of the sequence toggle bit for the next transaction during the DCP transfer. This bit is read as 0. When writing to this bit, only 1 can be written. 0: Writing has no effect 1: Specifies DATA1	R/W	R
b8	SQCLR	Toggle Bit Clear	Specifies DATA0 as the expected value of the sequence toggle bit for the next transaction during the DCP transfer. This bit is read as 0. When writing to this bit, only 1 can be written. 0: Writing has no effect 1: Specifies DATA0	R/W	R
b10, b9	—	Reserved	When writing to these bits, write 0.	R/W	—
b11			When writing to this bit, write 1. However, this bit is read as 0.	R/W	—
b12			When writing to this bit, write 0.	R/W	—
b14, b13			When writing to these bits, write 1. However, these bits are read as 0.	R/W	—
b15	BSTS	Buffer Status	Indicates whether access to the DCP FIFO buffer is enabled or disabled. 0: Buffer access disabled 1: Buffer access enabled	R	W

Response PID Bits (PID)

Change the setting of these bits from NAK to BUF by software during data stage or status stage of control transfers.

This controller changes the setting of these bits in the following conditions:

(1) This controller changes the setting of the PID bits to select NAK response (00) on receiving the setup packet. Here, this controller sets VALID to 1. The setting of the PID bits cannot be changed until VALID is set to 0.
(2) This controller sets the PID bits to select STALL response (11) on receiving the data of the size exceeding the maximum packet size when the PID bits have been set to select BUF response.
(3) This controller sets the PID bits to select STALL response (1x) on detecting the control transfer sequence error.
(4) This controller sets the PID bits to select NAK response on detecting the USB bus reset.

This controller does not reference to the setting of the PID bits while the SET_ADDRESS request is processed (auto processing).

Control Transfer End Enable Bit (CCPL)

Setting this bit to 1 while the corresponding PID bits are set to BUF enables the stage of the control transfer to be completed.
Specifically, during control read transfer, this controller transmits the ACK handshake in response to the OUT transaction from the USB host, and outputs the zero-length packet in response to the IN transaction from the USB host during control writing or no-data control transfer. However, on detecting the SET_ADDRESS request, this controller operates in auto response mode from the setup stage up to the status stage completion irrespective of the setting of this bit.

This controller modifies this bit from 1 to 0 on receiving the new setup packet.

1 cannot be written by software to this bit while VALID is 1.

PIPE Busy Bit (PBUSY)

This controller modifies this bit from 0 to 1 upon start of the USB transaction for the selected pipe, and modifies the bit from 1 to 0 upon completion of one transaction.

Reading this bit after the PID bits have been set to NAK allows checking that modification of the pipe settings is possible.

Sequence Toggle Bit Monitor Bit (SQMON)

This bit indicates the expected value of the sequence toggle bit of the selected pipe.

This bit is toggled upon normal completion of the transaction. However, this bit is not toggled when a DATA-PID disagreement occurs during the receiving transfer.

This controller sets this bit to 1 (specifies DATA1 as the expected value) upon normal reception of the setup packet. This controller does not reference to this bit during the IN/OUT transaction of the status stage, and does not allow this bit to toggle upon normal completion.

Sequence Toggle Bit Set Bit (SQSET)

Setting this bit to 1 by software allows this controller to set DATA1 as the expected value of the sequence toggle bit of the selected pipe. This bit always indicates 0.

Do not set the SQCLR and SQSET bits to 1 simultaneously.

Set this bit to 1 while PID is NAK and before setting the CURPIPE bits.
Before setting this bit to 1 after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Clear Bit of Sequence Toggle Bit (SQCLR)

Setting this bit to 1 by software allows this controller to set DATA0 as the expected value of the sequence toggle bit of the selected pipe. This bit always indicates 0.

Do not set the SQCLR and SQSET bits to 1 simultaneously.

Set this bit to 1 while PID is NAK and before setting the CURPIPE bits.
Before setting this bit to 1 after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Buffer Status Bit (BSTS)

This bit indicates whether access from the CPU to the FIFO buffer assigned to the DCP is possible.
The meaning of the BSTS bit depends on the ISEL bit setting as follows.

(1) When ISEL = 0: indicates whether the received data can be read from the buffer.
(2) When ISEL = 1: indicates whether the data for transmission can be written to the buffer.

24.2.13 Pipe Configuration Register

Pipes 1 to 9 should be set using the PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPExCTR, PIPExTRE and PIPExTRN registers.

After selecting the pipe using the PIPESEL register, functions of the pipe should be set using the PIPECFG, PIPEBUF, PIPEMAXP and PIPEPERI registers. The PIPExCTR, PIPExTRE and PIPExTRN registers can be set regardless of the pipe selection in the PIPESEL register.

24.2.13.1 Pipe Window Select Register (PIPESEL)

Address(es): A006 0064h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL[3:0]			
Value after hardware reset:	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b3 to b0	PIPESEL [3:0]	Pipe Window Select	Specify the pipe corresponding to the registers allocated at address ranges from 68H to 6EH. b3 b0 0000: No pipe selected 0001: Pipe 1 0010: Pipe 2 0011: Pipe 3 0100: Pipe 4 0101: Pipe 5 0110: Pipe 6 0111: Pipe 7 1000: Pipe 8 1001: Pipe 9	R/W	R
b15 to b4	—	Reserved	When writing to these bits, write 0.	R/W	—

Note 1. When 0000b is set in PIPESEL, 0 is read from all of the bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers. When 0000b is set in PIPESEL, writing to the bits in these registers (at address ranges from 68H to 6EH) has no effect.

Pipe Window Select Bits (PIPESEL)

When a value between 0001b and 1001b is set in these bits, the information and settings for the corresponding pipe can be read from the registers at address ranges from A006 0068h to A006 006Eh. After a pipe is specified by setting these bits, the values set by software to the registers at address ranges from A006 0068h to A006 006Eh are reflected in the corresponding pipe transfer type by this controller.

When 0000b is set in PIPESEL, 0 is read from all of the bits from the registers at address ranges from A006 0068h to A006 006Eh. When 0000b is set in PIPESEL, writing to the bits in these registers has no effect.

24.2.13.2 Pipe Configuration Register (PIPECFG)

Address(es): A006 0068h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TYPE[1:0]		—	—	—	BFRE	DBLB	CNTMD	SHTNAK	—	—	DIR	EPNUM[3:0]			
Value after hardware reset:	0	0	x	x	x	0	0	0	0	x	x	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b3 to b0	EPNUM[3:0]	Endpoint Number	Specify endpoint number for the selected pipe.	R/W	R
b4	DIR	Transfer Direction	Specifies pipe transfer direction for the selected pipe. 0: Receiving direction 1: Transmitting direction	R/W	R
b6, b5	—	Reserved	When writing to these bits, write 0.	R/W	—
b7	SHTNAK	Pipe Disable at the End of Transfer	Specifies whether to change PID to NAK at the end of transfer when the selected pipe is in the receiving direction. 0: Pipe continued at end of transfer 1: Pipe disabled at end of transfer	R/W	R
b8	CNTMD	Continuous Transfer Mode	Specifies whether to use the selected pipe in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode	R/W	R
b9	DBLB	Double Buffer Mode	Selects either single- or double-buffer operation for the FIFO buffer used by the selected pipe. 0: Single-buffer operation 1: Double-buffer operation	R/W	R
b10	BFRE	BRDY Interrupt Operation Specification	Specifies the BRDY interrupt generation timing from this controller to the CPU with respect to the selected pipe. 0: BRDY interrupt notification upon transmitting or receiving data 1: BRDY interrupt notification upon reading data	R/W	R
b13 to b11	—	Reserved	When writing to these bits, write 0.	R/W	—
b15, b14	TYPE[1:0]	Transfer Type	Specify the transfer type for the pipe selected by the PIPESEL bits (selected pipe). b15 b14 00: Pipe is not in use. 01: Bulk transfer 10: Interrupt transfer 11: Isochronous transfer	R/W	R

Endpoint Number Bits (EPNUM)

Specify the endpoint number for the selected pipe in these bits by software.
Setting 0000b indicates that the pipe is not being used.

These bits should be modified while PID is NAK, and the pipe is not selected by the CURPIPE bits.

Before modifying these bits after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Do not make the settings such that the combination of the set values in the DIR and EPNUM bits should be the same for two or more pipes (EPNUM = 0000b (the selected pipe is not in use) can be set for all the pipes).

Transfer Direction Bit (DIR)

When this bit is set to 0 by software, this controller uses the selected pipe in the receiving direction, and when this bit is set to 1, this controller uses the selected pipe in the transmitting direction.

This bit should be modified while PID is NAK and the pipe is not selected by the CURPIPE bits.
To modify this bit after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously to clear the FIFO buffer assigned to the selected pipe while the PID and CURPIPE bits are in the above-described state.

Before modifying this bit after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Pipe Disable Bit at the End of Transfer (SHTNAK)

This bit is valid when pipes 1 to 5 are selected in the receiving direction.
When this bit is set to 1 for the selected pipe in the receiving direction, this controller modifies the PID bits for the selected pipe to NAK on determining the end of the transfer. This controller determines that the transfer has ended on any of the following conditions.

(1) A short packet (including a zero-length packet) is successfully received.
(2) The transaction counter is used and the number of packets specified for the counter is successfully received.

This bit should be modified while PID is NAK.

Before modifying this bit after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

This bit should be cleared to 0 for the pipe in the transmitting direction.

Continuous Transfer Mode Bit (CNTMD)

This bit is valid when pipes 1 to 5 are selected and the transfer type of the selected pipe is set to bulk.

According to the setting value of this bit, this controller determines transmission/reception completion for the FIFO buffer assigned to the selected pipe. See Table 24.10.

Table 24.10 Relationship between Transfer Mode Settings by CNTMD Bit and Timings at which Reading Data or Transmitting Data from FIFO Buffer is Enabled

CNTMD Bit Setting	When Reading Data or Transmitting Data is Enabled
0	In the receiving direction (DIR = 0), the FIFO buffer is ready for reading in the following case. This controller receives one packet.
	In the transmitting direction (DIR = 1), the FIFO buffer is ready for transmission in either of the following cases. (1) Data of the maximum packet size is written to the FIFO buffer by software (or DMAC). (2) Data of the short packet size (including 0-byte data) is written to the FIFO buffer and then 1 is written to BVAL by software (or DMAC).
1	In the receiving direction (DIR = 0), the FIFO buffer is ready for reading in any of the following cases. (1) The number of the data bytes received in the FIFO buffer assigned to the selected pipe is equal to the number of assigned data bytes ((BUFSIZE + 1) × 64). (2) This controller receives a short packet other than a zero-length packet. (3) This controller receives a zero-length packet when data is already stored in the FIFO buffer assigned to the selected pipe. (4) This controller receives the number of packets equal to the transaction counter value specified for the selected pipe by software.
	In the transmitting direction (DIR = 1), the FIFO buffer is ready for transmission in any of the following cases (1) to (3). (1) The number of the data bytes written to the FIFO buffer by software (or DMAC) is equal to the number of data bytes in a single FIFO buffer plane assigned to the selected pipe. (2) The number of data bytes less than the size of a single FIFO buffer plane (including 0-byte data) assigned to the selected pipe is written to the FIFO buffer and then 1 is written to BVAL by software (or DMAC) . (3) The number of data bytes less than the size of a single FIFO buffer plane (including 0-byte data) assigned to the selected pipe is written to the FIFO buffer and then the DENDx_N signal is asserted at the same time as writing the last data by software (or DMAC).

This bit should be modified while PID is NAK and the pipe is not selected by the CURPIPE bits.
To modify this bit after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously to clear the FIFO buffer assigned to the selected pipe while the PID and CURPIPE bits are in the above-described state.

Before modifying this bit after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Double Buffer Mode Bit (DBLB)

This bit is valid when pipes 1 to 5 are selected.

When this bit is set to 1, this controller assigns two planes of the FIFO buffer size specified by the BUFSIZE bits in PIPEBUF to the selected pipe.
Specifically, the following expression determines the FIFO buffer size assigned to the selected pipe by this controller.
(BUFSIZE + 1) × 64 × (DBLB + 1) [bytes]

This bit should be modified while PID is NAK and the pipe is not selected by the CURPIPE bits.
To modify this bit after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously by software to clear the FIFO buffer assigned to the selected pipe while the PID and CURPIPE bits are in the above-described state.

Before modifying this bit after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

BRDY Interrupt Operation Specification Bit (BFRE)

This bit is valid when pipes 1 to 5 are selected.

When this bit is set to 1 and the selected pipe is in the receiving direction (the DIR bit is set to 0), this controller detects the transfer completion and generates the BRDY interrupt on having read the pertinent packet.
When the BRDY interrupt is generated with the above conditions, 1 must be written to BCLR. The FIFO buffer assigned to the selected pipe is not enabled for reception until 1 is written to BCLR.

When this bit is set to 1 and the selected pipe is in the transmitting direction (the DIR bit is set to 1), this controller does not generate the BRDY interrupt. For details, refer to the description of the PIPEBRDY interrupt register.

This bit should be modified while PID is NAK and the pipe is not selected by the CURPIPE bits.
To modify this bit after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously to clear the FIFO buffer assigned to the selected pipe while the PID and CURPIPE bits are in the above-described state.

Before modifying this bit after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Transfer Type Bits (TYPE)

Specify the USB transfer type for the pipe selected by the PIPESEL bits (selected pipe) in the TYPE bits.
Table 24.11 lists the selected pipes and transfer types that can be set in the TYPE bits.

Table 24.11 Selected Pipes and Transfer Types that can be Set in TYPE Bits

Selected Pipe	TYPE Bits	USB Transfer Type
Pipe 1 or Pipe 2	01b or 11b	Bulk or isochronous transfer
Pipe 3 to Pipe 5	01b	Bulk transfer
Pipe 6 to Pipe 9	10b	Interrupt transfer

Before setting PID to BUF for the selected pipe (before starting USB communication using the selected pipe), be sure to set these bits to the value other than 00b.

These bits should be modified while the PID bits for the selected pipe are set to NAK. Before modifying these bits after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

24.2.13.3 Pipe Buffer Specification Register (PIPEBUF)

Address(es): A006 006Ah

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	BUFSIZE[4:0]					—	—	BUFNMB[7:0]							
Value after hardware reset:	x	0	0	0	0	0	x	x	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b7 to b0	BUFNMB[7:0]	Buffer number	Specify the pipe FIFO buffer number for the selected pipe. (4h to 80h)	R/W	R
b9, b8	—	Reserved	When writing to these bits, write 0.	R/W	—
b14 to b10	BUFSIZE[4:0]	Buffer size	Specify the FIFO buffer size for the pipe specified in PIPESEL bits (selected pipe). 00h: 64 bytes 01h: 128 bytes | 1Fh: 2 Kbytes	R/W	R
b15	—	Reserved	When writing to this bit, write 0.	R/W	—

Note 1. The bits in this register should be modified while PID is NAK and the pipe is not selected by the CURPIPE bits.
Note 2. Before modifying the bits of this register after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Buffer Number Bits (BUFNMB)

The first block number in the FIFO buffer to be allocated for the selected pipe should be set in these bits. The FIFO buffer blocks allocated for the selected pipe by this controller are determined as follows:
Block number: BUFNMB to block number of BUFNMB + (BUFSIZE + 1) × (DBLB + 1) - 1

These bits must be set to a value that does not exceed the size of installed memory (0 [00h] to 8192 [0x80] for 8-Kbyte memory). Observe the following conditions:
00h is used exclusively for DCP.
04h is used exclusively for pipe 6. However, when pipe 6 is not used, 04h can be used for other pipes. When pipe 6 is selected, writing to these bits has no effect and 04h is automatically assigned by this controller.
05h is used exclusively for pipe 7. However, when pipe 7 is not used, 05h can be used for other pipes. When pipe 7 is selected, writing to these bits has no effect and 05h is automatically assigned by this controller.
06h is used exclusively for pipe 8. However, when pipe 8 is not used, 06h can be used for other pipes. When pipe 8 is selected, writing to these bits has no effect and 06h is automatically assigned by this controller.
07h is used exclusively for pipe 9. However, when pipe 9 is not used, 07h can be used for other pipes. When pipe 9 is selected, writing to these bits has no effect and 07h is automatically assigned by this controller.

Buffer Size Bits (BUFSIZE)

Specify the size of the FIFO buffer for the selected pipe in these bits in terms of blocks, where one block comprises 64 bytes. When the DBLB bit is set to 1 by software, this controller assigns two planes of the FIFO buffer size specified by the BUFSIZE bits to the selected pipe. Specifically, the following expression determines the FIFO buffer size assigned to the selected pipe by this controller.
(BUFSIZE + 1) × 64 × (DBLB + 1) [bytes]

Set the following value in these bits.
(1) When pipes 1 to 5 are selected, any value from 0h to 1Fh can be set.
(2) When pipes 6 to 9 are selected, only 0h should be set.

When used with CNTMD = 1, set an integral multiple of the maximum packet size to the BUFSIZE bits.

24.2.13.4 Pipe Maximum Packet Size Register (PIPEMAXP)

Address(es): A006 006Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	MXPS[10:0]										
Value after hardware reset:	0	0	0	0	x	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b10 to b0	MXPS[10:0]	Maximum Packet Size	Specify the maximum data payload (maximum packet size) of the selected pipe. Pipes 6 to 9 can be set to 01h to 40h bytes.	R/W	R
b15 to b11	—	Reserved	When writing to these bits, write 0.	R/W	—

Note 1. The value after a reset of the MXPS bit is 00h when no pipe is selected with the PIPESEL bits in PIPESEL and 40h when a pipe is selected with the PIPESEL bits in PIPESEL.

Maximum Packet Size Bits (MXPS)

Set the maximum data payload (maximum packet size) for the selected pipe in these bits.
For pipes 1 and 2, a value between 1 byte (1h) to 1024 bytes (400h) can be set.
For pipes 3 to 5, any value from 8 bytes (8h), 16 bytes (10h), 32 bytes (20h), 64 bytes (40h) and 512 bytes (200h) can be set (bits [2:0] are not provided).
The value after a reset is 040h (64 bytes).

These bits should be set to the appropriate value for each transfer type based on the USB specification.
Set these bits to 1 while PID is NAK and before setting the CURPIPE bits.
Before modifying these bits after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

While MXPS is 0, do not write to the FIFO buffer or set PID to BUF.

24.2.13.5 Pipe Timing Control Register (PIPEPERI)

Address(es): A006 006Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV[2:0]		
Value after hardware reset:	x	x	x	0	x	x	x	x	x	x	x	x	x	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b2 to b0	IITV[2:0]	Interval Error Detection Spacing	Specify the interval error detection timing for the selected pipe in terms of frames, which is expressed as n-th power of 2.	R/W	R
b11 to b3	—	Reserved	When writing to these bits, write 0.	R/W	—
b12	IFIS	Isochronous IN Buffer Flush	Specifies whether to flush the buffer when the pipe selected by the PIPESEL bits (selected pipe) is used for isochronous IN transfers. 0: The buffer is not flushed 1: The buffer is flushed	R/W	R
b15 to b13	—	Reserved	When writing to these bits, write 0.	R/W	—

Interval Error Detection Spacing Bits (IITV)

Specify the interval error detection timing in terms of frames, which is expressed as n-th power of 2, in these bits.

These bits should be modified while PID is NAK and before setting the CURPIPE bits.
Before modifying these bits after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Before modifying these bits after USB communication has been completed with these bits set to a certain value, set PID to NAK and then set ACLRM to 1 to initialize the interval timer.

The IITV bits are invalid for pipes 3 to 5; set these bits to 0 for these pipes.

These bits can be set when the selected pipe transfer is for isochronous transfers.

(1) When the selected pipe is for isochronous OUT transfers

This controller generates the NRDY interrupt when it fails to receive a data packet within the interval set for (μ) frames by the IITV bits.

This controller generates the NRDY interrupt when this controller fails to receive a data packet because of a CRC error or other errors contained in the packet, or because of the FIFO buffer being full (because reading data from the FIFO buffer by software (DMAC) is slow).

This controller generates the NRDY interrupt on receiving an SOF packet. Even if the SOF packet is corrupted, the internal interpolation is used and allows the interrupt to be generated at the timing to receive the SOF packet. However, when the IITV bits are set to the value other than 0, this controller generates the NRDY interrupt on receiving an SOF packet for every interval after starting interval counting operation.

When the PID bits are set to NAK by software after starting the interval timer, this controller does not generate the NRDY interrupt on receiving an SOF packet.

The interval counting starts at the different timing depending on the IITV bit setting as follows.

(a) When IITV = 0: The interval counting starts when the PID bits for the selected pipe are set to BUF.

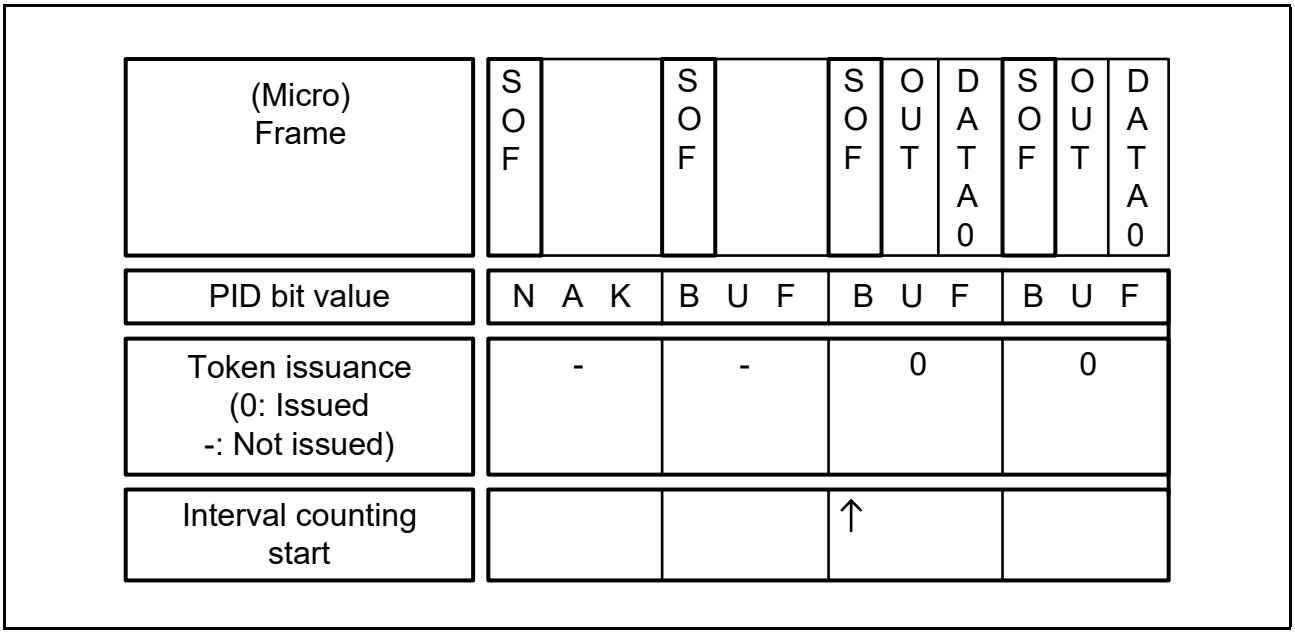


Figure 24.2 Relationship between (μ) Frames and Expected Token Reception when IITV = 0

(b) When IITV ≠ 0: The interval counting starts on completion of successful reception of the first data packet after the PID bits for the selected pipe have been modified to BUF.

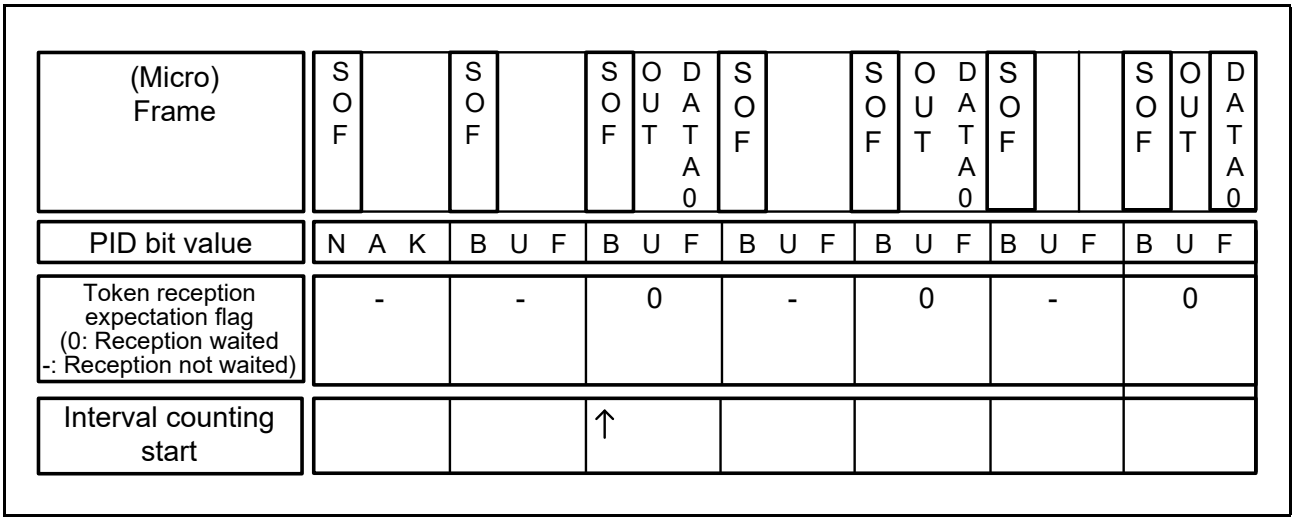


Figure 24.3 Relationship between (μ) Frames and Expected Token Reception when IITV = 1

(2) When the selected pipe is for isochronous IN transfers

The IFIS bit should be 1 for this use. When IFIS = 0, this controller transmits a data packet in response to the received IN token irrespective of the IITV bit setting.

When IFIS = 1, this controller clears the FIFO buffer when this controller fails to receive an IN token within the interval set for (μ) frames by the IITV bits in a state in which there is data to be transmitted in the FIFO buffer.
This controller also clears the FIFO buffer when this controller fails to receive an IN token successfully because of a bus error such as a CRC error contained in the token.
This controller clears the FIFO buffer on receiving an SOF packet. Even if the SOF packet is corrupted, the internal interpolation is used and allows the FIFO buffer to be cleared at the timing to receive the SOF packet.

The interval counting starts at the different timing depending on the IITV bit setting (similar to the timing during OUT transfers).

The counting conditions for the interval counter are any of the following.

(1) When a hardware reset is applied to this controller (at this point, the value of the IITV bits is also cleared to 0).
(2) When the ACLRM bit is set to 1 by software.
(3) When the controller detects a USB bus reset.

Isochronous IN Buffer Flush Bit (IFIS)

When the selected pipe is for isochronous IN transfers, this controller automatically clears the FIFO buffer when this controller fails to receive the IN token from the USB host within the interval set by the IITV bits in terms of (μ) frames. In double-buffer operation (DBLB = 1), this controller only clears the data in the plane used earlier.
This controller clears the FIFO buffer on receiving the SOF packet immediately after the (μ) frame in which this controller has expected to receive the IN token. Even if the SOF packet is corrupted, this controller also clears the FIFO buffer at the right timing to receive the SOF packet by using the internal interpolation.

When the selected pipe is not for the isochronous transfer, set this bit to 0.

24.2.14 Pipe Control Registers

24.2.14.1 PIPE1 Control Register (PIPE1CTR)
PIPE2 Control Register (PIPE2CTR)
PIPE3 Control Register (PIPE3CTR)
PIPE4 Control Register (PIPE4CTR)
PIPE5 Control Register (PIPE5CTR)

Address(es): PIPE1CTR: A006 0070h
PIPE2CTR: A006 0072h
PIPE3CTR: A006 0074h
PIPE4CTR: A006 0076h
PIPE5CTR: A006 0078h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BSTS	INBUFM	—	—	—	ATREPM	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID[1:0]	
Value after hardware reset:	0	0	0	0	x	0	0	0	0	0	0	x	x	x	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b1, b0	PID[1:0]	Response PID	Specify the response type for the next transaction of the selected pipe. b1 b0 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response	R/W	R/W
b4 to b2	—	Reserved	When writing to these bits, write 0.	R/W	—
b5	PBUSY	PIPE Busy	Indicates whether or not the selected pipe is being currently used for the USB bus. 0: The pipe is not being currently used for the USB bus. 1: The pipe is being currently used for the USB bus.	R	W
b6	SQMON	Toggle Bit Confirm	Sets the expected value of the sequence toggle bit in the next transaction of the selected pipe. 0: DATA0 1: DATA1	R	W
b7	SQSET	Toggle Bit Set	Specifies 1 when setting the expected value of the sequence toggle bit in the next transaction of the selected pipe, to DATA1. This bit is read as 0. When writing to this bit, only 1 can be written. 0: Writing has no effect. 1: Specifies DATA1	R/W	R
b8	SQCLR	Toggle Bit Clear	Specifies 1 when clearing the expected value of the sequence toggle bit in the next transaction of the selected pipe, to DATA0. This bit is read as 0. When writing to this bit, only 1 can be written. 0: Writing has no effect. 1: Specifies DATA0	R/W	R
b9	ACLRM	Auto Buffer Clear Mode	Enables or disables automatic buffer clear mode for the selected pipe. 0: Disabled 1: Enabled (all buffers are initialized)	R/W	R

Bit	Symbol	Bit Name	Description	S/W	H/W
b10	ATREPM	Auto Response Mode	Enables or disables auto response mode for the selected pipe. 0: Disabled 1: Enabled (a zero-length packet response while sending, NAK response and an NRDY interrupt is issued while receiving)	R/W	R
b12, b11	—	Reserved	When writing to these bits, write 0.	R/W	—
b13			When writing to this bit, write 1. However, this bit is read as 0.	R/W	—
b14	INBUFM	Transfer Buffer Monitor	Indicates the selected FIFO buffer state when the selected pipe is in the transmitting direction. 0: FIFO buffer contains no transmittable data. 1: FIFO buffer contains transmittable data.	R	W
b15	BSTS	Buffer Status	Indicates the FIFO buffer state of the selected pipe. 0: Buffer access is disabled. 1: Buffer access is enabled.	R	W

Response PID Bits (PID)

Set a response of the controller in each pipe in this bit by the software.

The default value of this bit is NAK. When executing a USB transfer through the selected pipe, change the setting of the PID bits to select BUF. The basic operations (operations when there is no error in the communication packet) of this controller for each value of the PID bit are given in Table 24.12.

To change the PID bits from BUF to NAK by a software while USB communications through the selected pipe is in progress, confirm that the USB transfer status of the known pipe has changed to NAK by reading PBUSY = 0. Note that reading of the PBUSY bit by a software is not required if the setting of the PID bits was changed to NAK by the controller.

In following cases, the controller modifies the value of this bit:

(1) When the selected pipe is for receiving and when the software has written 1 to the SHTNAK bit of the selected pipe, the controller sets PID = NAK upon identifying the transfer end.

(2) For the selected pipe, when the data packet of payload exceeding the maximum packet size is received, the controller sets PID = STALL (11b).

(3) When the USB bus reset is detected, the controller sets PID = NAK.

Write 10b to shift from PID = NAK (00b) to PID = STALL.
Write 11b to shift from BUF (01b) to STALL.
First write 10b and then write 00b to shift form STALL (11b) to NAK.
First, shift to NAK and then to BUF to shift from STALL to BUF.

Table 24.12 List of Controller Operations According to the PID Bit

PID Bit Value	Transfer Type (TYPE Bits Value)	Transfer Direction (DIR Bit Value)	Controller Operations
00b (NAK)	Bulk (TYPE = 01b) or Interrupt (TYPE = 10b)	Not dependent on setup value	NAK response is sent to the token from the USB host.
	Isochronous (TYPE = 11b)	Reception (DIR = 0)	Does not respond to the token from the USB host.
		Transmission (DIR = 1)	A zero-length packet is sent to the token from the USB host.
01b (BUF)	Bulk (TYPE = 01b)	Reception (DIR = 0)	When an OUT token is sent from the USB host, if the FIFO buffer for the selected pipe is ready for reception, the data is received and an ACK or NYET response is returned. A NAK response is returned if not ready. When a PING token is sent from the USB host, if the FIFO buffer of the selected pipe is ready for reception, an ACK response is returned. A NAK response is returned if not ready.
	Interrupt (TYPE = 10b)	Reception (DIR = 0)	For the Out token from the USB host, if the FIFO buffer of the selected pipe is ready for reception, the data is received and an ACK response is sent. A NAK response is returned if not ready.
	Bulk (TYPE = 01b) or Interrupt (TYPE = 10b)	Transmission (DIR = 1)	If the corresponding FIFO buffer is ready for transmission, the data is sent for the token from the USB. A NAK response is returned if not ready.
	Isochronous (TYPE = 11b)	Reception (DIR = 0)	For the Out token from the USB host, if the FIFO buffer of the selected pipe is ready for reception, the data is received. The data is deleted if not ready.
		Transmission (DIR = 1)	If the corresponding FIFO buffer is ready for transmission, the data is sent for the token from the USB. A zero-length packet is sent if not ready.
10b (STALL) or 11b (STALL)	Bulk (TYPE = 01b) or Interrupt (TYPE = 10)	Not dependent on setup value	A STALL response is sent to the token from the USB host.
	Isochronous (TYPE = 11b)	Not dependent on setup value	Returns nothing in response to the token from the USB host.

PIPE Busy Bit (PBUSY)

This controller modifies this bit from 0 to 1 upon start of the USB transaction for the selected pipe, and modifies the bit from 1 to 0 upon completion of one transaction.

Reading this bit after PID has been set to NAK allows checking that modification of the pipe settings is possible.

Sequence Toggle Bit Monitor Bit (SQMON)

This bit indicates the expected value of the sequence toggle bit of the selected pipe.

When the selected pipe is not for the isochronous transfer, this bit is toggled upon normal completion of the transaction. However, this bit is not toggled when a DATA-PID disagreement occurs during the receiving transfer.

Sequence Toggle Bit Set Bit (SQSET)

Setting this bit to 1 by software allows this controller to set DATA1 as the expected value of the sequence toggle bit of the selected pipe. This bit always indicates 0.

Set this bit to 1 while PID is NAK.
Before setting this bit to 1 after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Sequence Toggle Bit Clear Bit (SQCLR)

Setting this bit to 1 by software allows this controller to set DATA0 as the expected value of the sequence toggle bit of the selected pipe. This bit always indicates 0.

Set this bit to 1 while PID is NAK.
Before setting this bit to 1 after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Auto Buffer Clear Mode Bit (ACLRM)

To clear the contents in the FIFO buffer assigned to the selected pipe completely, write 1 and then 0 to this bit continuously.
Table 24.13 shows the contents cleared by writing 1 and 0 to this bit continuously.
Table 24.14 shows the cases in which clearing the contents is necessary.

Table 24.13 Contents Cleared by this Controller by Setting ACLRM = 1

No.	Contents Cleared by ACLRM Bit Manipulation
(1)	All the contents in the FIFO buffer assigned to the selected pipe (all the information in two FIFO buffer planes in double-buffer operation)
(2)	The interval count value when the selected pipe is for isochronous transfer

Table 24.14 Cases where Setting ACLRM = 1 is Required

No.	Case where Clearing is Required
(1)	All the contents in the FIFO buffer assigned to the selected pipe are to be cleared.
(2)	The interval counter is to be reset.
(3)	When the BFRE setting is modified
(4)	When the DBLB setting is modified
(5)	When the transaction count function is forcibly terminated

Set this bit to 1 while PID is NAK and before specifying the pipe in CURPIPE bits. Before setting this bit to 1 after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Auto Response Mode Bit (ATREPM)

When the selected pipe is for bulk transfer, this bit can be set to 1.
When this bit is set to 1, this controller responds to the token from the USB host as described below.

(1) When the selected pipe is for bulk IN transfer (when TYPE = 01b and DIR = 1)
When ATREPM = 1 and PID = BUF, this controller transmits a zero-length packet in response to the IN token. This controller updates (toggles) the sequence toggle bit (DATAPID) each time this controller receives the ACK from the USB host (in a single transaction, IN token is received, zero-length packet is transmitted, and then ACK is received). In this case, this controller does not generate the BRDY or BEMP interrupt.

(2) When the selected pipe is for bulk OUT transfer (when TYPE = 01b and DIR = 0)
When ATREPM = 1 and PID = BUF, this controller returns NAK in response to the OUT token (or PING token) and generates the NRDY interrupt.

This bit should be modified while PID is NAK. Before setting this bit to 1 after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

For USB communication in auto response mode, set this bit to 1 while the FIFO buffer is empty. Do not write to the FIFO buffer during USB communication in auto response mode.

When the selected pipe is for isochronous transfer, be sure to set this bit to 0.

Transmission Buffer Monitor Bit (INBUFM)

When the selected pipe is in the transmitting direction (DIR = 1), this controller sets this bit to 1 when writing to at least one FIFO buffer plane is completed by software (or DMAC).
This controller sets this bit to 0 when this controller completes transmitting the data from the FIFO buffer plane to which all data has been written. In double-buffer operation (DBLB = 1), this controller sets this bit to 0 after this controller has completed transmitting the data from both FIFO buffer planes but before it has completed writing data to a single FIFO buffer plane.

This bit indicates the same value as the BSTS bit when the selected pipe is in the receiving direction (DIR = 0).

Buffer Status Bit (BSTS)

This bit indicates whether access from the CPU to the FIFO buffer assigned to the selected pipe is enabled or disabled.
The meaning of the BSTS bit depends on the setting of the DIR, BFRE and DCLRM bits as follows.

Table 24.15 BSTS Bit Operations

DIR Bit Value	BFRE Bit Value	DCLRM Bit Value	Meaning of BSTS Bit
0	0	0	1: The received data can be read from the FIFO buffer. 0: The received data has been completely read from the FIFO buffer.
		1	Setting prohibited
	1	0	1: The received data can be read from the FIFO buffer. 0: The BCLR bit has been set to 1 after the received data has been completely read from the FIFO buffer.
		1	1: The received data can be read from the FIFO buffer. 0: The received data has been completely read from the FIFO buffer.
1	0	0	1: The transmit data can be written to the FIFO buffer. 0: The transmit data has been completely written to the FIFO buffer.
		1	Setting prohibited
	1	0	Setting prohibited
		1	Setting prohibited

24.2.14.2 PIPE6 Control Register (PIPE6CTR)
PIPE7 Control Register (PIPE7CTR)
PIPE8 Control Register (PIPE8CTR)
PIPE9 Control Register (PIPE9CTR)

Address(es): PIPE6CTR: A006 007Ah
PIPE7CTR: A006 007Ch
PIPE8CTR: A006 007Eh
PIPE9CTR: A006 0080h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	BSTS	—	—	—	—	—	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID[1:0]	
Value after hardware reset:	0	x	0	0	x	x	0	0	0	0	0	x	x	x	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b1, b0	PID[1:0]	Response PID	Specify the response type for the next transaction of the selected pipe. b1 b0 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response	R/W	R/W
b4 to b2	—	Reserved	When writing to these bits, write 0.	R/W	—
b5	PBUSY	PIPE Busy	Indicates whether or not the selected pipe is being currently used for the USB bus. 0: The pipe is not being currently used for the USB bus. 1: The pipe is being currently used for the USB bus.	R	W
b6	SQMON	Toggle Bit Confirm	Sets the expected value of the sequence toggle bit in the next transaction of the selected pipe. 0: DATA0 1: DATA1	R	W
b7	SQSET	Toggle Bit Set	Specifies 1 while clearing the expected value of the sequence toggle bit in the next transaction of the selected pipe to DATA1.This bit is read as 0. When writing to this bit, write 1. 0: Writing has no effect. 1: Specifies DATA1	R/W	R
b8	SQCLR	Toggle Bit Clear	Specifies 1 while clearing the expected value of the sequence toggle bit in the next transaction of the selected pipe to DATA0. This bit is read as 0. When writing to this bit, write 1. 0: Writing has no effect. 1: Specifies DATA0	R/W	R
b9	ACLRM	Auto Buffer Clear Mode	Enables or disables automatic buffer clear mode for the selected pipe. 0: Disabled 1: Enabled (all buffers are initialized)	R/W	R
b12 to b10	—	Reserved	When writing to these bits, write 0.	R/W	—
b13			When writing to this bit, write 1. However, this bit is read as 0.	R/W	—
b14			When writing to this bit, write 0.	R/W	—
b15	BSTS	Buffer Status	Indicates the FIFO buffer state of the selected pipe. 0: Buffer access is disabled 1: Buffer access is enabled	R	W

As for the bits listed below, see the description of each bit in section 24.2.14.1, PIPE1 Control Register (PIPE1CTR) PIPE2 Control Register (PIPE2CTR) PIPE3 Control Register (PIPE3CTR) PIPE4 Control Register (PIPE4CTR) PIPE5 Control Register (PIPE5CTR).

- Response PID Bits (PID)
- PIPE Busy Bit (PBUSY)
- Buffer Status Bit (BSTS)
- Sequence Toggle Bit Monitor Bit (SQMON)
- Sequence Toggle Bit Set Bit (SQSET)
- Sequence Toggle Bit Clear Bit (SQCLR)

Auto Buffer Clear Mode Bit (ACLRM)

To clear all contents from the FIFO buffer allocated to the selected pipe, write 1 and 0 sequentially in the ACLRM bit. Table 24.16 shows the contents of data to be cleared by the controller when 1 and 0 are written to this bit sequentially. The cases that require this processing are listed in Table 24.17.

Table 24.16 Contents Cleared by the Controller when ACLRM = 1

No.	Contents Cleared by ACLRM Bit Manipulation
(1)	Entire contents of the FIFO buffer allocated to the selected pipe

Table 24.17 Cases where Setting ACLRM = 1 is Required

No.	Cases where Clearing is Required
(1)	All the contents in the FIFO buffer assigned to the selected pipe are to be cleared.
(2)	The interval counter is to be reset.
(3)	The value of the BFRE bit is changed.
(4)	The transaction count function is forcibly terminated.

The ACLRM bit should be set while PID = NAK and before specifying the pipe in the CURPIPE bits.
Before modifying this bit after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY is 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

24.2.15 Transaction Counters

24.2.15.1 PIPE1 Transaction Counter Enable Register (PIPE1TRE)
PIPE2 Transaction Counter Enable Register (PIPE2TRE)
PIPE3 Transaction Counter Enable Register (PIPE3TRE)
PIPE4 Transaction Counter Enable Register (PIPE4TRE)
PIPE5 Transaction Counter Enable Register (PIPE5TRE)

Address(es): PIPE1TRE: A006 0090h
PIPE2TRE: A006 0094h
PIPE3TRE: A006 0098h
PIPE4TRE: A006 009Ch
PIPE5TRE: A006 00A0h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	TRENB	TRCLR	—	—	—	—	—	—	—	—
Value after hardware reset:	x	x	x	x	x	x	0	0	x	x	x	x	x	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b7 to b0	—	Reserved	When writing to these bits, write 0.	R/W	—
b8	TRCLR	Transaction Counter Clear	The transaction counter can be cleared to 0 by writing 1 to this bit. This bit is read as 0. When writing to this bit, write 1. 0: Invalid 1: The current counter value is cleared	R/W	R
b9	TRENB	Transaction Counter Enable	Enable or disables the transaction counter 0: The transaction counter is disabled 1: The transaction counter is enabled	R/W	R
b15 to b10	—	Reserved	When writing to these bits, write 0.	R/W	—

Note 1. Modify each bit in this register while PID is NAK.
Before modifying each bit after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.

Transaction Counter Clear Bit (TRCLR)

When this bit is set to 1 by software, this controller clears the current counter value of the transaction counter corresponding to the selected pipe and then sets this bit to 0.

Transaction Counter Enable Bit (TRENB)

For the pipe in the receiving direction, setting this bit is set to 1 by software after setting the total number of the packets to be received in the TRNCNT bits allows the following control on having received the number of packets equal to the set value in the TRNCNT bits.

(1) In continuous transmission/reception mode (CNTMD = 1), this module switches the FIFO buffer to the CPU side even if the FIFO buffer is not full on completion of reception.
(2) While SHTNAK is 1, this module changes the setting of the PID bits to NAK for the corresponding pipe on having received the number of packets equal to the set value in the TRNCNT bits.
(3) While DENDE is 1 and PKTMD is 0, the DEND signal is asserted when the number of packets specified in the TRNCNT bits is received and the last data is to be read.
(4) While BFRE is 1, this module asserts the BRDY interrupt on having received the number of packets equal to the set value in the TRNCNT bits and then reading out the last received data.

For the pipe in the transmitting direction, set this bit to 0.
When the transaction counter is not used, set this bit to 0.
When the transaction counter is used, set the TRNCNT bits before setting this bit to 1. Set this bit to 1 before receiving the first packet to be counted by the transaction counter.

24.2.15.2 PIPE1 Transaction Counter Register (PIPE1TRN)
PIPE2 Transaction Counter Register (PIPE2TRN)
PIPE3 Transaction Counter Register (PIPE3TRN)
PIPE4 Transaction Counter Register (PIPE4TRN)
PIPE5 Transaction Counter Register (PIPE5TRN)

Address(es): PIPE1TRN: A006 0092h
PIPE2TRN: A006 0096h
PIPE3TRN: A006 009Ah
PIPE4TRN: A006 009Eh
PIPE5TRN: A006 00A2h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TRNCNT[15:0]															
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b15 to b0	TRNCNT [15:0]	Transaction Counter	When written to: Specify the total number of reception packets (number of transactions) to be received by the selected pipe. When read from: Indicate the specified number of transactions if TRENB is 0. Indicate the number of currently counted transaction if TRENB is 1.	R/W	R/W

Transaction Counter Bits (TRNCNT)

For the pipe in the receiving direction, setting these bits to 1 by software after setting the total number of the packets to be received in the TRNCNT bits allows the control of the transaction counter clear bit (TRCLR).
When TRENB = 0, these bits indicate the number of transactions set by software.
When TRENB = 1, these bits indicate the number of transactions being currently counted.

This controller increments the value of these bits by one when all of the following conditions are satisfied on receiving the packet.

(a) TRENB = 1
(b) (TRNCNT setting ≠ current counter value +1) on receiving the packet
(c) The payload of the received packet agrees with the set value in the MXPS bits.

This module clears the value of these bits to 0 when any of the following conditions are satisfied.

(1) When all the following conditions are satisfied:
(a) TRENB = 1
(b) (TRNCNT setting = current counter value +1) on receiving the packet
(c) The payload of the received packet agrees with the set value in the MXPS bits.

(2) When both of the following conditions are satisfied:
(a) TRENB = 1
(b) A short packet is received.

(3) When the following condition is satisfied:
(a) When the TRCLR bit is set to 1 by software

For the pipe in the transmitting direction, set these bits to 0. When the transaction counter is not used, set these bits to 0.

These bits should be modified while PID is NAK and TRENB is 0.
Before modifying these bits after changing the setting of the PID bits for the selected pipe from BUF to NAK, check that PBUSY = 0. However, checking of the PBUSY bit is not required if the setting of the PID bits has been changed to select NAK response by the controller.
To modify the value of these bits, set TRCLR to 1 before setting TRENB to 1.

24.2.16 Low-Power Status Register (LPSTS)

Address(es): A006 0102h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	SUSPM	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after hardware reset:	x	0	x	0	x	x	x	0	x	x	x	x	0	x	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b13 to b0	—	Reserved	When writing to these bits, write 0.	R/W	—
b14	SUSPM	UTMI SuspendM Control	Controls the SuspendM signal to the UTMI. 0: UTMI suspend mode 1: UTMI normal mode	R/W	R/W
b15	—	Reserved	When writing to this bit, write 0.	R/W	—

UTMI SuspendM Control Bit (SUSPM)

The USBf module controls output of the clock signal from the PLL in the USB-PHY layer. Accordingly, supply of the clock signal to the USBf module is stopped while SUSPM = 0.
After setting this bit to 1, wait for at least 100 us to allow oscillation of the UTMI clock to become stable.

Writing to this controller is disabled when the SUSPM bit is set to 0 (the UTMI clock is stopped). Reading from the controller area is allowed. However, writing to the registers listed in Table 24.18 is enabled even when the SUSPM bit is set to 0.

Table 24.18 List of Registers that can be Written by Software when SUSPM = 0

Address	Register Name
A006 0000h	SYSCFG0
A006 0002h	SYSCFG1
A006 0102h	LPSTS

However, the value written to the SYSCFG0 register while the UTMI clock is stopped (SUSPM = 0) is reflected after the UTMI clock oscillation is started (SUSPM = 1).

24.2.17 FIFO Continuous Transfer Ports

24.2.17.1 D0FIFO Continuous Transfer Port Register n (D0FIFOBn) (n = 0 to 7)
D1FIFO Continuous Transfer Port Register n (D1FIFOBn) (n = 0 to 7)

Address(es): D0FIFOBn: A006 0160h
D1FIFOBn: A006 0180h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	FIFOPORT[31:0]															
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	FIFOPORT[31:0]															
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

Bit	Symbol	Bit Name	Description	S/W	H/W
b31 to b0	FIFOPORT [31:0]	FIFO Port	Accessing these bits allows reading the received data from the FIFO buffer or writing the transmit data to the FIFO buffer.	R/W	R/W

FIFO Port Control

When DFACC is set to 01b or 10b (16-byte/32-byte continuous access mode), use the DxFIFO continuous transfer port registers for access to the DxFIFO buffers.

24.2.18 PHY Setting Register 1

24.2.18.1 PHY Setting Register 1 (PHYSET1)

Address(es): A006 01A0h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	PHYVBUSIN	PHYRESET	PHYPD	P1PORTSEL[1:0]	
Value after hardware reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
Value after USB bus reset:	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x

x: Undefined

Bit	Symbol	Bit Name	Description	S/W	H/W
b1, b0	P1PORTSEL[1:0]	Port Connection Path Select Input Signal	b1 b0 00: Signals are not driven. 01: USB host controller 10: Setting prohibited 11: USB function controller	R/W	R/W
b2	PHYPD	USB PHY Power-Down Signal	0: Normal mode 1: Power-down mode	R/W	R/W
b3	PHYRESET	USB PHY Reset	0: Release from the reset state 1: The reset signal is asserted.	R/W	R/W
b4	PHYVBUSIN	Setting for VBUS Input to USB PHY	Enables the pull-up resistors for the D+ signal. 0: No VBUS input (D+ pull-up resistor disabled) 1: VBUS input (D+ pull-up resistor enabled)	R/W	R/W
b15 to b5	—	Reserved	When writing to these bits, write 0.	R/W	—

Note: Placing the PLL in the PHY layer in power-down mode is prohibited once the PHY-PLL has been activated.

P1PORTSEL[1:0]

Port connection path selection input signals
00b: Signals are not driven.
01b: USB host controller
10b: Setting prohibited
11b: USB function controller

PHYPD

USB PHY power-down signal
0: Normal mode
1: Power-down mode

PHYRESET

USB PHY reset input
0: Release from the reset state
1: The reset signal is asserted.

PHYVBUSIN

This bit sets the VBUS input to the USB PHY layer. The setting of this bit enables the pull-up resistor for the D+ signal.
0: No VBUS input (D+ pull-up resistor disabled)
1: VBUS input (D+ pull-up resistor enabled)

24.3 Operation

24.3.1 System Controls and Oscillation Controls

This section describes the register operations required for initial settings of this module, and the registers necessary for power consumption control. The startup sequence is as follows.

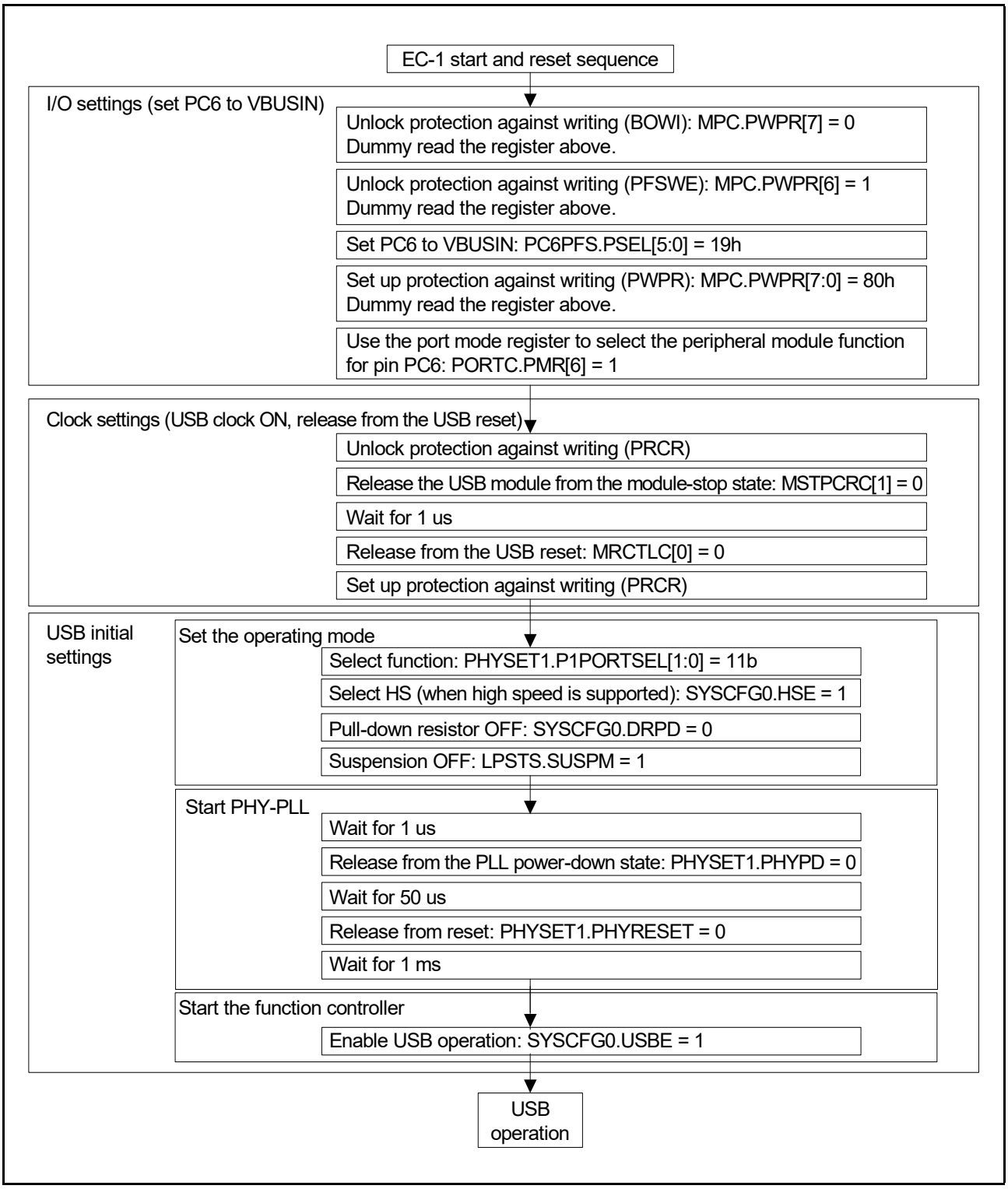


Figure 24.4 Startup Sequence

24.3.2 Resets

Table 24.19 lists the types of reset for this controller. For the initial states of the registers following the reset operations, refer to section 5, I/O Registers.

Table 24.19 Types of Reset

Name	Operation
USB bus reset	Automatically detected by this module from the D+ and D- lines

24.3.3 USB Data Bus Resistor Controller

This controller controls the pull-up resistors for the D+ signal of the USB2.0-PHY port.
Make pull-up setting for the D+ signal using the DPRPU and DRPD bits of the SYSCFG0 register.

Confirm the connection to the USB Host before setting the DPRPU bit of the SYSCFG0 register to 1 to pull up D+.
Set PHYSET1.PHYVBUSIN to 1 at the same time.

This controller incorporates termination resistors for the D+ and D- signals (in high-speed operation mode) and output resistors (in full-speed operation mode). The switching of the internal resistors following the connection to the PC is automatically done by the controller during reset handshake, suspend, and resume processing.

If the DPRPU bit of the SYSCFG0 register is set to 0 during communication with the PC, the controller disables the pull-up resistors (or termination resistors) of the USB data lines, so that it can notify the host controller of the disconnection of the device. At that time, set PHYSET1.PHYVBUSIN to 0 at the same time.

24.3.4 Supply of Clocks

Table 24.20 lists two clocks required for this controller.

Table 24.20 Input Clocks

Input Clock Name	Description
CPU clock (CPUCLK)	CPU clock input There are no restrictions on the clock frequency.
PHY clock (UTMI clock)	PHY clock input 60 MHz is internally supplied.

24.3.5 Notes on Stopping Clocks

- The PHY clock can be stopped with the SUSPM register in suspend mode.
- If a clock is to be stopped while the controller is in USB suspended state, however, it is necessary to resume the supply of the clock during resume processing. The resumption of the PHY clock need be accomplished within 5.5 ms after a resume interrupt occurs.
- After starting the PLL in the USB-PHY, do not stop it by setting the PHY power down bit (PHYSET1.PHYPD) or the PHY reset bit (PHYSET1.PHYRESET).
- When the USB function is not in use, place the USB module in the module-stop state (MSTPCRC[1] = 1 (MRCTLC[0] = 1 for making the USB reset setting)).

24.4 Interrupt Functions

24.4.1 Overview of Interrupt Functions

Table 24.21 lists the interrupt functions of this controller.

Table 24.21 List of Interrupt Functions

Bit	Interrupt Name	Interrupt Source	Related Status
VBINT	VBUS interrupt	When a change in the state of the VBUS input pin has been detected (low to high or high to low).	VBSTS
RESM	Resume interrupt	When a change in the state of the USB bus has been detected in the suspended state (J-state to K-state or J-state to SE0)	—
SOFR	Frame number update interrupt	If SOFRM = 0: When an SOF packet with a different frame number is received. If SOFRM = 1: When an SOF packet for a μ frame number of 0 cannot be received due to a corrupted packet.	—
DVST	Device state transition interrupt	When transition in device state is detected: A USB bus reset detected The suspended state detected Set Address request received Set Configuration request received	DVSQ
CTRT	Control transfer stage transition interrupt	When a stage transition is detected in control transfer: Setup stage completed Control write transfer status stage transition Control read transfer status stage transition Control transfer completed A control transfer sequence error occurred	CTSQ
BEMP	Buffer empty interrupt	When the buffer becomes empty after all data in the buffer memory has been transmitted. When a packet exceeding the maximum packet size is received.	PIPEBEMP
NRDY	Buffer not ready interrupt	When a token is received when PID = BUF and the buffer memory is not ready to transmit or receive data. When a CRC error or bit stuff error occurs while receiving data in isochronous transfer. When an interval error occurs while receiving data in isochronous transfer.	PIPENRDY
BRDY	Buffer ready interrupt	When the buffer is ready (readable or writable).	PIPEBRDY

Table 24.22 shows operations for a USBf interrupt output from this controller. In case more than one interrupt source is generated, the USBf interrupt output method can be set by using the INTL bit in the SOFCFG register. Set the USBf interrupt output operation according to the user system.

Table 24.22 Operations for USBf Interrupt Output

USBf Interrupt Output Pin Operation / INTL Setting	When One Interrupt Source is Generated	When More than One interrupt Source is Generated
Edge sense (INTL = 0)	Holds the pin low until the interrupt source is cleared.	Negated (H pulse output) for 32 clock cycles at 48 MHz when one source is cleared.
Level sense (INTL = 1)	Holds the pin low until the interrupt source is cleared.	Holds the pin low until all sources are cleared.

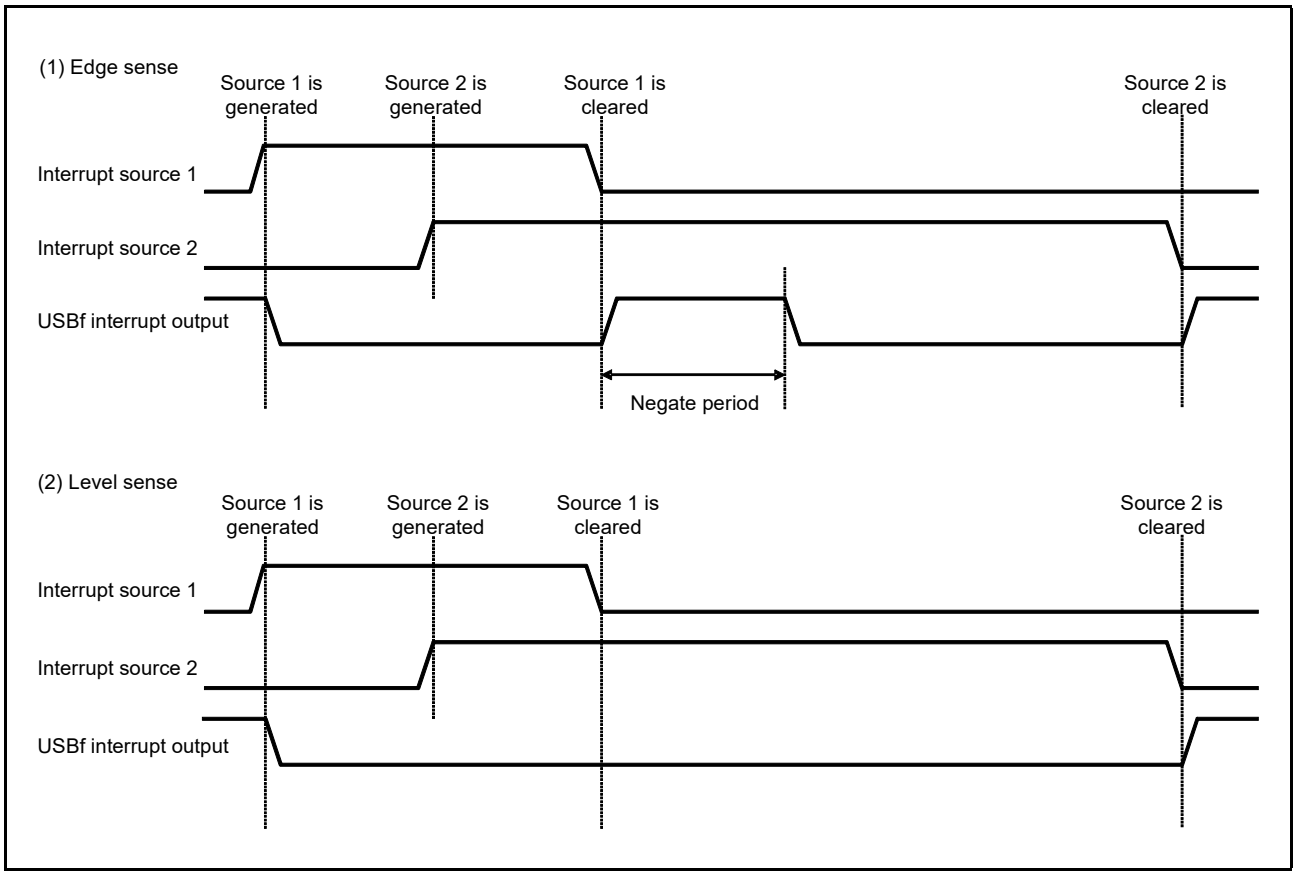


Figure 24.5 USBf Interrupt Output Operation Diagram

Figure 24.6 shows the interrupt configurations for the controller.

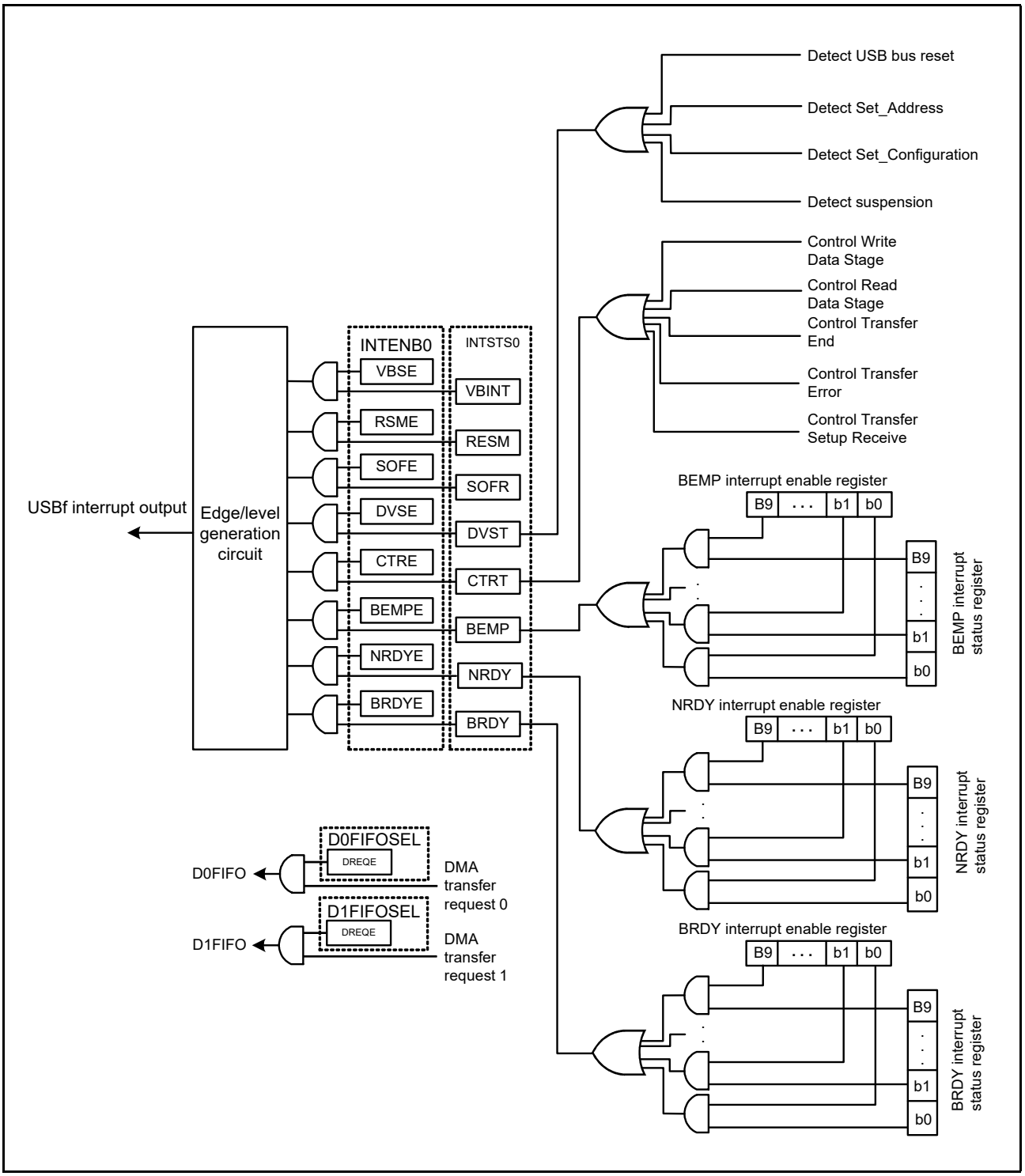
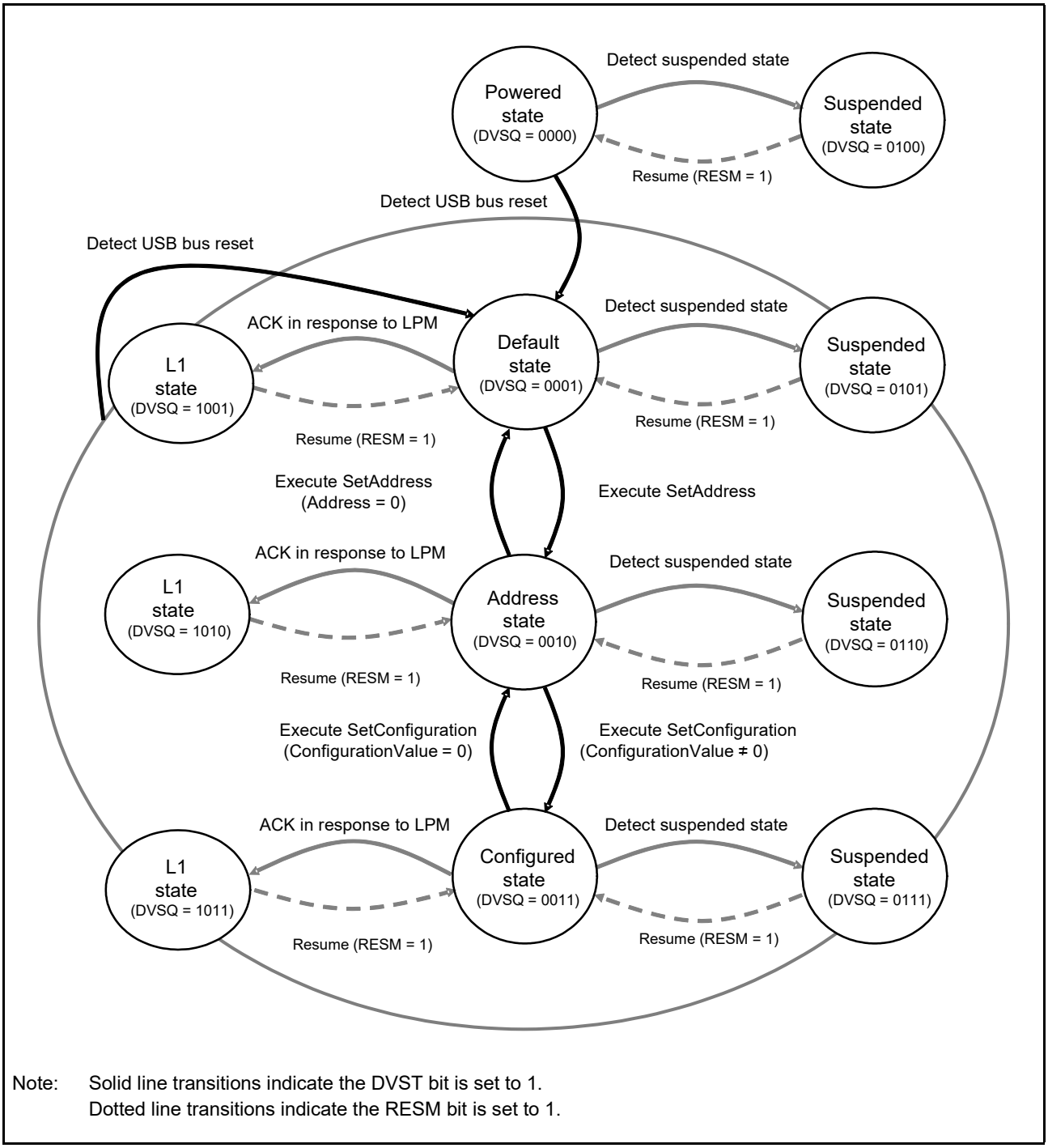


Figure 24.6 Interrupt Configuration Diagram

24.4.2 Device State Transition Interrupt

Figure 24.7 shows a diagram of how this module handles the device state transitions. The controller monitors the device states and generates the device state transition interrupts. However, recovery from the suspended state (resume signal detection) is detected by the resume interrupt. The device state transition interrupts can be enabled or disabled individually by setting the INTENB0 register. The device state after a transition can be confirmed using the DVSQ bits in the INTSTS0 register.

When making a transition to the default state, the device state transition interrupt is generated after the reset handshake protocol has been completed.



Note: Solid line transitions indicate the DVST bit is set to 1.
Dotted line transitions indicate the RESM bit is set to 1.

Figure 24.7 Device State Transitions

24.4.3 Control Transfer Stage Transition Interrupt

Figure 24.8 shows a diagram of how this module handles the control transfer stage transition. The controller monitors the control transfer sequence and generates the control transfer stage transition interrupts. The control transfer stage transition interrupts can be enabled or disabled individually in the INTENB0 register. The control transfer stage after a transition can be confirmed in the CTSQ bits in the INTSTS0 register.

The control transfer sequence errors are described below. If an error occurs, the PID bits in DCPCTR are set to 1xb (STALL).

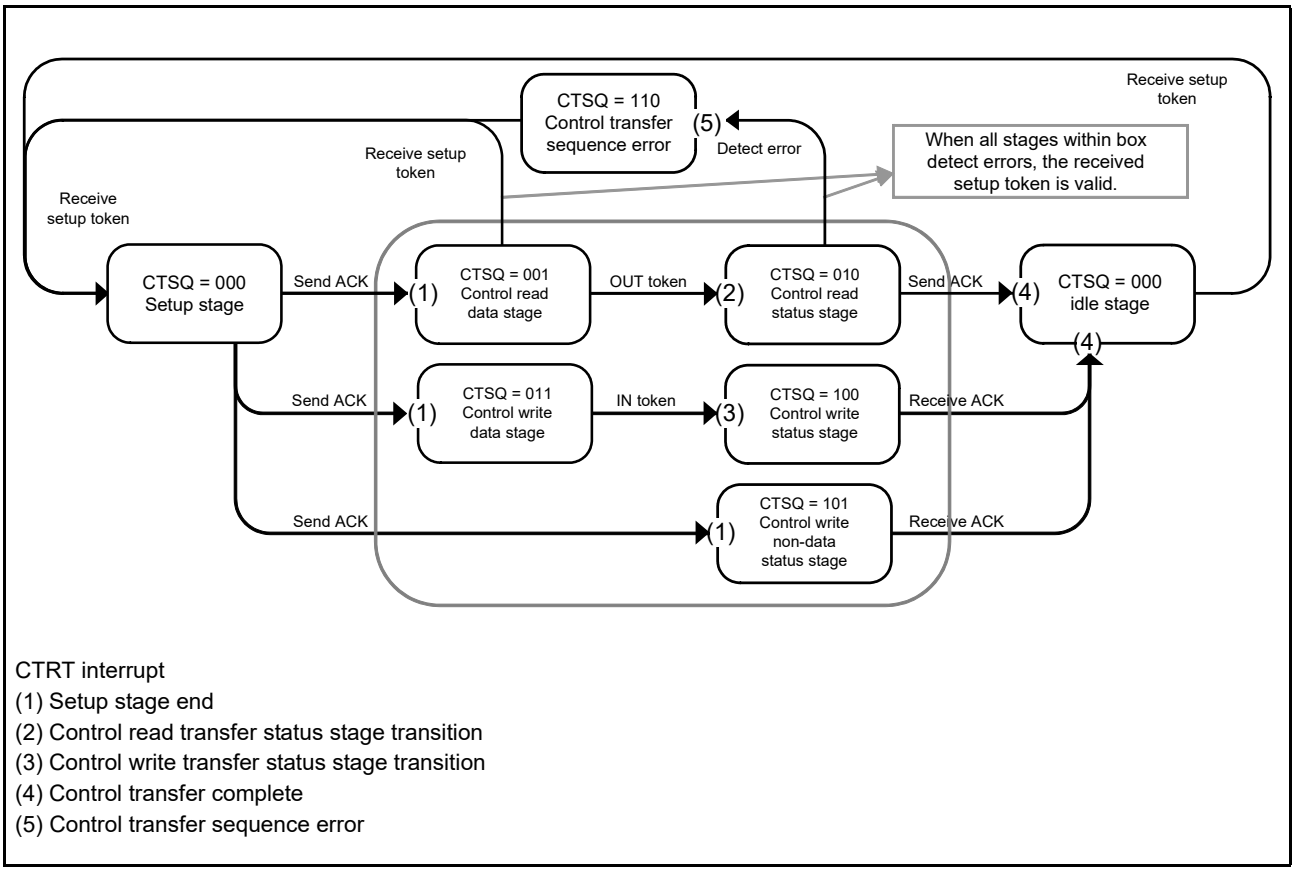
(1) During control read transfers
  (a) At the IN token of the data stage, an OUT or PING token is received when there have been no data transfers at all.
  (b) An IN token is received at the status stage.
  (c) A packet is received at the status stage for which the data packet is DATAPID = DATA0.

(2) During control write transfers
  (a) At the OUT token of the data stage, an IN token is received when there have been no ACK response at all.
  (b) A packet is received at the data stage for which the first data packet is DATAPID = DATA0.
  (c) At the status stage, an OUT or PING token is received.

(3) During control write no-data transfers
  (a) At the status stage, an OUT or PING token is received.

Note that in the control write transfer data stage, if the number of received data is more than the USB request wLength value, the control transfer sequence error cannot be recognized. Also, in the control read transfer status stage, when a packet other than a zero-length packet is received, an ACK response is returned and the transfer is successfully completed.

When a CTRT interrupt is generated due to a sequence error (SERR = 1), the value of CTSQ = 110b is retained until CTRT = 0 is written by the user system (clearing the interrupt state). Therefore, while CTSQ = 110 is retained, the CTRT interrupt for the completion of the setup stage is not generated, even when a new USB request is received. Events occurring after the setup stage are saved by the controller and the CTRT interrupt is generated after the interrupt state is cleared by software.



CTRT interrupt
(1) Setup stage end
(2) Control read transfer status stage transition
(3) Control write transfer status stage transition
(4) Control transfer complete
(5) Control transfer sequence error

Figure 24.8 Control Transfer Stage Transition

24.5 Pipe Control

Table 24.23 provides a list of pipe settings for the controller. In USB data transfers, data transmission is executed in logic pipes called endpoints. This controller comes with ten pipes for data transfer. Each pipe can be set to meet the requirements of the user system.

Table 24.23 Pipe Settings

Register Name	Bit Name	Setting	Remark
PIPECFG	TYPE	Specifies the transfer type	Can be set for pipes 1 to 9.
	BFRE	Selects BRDY interrupt mode	Can be set for pipes 1 to 5.
	DBLB	Selects double buffer configuration	Can be set for pipes 1 to 5.
	CNTMD	Selects continuous transfer or non-continuous transfer	Can be set for pipes 1 and 2 only in bulk transfers. Can be set for pipes 3 to 5.
	DIR	Selects transfer direction	Set to IN or OUT
	EPNUM	Endpoint number	Can be set for pipes 1 to 9. Set to a value other than 0000 when a pipe is in use.
	SHTNAK	Disables pipe when transfer is completed.	Can be set for pipes 1 and 2 only in bulk transfers. Can be set for pipes 3 to 5.
PIPEBUF	BUFSIZE	Buffer memory size	Cannot be set for DCP (fixed to 256 bytes). Up to 2 Kbytes can be set for pipes 1 to 5. Cannot be set for pipes 6 to 9 (fixed to 64 bytes).
	BUFNMB	Buffer memory number	Cannot be set for DCP (fixed at areas 0 to 3hex). Can be set for pipes 1 to 5 (specifiable in area ranging from 8 to 80hex). Cannot be set for pipes 6 to 9 (fixed at areas ranging from 4 to 7hex).
DCPMAXP PIPEMAXP	MXPS	Maximum packet size	Setting conforming to the USB standard
PIPEPERI	IFIS	Buffer flush	Can be set for pipes 1 and 2 only in isochronous transfers. Cannot be set for pipes 3 to 5. Cannot be set for pipes 6 to 9.
	IITV	Interval counter	Can be set for pipes 1 and 2 only in isochronous transfers. Cannot be set for pipes 3 to 5. Cannot be set for pipes 6 to 9.
DCPCTR PIPExCTR	BSTS	Buffer status	DCP state switched between receive and transmission buffer by ISEL bit
	INBUFM	IN buffer monitor	Available only for pipes 3 to 5.
	ATREPM	Auto response mode	Can be set for pipes 1 to 5.
	ACLRM	Auto buffer clear	Can be set for pipes 1 to 9.
	SQCLR	Sequence clear	Clears data toggle bit.
	SQSET	Sequence set	Sets data toggle bit.
	SQMON	Sequence check	Monitors data toggle bit.
	PBUSY	Pipe busy check	
	PID	Response PID	
PIPExTRE	TRENB	Transaction count enable	Can be set for pipes 1 to 5.
	TRCLR	Current transaction counter clear	Can be set for pipes 1 to 5.
PIPExTRN	TRNCNT	Transaction counter	Can be set for pipes 1 to 5.

24.5.1 Maximum Packet Size Setting

The MXPS bits in DCPMAXP and PIPEMAXP are used to specify the maximum packet size for each pipe. DCP and PIPE1 to PIPE5 can be set to any of the maximum packet sizes defined by the USB Specification. For PIPE6 to PIPE9, 64 bytes are the upper limit of the maximum packet size. The maximum packet size should be set before beginning the transfer (PID = BUF).
DCP: 64 should be set when using high-speed operation.
DCP: Select and set 8, 16, 32, or 64 when using full-speed operation.
Pipes 1 to 5: 512 should be set when using high-speed bulk transfer.
Pipes 1 to 5: Select and set 8, 16, 32, or 64 when using full-speed bulk transfer.
Pipes 1 to 2: Set a value between 1 and 1024 when using high-speed isochronous transfer.
Pipes 1 to 2: Set a value between 1 and 1023 when using full-speed isochronous transfer.
Pipes 6 to 9: Set a value between 1 and 64.

The high bandwidth transfers used with interrupt transfers and isochronous transfers are not supported.

24.5.2 Response PID

Set the response PID for each pipe with the PID bits of the DCPCTR and PIPExCTR registers.

(1) Response PID setting
The response PID specifies the response to a transaction from the Host.
(a) NAK setting: Always sends a NAK response when a transaction is issued.
(b) BUF setting: Responds to the transaction in accordance with the buffer memory state.
(c) STALL setting: Always sends a STALL response when a transaction is issued.
Regardless of the value set in the PID bit, an ACK is always sent as a response to a setup transaction and the USB request is stored in corresponding registers.

Based on the results of the transaction, the controller may trigger the PID bits to be written.
The controller will trigger a write event to the PID bit in the following cases.

- Hardware setting of response PID

(a) NAK setting:
1) When SETUP token is received normally (only DCP)
2) In bulk transfers when the SHTNAK bit of the PIPECFG register is set to 1 and short packet is received
3) In bulk transfers when SHTNAK bit is set to 1 and the transaction counter is completed.
(b) BUF setting: The BUF cannot be written by the controller.
(c) STALL setting:
1) When a maximum packet size over error is detected in the received data packet
2) When a control transfer sequence error is detected

24.5.3 Pipe Control Register Switching Procedures

The bits in the following pipe control registers can be re-written only when USB transmission is disabled (PID = NAK). Figure 24.9 shows the procedure for switching the pipe control register from the USB transmission enabled (PID = BUF) state.

Registers for which settings are prohibited when the USB transmission is enabled (PID = BUF):

- All bits of the DCPMAXP register
- Bits SQCLR and SQSET of the DCPCTR register
- All bits of the PIPECFG, PIPEBUF, PIPEMAXP and PIPEPERI registers
- ATREPM, ACLRM, SQCLR, and SQSET bits of the PIPExCTR register
- All bits of the PIPExTRE and PIPExTRN registers

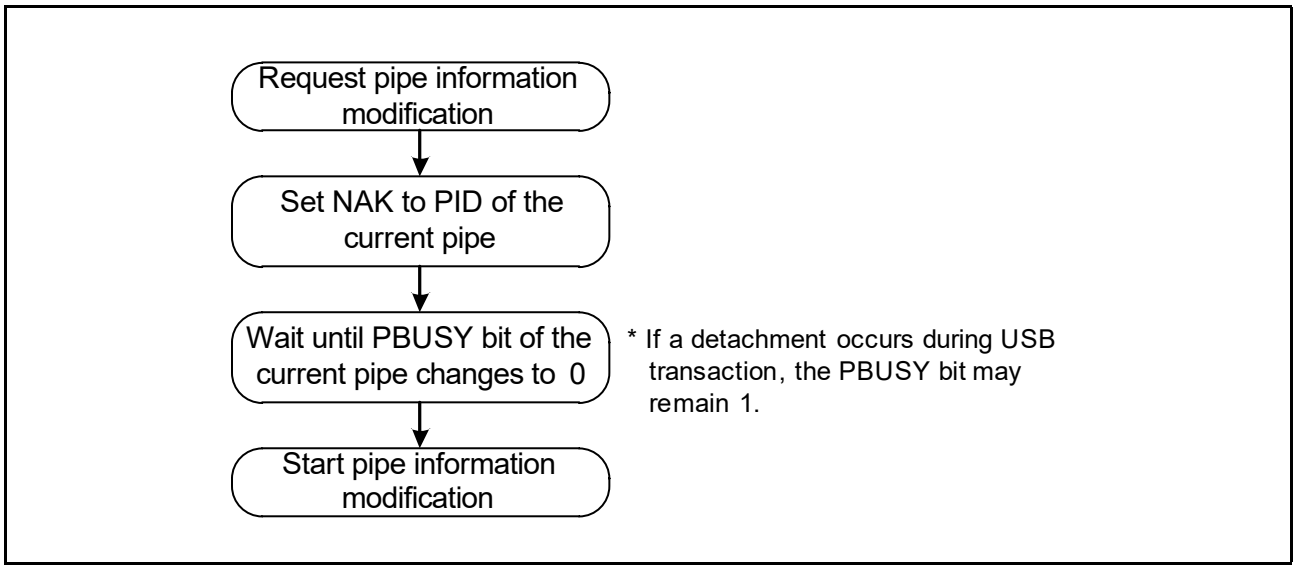


Figure 24.9 Procedure for Changing Pipe Information from USB Transmission Enabled (PID = BUF) State

In addition, the following bits of the pipe control registers can only be re-written with pipe information that is not set in the CURPIPE bits of CPU/DMA0/DMA1-FIFO ports.

Register for which settings are prohibited during setting of the CURPIPE bits in the FIFO port register:

- All bits of the DCPMAXP register
- All bits of the PIPECFG, PIPEBUF, PIPEMAXP and PIPEPERI registers
- Bit ACLRM of the PIPExCTR register

When modifying information of a pipe, specify other pipe number in the CURPIPE bits. Also, after setting the DCP pipe information, clear the buffer using the BCLR bit.

24.5.4 Data PID Sequence Bit

When a normal data transfer occurs in the control transfer data stage, bulk transfer or interrupt transfer, the controller automatically toggles the data PID sequence bit. The next data PID sequence bit for data transfer can be confirmed in the SQMON bit in the DCPCTR or PIPExCTR registers. The sequence bit is switched in the ACK handshake receive timing when data is sent or in the ACK handshake send timing when data is received. The data PID sequence bit can also be modified for the SQCLR and SQSET bits of the DCPCTR and PIPExCTR registers.
In control transfers, the controller automatically sets the sequence bit for stage transitions. DATA1 is established at the end of the setup stage. In the status stage, the controller does not reference the sequence bit and returns a response with PID = DATA1. Therefore, the bit does not need to be set with software. Note that the data PID sequence bit must be set with software when a ClearFeature request is received.
Finally, the sequence bit cannot be manipulated through the SQSET bit for the isochronous transfer setup pipe.

24.6 FIFO Buffers

This section describes the operation of the FIFO buffers in this controller.

24.6.1 FIFO Buffer Allocation

Figure 24.10 shows an example of memory map for the FIFO buffers of this controller. The FIFO buffer area is shared by the CPU controlling the user system and this controller. The access right of the FIFO buffers may be given to the user system (CPU side) or to this controller (SIE side).

An independent FIFO buffer area is allocated for each pipe. The memory area is made up of memory blocks of 64 bytes and defined by the starting block number (1 block is 64 bytes long) and the number of blocks (specified by the BUFNMB and BUFSIZE bits of the PIPEBUF register). If the CNTMD bit of the PIPExCFG register is set to continuous transfer mode, the value specified in the BUFSIZE bits must be an integral multiple of the maximum packet size. If the double buffer configuration is selected through the DBLB bit of the PIPExCFG register, two planes of memory area the size of which is specified by the BUFSIZE bits of the PIPEBUF register are allocated to a single pipe.

Two FIFO ports are used to access an FIFO buffer (data read/write). The pipe to be assigned to an FIFO port is designated by specifying the pipe number in the CURPIPE bits of the CFIFOSEL or DxFIFOSEL register.

The FIFO buffer state of each pipe can be confirmed by using the BSTS and INBUFM bits of the DCPCTR and PIPExCTR registers. The access right of a FIFO port can be confirmed by using the FRDY bit of the CFIFOCTR or DxFIFOCTR register.

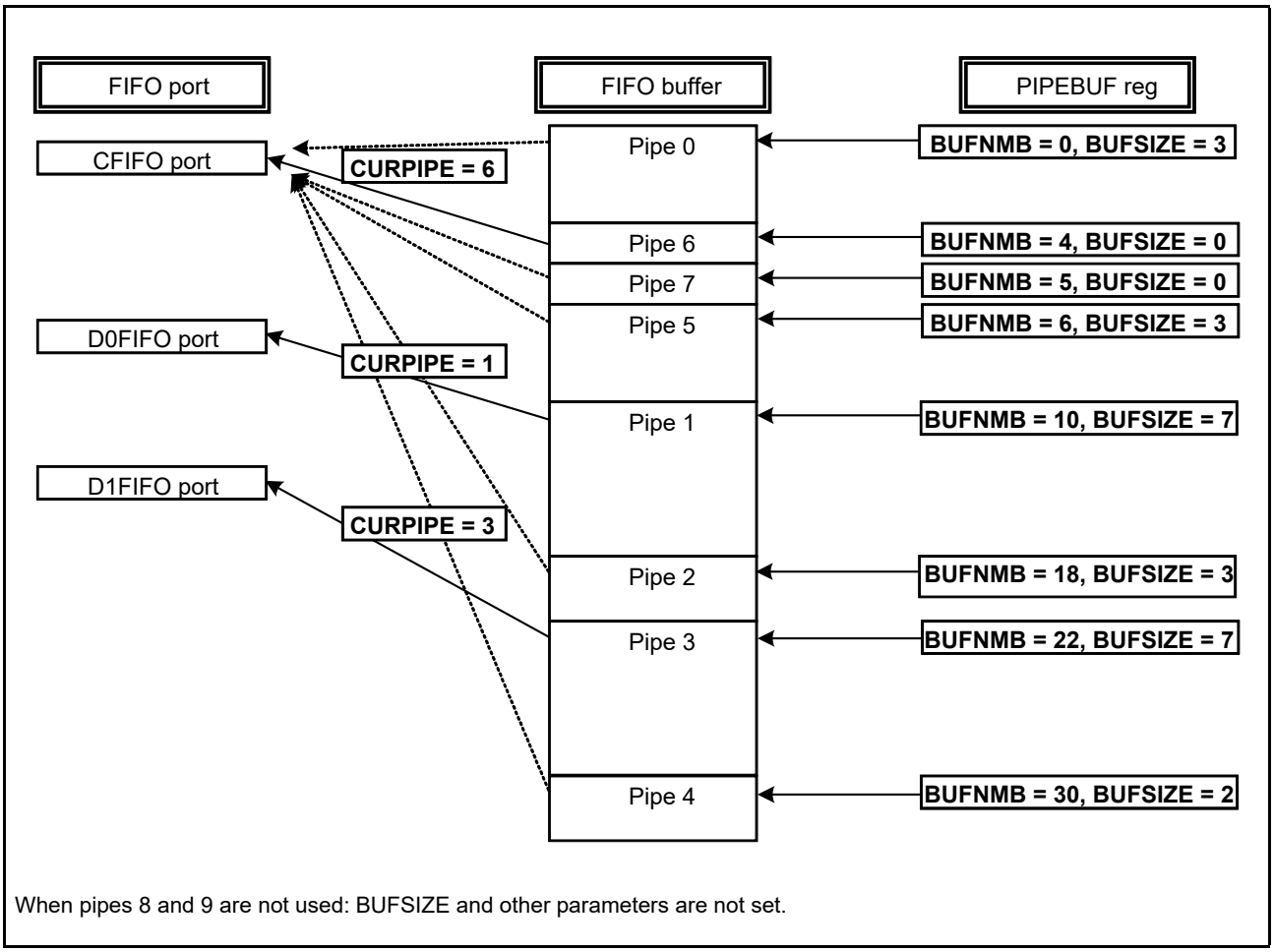


Figure 24.10 Example of FIFO Buffer Memory Map

24.6.2 Clearing FIFO Buffers

Table 24.24 shows a list of modes of clearing the FIFO buffers by this controller. The FIFO buffers can be cleared by the 3 bits that are listed in the table.

Table 24.24 List of FIFO Buffer Clearing Modes

Bit Name	BCLR	DCLRM	ACLRM
Register	CFIFOCTR register DxFIFOCTR register	DxFIFOSEL register	PIPExCTR register
Function	The FIFO buffer on the CPU side is cleared.	The FIFO buffer is automatically cleared after the data is read from the designated pipe.	The buffer is automatically cleared to discard all the received packets.
Clearing method	Write 1 to clear.	1: Mode valid 0: Mode invalid	1: Mode valid 0: Mode invalid

24.7 FIFO Port Function

This section describes FIFO port functions. Table 24.25 lists settings of FIFO port functions of this controller. If writing data is continued until the buffer becomes full (up to the maximum packet size in discontinuous transfer mode) during the data write access, the FIFO port automatically enters the state where data can be transmitted to the USB bus. To make data with a size less than the buffer full (the maximum packet size in discontinuous transfer mode) transmittable, write completion must be set by the BVAL bit in the CFIFOCTR or DxFIFOCTR register (DMA transfer: TEND signal of the DMAC (for details, see section 14, DMA Controller (DMACAa)). Furthermore, to transmit a zero-length packet, clear the buffer by the BCLR bit in the CFIFOCTR or DxFIFOCTR register and set write completion by the BVAL bit.
When all data has been read during the read access, the FIFO port automatically enters the state where a new packet can be received. However, when a zero-length packet is received (DTLN = 0), the data cannot be read. In that case, clear the buffer by the BCLR bit in the CFIFOCTR or DxFIFOCTR register. The receive data length is checked by the DTLN bits in the CFIFOCTR or DxFIFOCTR register.

Table 24.25 FIFO Port Function Settings

Register Name	Bit Name	Function	Notes
C/DxFIFOSEL	RCNT	DTLN read mode selection	
	REW	Buffer memory rewind (re-read, re-write)	
	DCLRM	Automatic clearing of buffer memory after specified pipe received data is read	DxFIFO only
	DREQE	DREQ signal assertion	DxFIFO only
	MBW	FIFO port access bit width	
	BIGEND	FIFO port endian selection	
	ISEL	FIFO port access direction	DCP only
	CURPIPE	Current pipe selection	
C/DxFIFOCTR	BVAL	Buffer memory write completion	
	BCLR	Clearing of CPU-side buffer memory	
	FRDY	Monitoring of FIFO port ready	
	DTLN	Confirmation of received data length	

24.7.1 FIFO Port Selection

Table 24.26 shows the list of pipes that can be selected in each FIFO port. The pipes to be accessed are selected with the CURPIPE bits of the CFIFOSEL or DxFIFOSEL register. After selecting the pipes, confirm that the value of the CURPIPE bits written was read correctly (if the previous pipe number is read out, this indicates the controller is still changing the pipe), then confirm that FRDY = 1 and access the FIFO port. Figure 24.11 shows the procedure for switching pipes for access to the FIFO port.
Also, select the bus width for the FIFO port access with the MBW bit. The buffer memory access direction is determined by the ISEL bit for DCP, and the DIR bit of the PIPExCFG register for all other pipes.

Table 24.26 FIFO Port Access by Pipe

Pipe	Access Method	Usable Ports
DCP	CPU access	CFIFO port register
Pipes 1 to 9	CPU access	CFIFO port register
	DMA access	DxFIFO port register

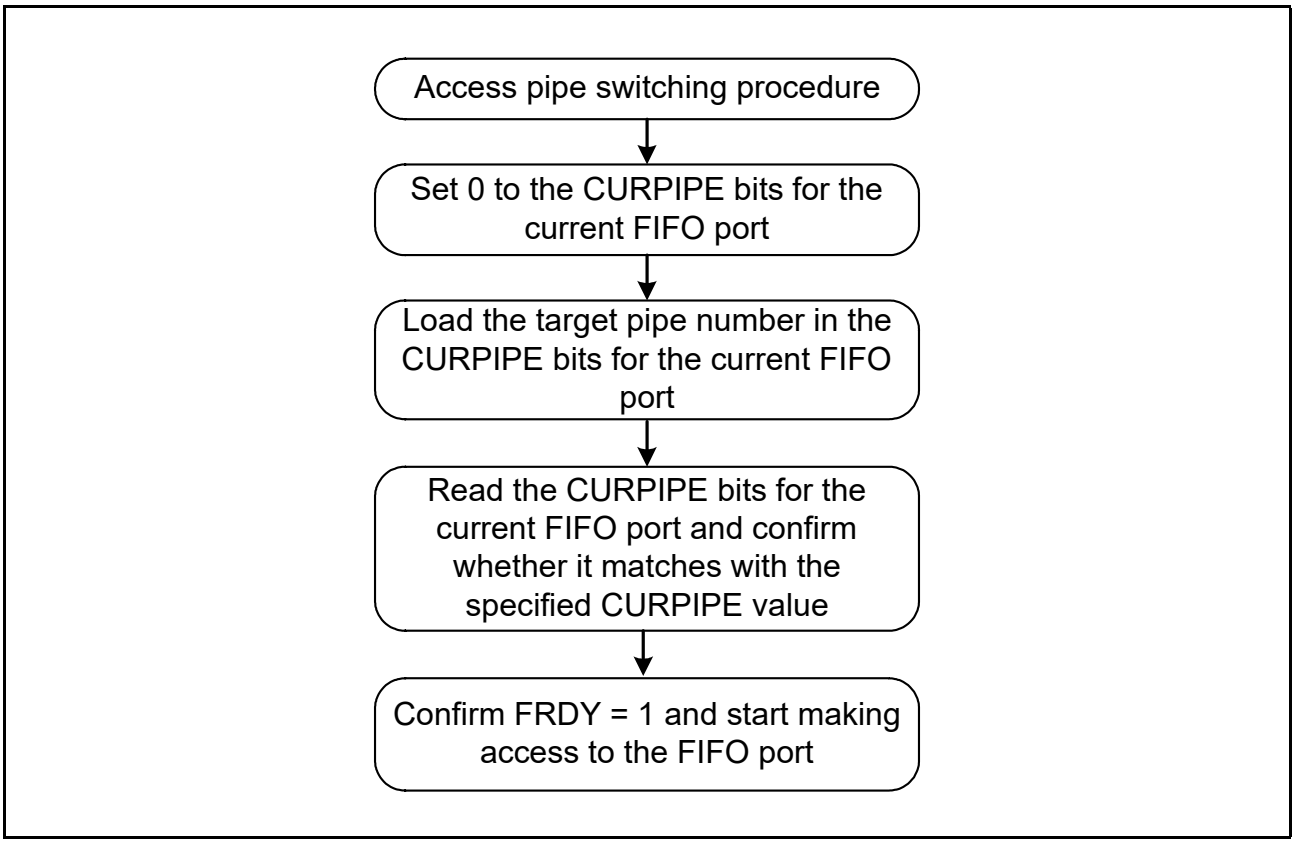


Figure 24.11 Pipe Switching Procedure for FIFO Port Access

24.7.2 DxFIFO Automatic Clear Mode (DxFIFO Port Read Direction)

When a data read event of the controller buffer memory is completed with setting the DCLRM bit of the DxFIFOSEL register to 1, the buffer memory of the corresponding pipe is automatically cleared.

Table 24.27 shows the correspondence between packet reception and buffer memory clearing by software in each setting.
As indicated in Table 24.27, the buffer clear conditions differ according to the set value of the BFRE bit, even for states in which clear is normally required, using the DCLRM bit eliminates the need for clearing of the buffer by software, enabling DMA transfers without the use of software.
Note that this function only supports the buffer memory read direction setting.

Table 24.27 Correspondence of Packet Reception and Buffer Memory Clearing by Software

Register Setting / Buffer State when Packet is Received	DCLRM = 0		DCLRM = 1	
	BFRE = 0	BFRE = 1	BFRE = 0	BFRE = 1
Buffer full	Clearing not required	Clearing not required	Clearing not required	Clearing not required
Zero-length packet received	Clearing required	Clearing required	Clearing not required	Clearing not required
Normal short packet received	Clearing not required	Clearing required	Clearing not required	Clearing not required
Transaction count end	Clearing not required	Clearing required	Clearing not required	Clearing not required

24.7.3 BRDY Interrupt Timing Selection

The BFRE bit of the PIPECFG register can be set so that the BRDY interrupt is not generated when a data packet of maximum packet size is received.

When using a DMA transfer, this function enables an interrupt to be generated only when the last data is received. The last data indicates either a short packet reception or the transaction count end. By setting BFRE = 1, the BRDY interrupt will be generated after the received data is read. By reading the DTLN bit of the DnFIFOCTR register, the receive data length of last data packet received just before the BRDY interrupt was generated can be confirmed.

Table 24.28 shows the timing of the BRDY interrupt.

Table 24.28 BRDY Interrupt Generation Timing

Register Setting / Buffer State when Packet is Received	BFRE = 0	BFRE = 1
Buffer full (normal packet received)	When packet is received	No interrupt generated
Zero-length packet received	When packet is received	When packet is received
Normal short packet received	When packet is received	When read event of data received from buffer memory is completed
Transaction count end	When packet is received	When read event of data received from buffer memory is completed

The BFRE bit function is only valid in reading direction of the buffer memory. When in writing direction, fix the BFRE bit to 0.

24.8 Control Transfer (DCP)

Data transfers of the data stage of control transfers are done using the default control pipe (DCP). The DCP buffer memory is a 64-byte single buffer, and is a fixed area that is shared for both control reading and control writing. The buffer memory can be accessed only through the CFIFO port.

24.8.1 Setup Stage

The controller always responds with an ACK when it receives a normal setup packet. The controller operations in the setup stage are as follows.

(1) When a new setup packet is received, the controller sets the following bits.
  (a) Sets the VALID bit of the INTSTS0 register to 1.
  (b) Sets the PID bits of the DCPCTR register to NAK.
  (c) Sets the CCPL bit of the DCPCTR register to 0.
(2) When a data packet is received following the setup packet, the USB request parameters are stored in the following registers: USBREQ, USBVAL, USBINDX and USBLENG.

Always set VALID = 0 before the response process to a control transfer. While VALID = 1, PID = BUF will not be set and the data stage cannot be completed.
The function of the VALID bit allows the controller to temporarily stop a request in-process when it receives a new USB request during a control transfer, and respond to the newest request.

In addition, the controller automatically judges the direction bit (bmRequestType bit 8) and the request data length (wLength) of the received USB request and determines whether it is a control read transfer, control write transfer or control write no-data transfer, and then handles the stage transition. If the sequence is incorrect, a sequence error for the control transfer stage transition interrupt is generated and is notified to the software. For more information concerning the controller stage management, see Figure 24.8.

24.8.2 Data Stage

Use the DCP for data transfers in response to receiving a USB request. Before accessing the DCP buffer memory, set the access direction in the ISEL bit of the CFIFOSEL register.
The transaction is executed by setting the PID bits of the DCPCTR register to BUF.
Data transfer completion is detected by the BRDY and BEMP interrupts. Use the BRDY interrupt for control write transfers and the BEMP interrupt for control read transfers.
For control write transfers in high-speed operation, a NYET handshake is sent in accordance with the buffer memory state.

24.8.3 Status Stage

When the setting of the PID bits of the DCPCTR register is BUF, set the CCPL bit to 1 to complete the control transfer. After the above settings, the controller automatically executes the status stage in accordance with the data transfer direction fixed in the setup stage. The detailed process is as follows.

(1) Control read transfers:
  The controller receives a zero-length packet from the USB host and sends an ACK response.
(2) Control write transfers and no-data control transfers:
  The controller sends a zero-length packet and receives an ACK response from the USB host controller.

24.8.4 Control Transfer Automatic Response

The controller automatically sends a response to a normal SET_ADDRESS request. If one of the following errors occurs, a response must be sent by software.

(1) bmRequestType ≠ 00h
(2) wIndex ≠ 00h
(3) wLength ≠ 00h
(4) wValue > 7Fh
(5) DVSQ = 011b (configured)

All requests other than the SET_ADDRESS request must be responded to by software.

24.9 Bulk Transfer (Pipes 1 to 5)

The user can select the buffer memory usage method (single/double buffer, continuous/non-continuous transfer mode) for bulk transfer. The maximum size that can be set for the buffer memory is 2 Kbytes. The controller manages the buffer memory state and automatically responds to PING packets and NYET handshakes.

24.9.1 NYET Handshake Control

Table 24.29 shows the list of responses to a token received in a bulk or control transfer. When an OUT token is received in a bulk or control transfer and there is only enough open space for one packet in the buffer memory, the controller sends a NYET response. However, when a short packet is received, the controller sends an ACK response instead of a NYET response, even under these conditions.

Table 24.29 List of Responses to Received Tokens

PID Bits Setting	Buffer Memory State	Received Token	Response	Notes
NAK/STALL	—	SETUP	ACK	—
	—	IN/OUT/PING	NAK/STALL	—
BUF	—	SETUP	ACK	—
	RCV-BRDY*1	OUT/PING	ACK	When OUT token is received, data packet is received.
	RCV-BRDY*2	OUT	NYET	Data packet is received.
	RCV-BRDY*2	OUT (Short)	ACK	Data packet is received.
	RCV-BRDY*2	PING	ACK	
	RCV-NRDY*3	OUT / PING	NAK	
	TRN-BRDY*4	IN	DATA0 / 1	Data packet is transmitted.
	TRN-NRDY*5	IN	NAK	

Note 1. RCV-BRDY: Buffer memory has enough space for 2 packets or more when an OUT or PING token is received.
Note 2. RCV-BRDY: Buffer memory has only enough space for one packet when an OUT token is received.
Note 3. RCV-NRDY: Buffer memory has not enough space when a PING token is received.
Note 4. TRN-BRDY: Buffer memory has data for transmission when an IN token is received.
Note 5. TRN-NRDY: Buffer memory does not have data for transmission when an IN token is received.

24.10 Interrupt Transfer (Pipes 6 to 9)

This controller executes an interrupt transfer in accordance with the period managed by the host controller. The controller ignores (no response) PING packets in interrupt transfers. In addition, the controller does not send a NYET handshake, but responds with ACK, NAK, or STALL.
The controller does not support high-bandwidth interrupt transfers.

24.11 Isochronous Transfer (Pipes 1 and 2)

The controller provides the following functions for isochronous transfers.

(1) Isochronous transfer error information notification
(2) Interval counter (IITV bit setting)
(3) Isochronous IN transfer data setup control (IDLY function)
(4) Isochronous IN transfer buffer flush function (IFIS bit setting)
(5) SOF pulse output function

The controller does not support high-bandwidth isochronous transfers.

24.11.1 Isochronous Transfer Error Detection

The controller manages isochronous transfer errors by software and therefore has the following error information detection functions. Table 24.30 and Table 24.31 describe the procedure in which errors are confirmed and the interrupts that are generated.

(1) PID error
When PID of the received packet is corrupted
(2) CRC error and bit stuffing error
When an error occurs in CRC of the received packet or when the bit stuffing is corrupted
(3) Maximum packet size over
This indicates the data size of the received packet is larger than the value set for the maximum packet size.
(4) Overrun and underrun
When there is no data in the buffer memory when an IN token is received in an IN-direction (send) transfer
When there is no empty space in the buffer memory when an OUT token is received in an OUT-direction (receive) transfer
(5) Interval error
The following will generate interval errors.
(a) When an IN token could not be received in the interval frame of an isochronous IN transfer
(b) When an OUT token could not be received in the interval frame of an isochronous OUT transfer

Table 24.30 Errors Detected in Token Reception/Transmission

Detection Priority	Error Type	Generated Interrupts and States at Time of Error Detection
1	PID error	No interrupt generated (ignored as corrupted packet)
2	CRC error or bit stuffing error	No interrupt generated (ignored as corrupted packet)
3	Overrun or underrun errors	NRDY interrupt is generated and OVRN bit is set. A zero-length packet is sent in response to an IN token. A data packet is not received in response to an OUT token.
4	Interval error	NRDY interrupt generated

Table 24.31 Errors Detected in Data Packet Reception

Detection Priority	Error Type	Generated Interrupts and States
1	PID error	No interrupt generated (ignored as corrupted packet)
2	CRC error or bit stuffing error	NRDY interrupt is generated and CRCE bit is set.
3	Maximum packet size over error	BEMP interrupt is generated and PID is set to STALL.

24.11.2 DATA-PID

This controller does not support high-bandwidth transfers. The following occurs in response to a received PID.

(1) IN direction:
  (a) DATA0: Transmitted as PID of the data packet
  (b) DATA1: Not transmitted
  (c) DATA2: Not transmitted
  (d) mDATA: Not transmitted

(2) OUT direction (in full-speed operation):
  (a) DATA0: Received successfully as PID of the data packet
  (b) DATA1: Received successfully as PID of the data packet
  (c) DATA2: The packet is ignored.
  (d) mDATA: The packet is ignored.

(3) OUT direction (in high-speed operation):
  (a) DATA0: Received successfully as PID of the data packet
  (b) DATA1: Received successfully as PID of the data packet
  (c) DATA2: Received successfully as PID of the data packet
  (d) mDATA: Received successfully as PID of the data packet

24.11.3 Interval Counter

24.11.3.1 Overview of Operation

The isochronous transfer interval can be set in the IITV bit of the PIPEPERI register. Table 24.32 shows the functions of the interval counter.

Table 24.32 Interval Counter Functions

Transfer Direction	Function	Detection Conditions
IN	Transfer buffer flush function	Cannot successfully receive IN token in interval frame during isochronous IN transfer.
OUT	Notifies that a token not being received	Cannot successfully receive OUT token in interval frame during isochronous OUT transfer.

Since the interval counting is performed upon reception of SOF or by the complemented SOF, the isochronism can be maintained even if the SOF is damaged. Frame intervals are set as 2^{IITV} (μ) frames.

24.11.3.2 Interval Counter Initialization

The controller initializes the interval counter under the following conditions.

(1) Hardware reset
Initializes the IITV bit.

(2) Clearing of the buffer memory by the ACLRM bit
This initializes the counter but not the IITV bit.

(3) USB bus reset

After the interval counter is initialized and a packet is successfully transferred, the interval count starts under the following conditions.

(1) SOF is received after data is sent in response to an IN token when PID = BUF.

(2) SOF is received after data is received in response to an OUT token when PID = BUF.

Note that the interval counter is not initialized in the following conditions.

(1) When the PID is set to NAK or STALL
The interval timer is not stopped at this time. The transaction will be attempted at the next interval.

(2) USB bus reset or USB suspend
The IITV bit is not initialized at this time. When the SOF is received, the count starts from the value before the reception.

24.11.4 Isochronous Transfer Transmission Data Setup

In the isochronous data transmission by this controller, after data is written to the buffer memory, the data packet can be sent out in the next frame after the SOF packet is detected. This function, called the isochronous transfer transmission data setup, allows specification of the frame that started transmission.
When the buffer memory is used for double buffering and writing to both buffers has been completed, only transfer from the first buffer to have received data can proceed. Therefore, even when several IN tokens are received in the same frame, only one packet of data is sent by the buffer memory.

When an IN token is received, if the buffer memory is ready for transmission, the data is transferred and a normal response is returned. However, if the buffer memory is not ready for transmission, a zero-length packet is sent and an underrun error occurs.

Figure 24.12 shows an example of transmission using the isochronous transfer transmission data setup function with this controller when IITV = 0 (for each frame) is set.

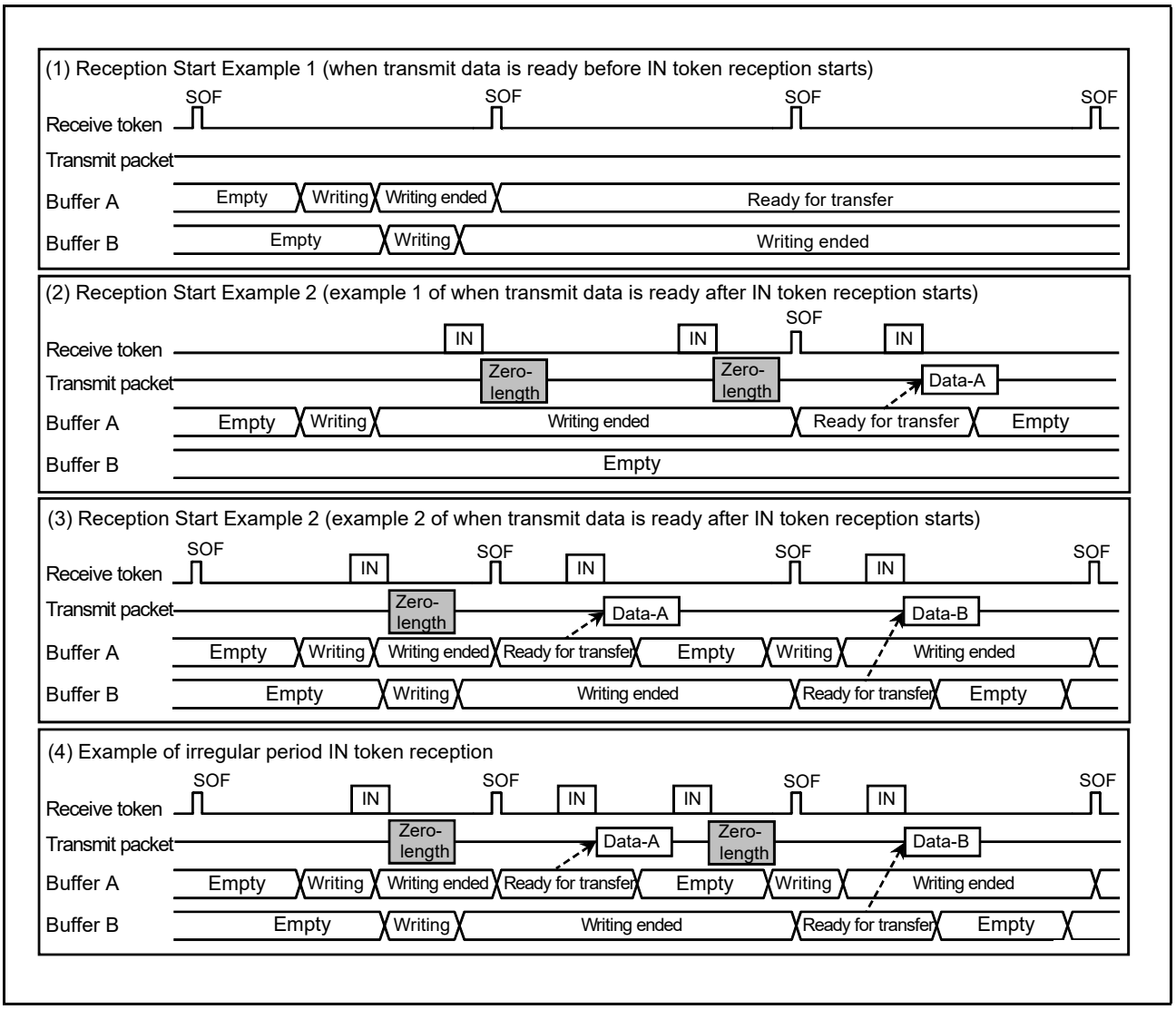


Figure 24.12 Example of Data Setup Function Operation

24.11.5 Isochronous Transfer Transmission Buffer Flush

When an SOF packet or a µSOF packet of the next frame is received without receiving an IN token in the interval frame during isochronous data transmission, this controller operates as if a corrupted IN token was received, and clears the buffer which is ready for transmission, making that buffer ready for writing.
If a double buffer is being used and writing to both buffers has been completed, data are considered to have been sent from the buffer memory that was cleared in the same interval frame, and transmission is enabled for the buffer memory that is not discarded with SOF or µSOF packets reception.
The timing at which the buffer flush function is activated varies depending on the setting of the IITV bits.

(1) When IITV = 0
The buffer flush operation proceeds from the first frame after the pipe becomes valid.

(2) When IITV ≠ 0
The buffer flush operation proceeds after the first successful transaction.

Figure 24.13 shows an operation example of the buffer flush function of this controller. When an unanticipated token is received prior to the interval frame, this controller sends the written data or a zero-length packet as an underrun error according to the data setup state.

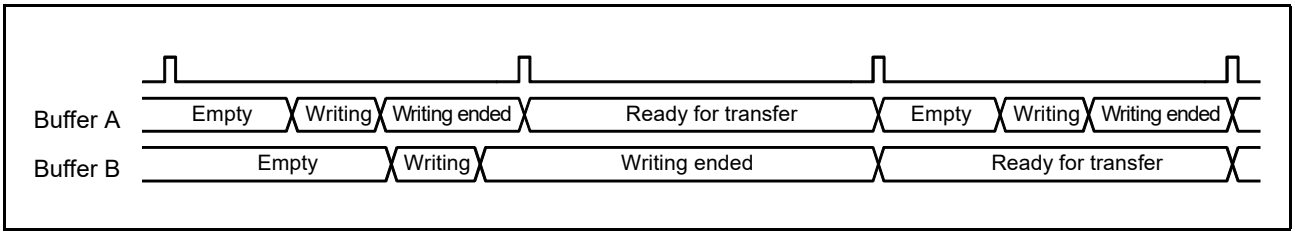


Figure 24.13 Buffer Flush Function Operation Example

Figure 24.14 shows an example of an interval error generated in the controller. There are five types of interval errors, as listed below. Timing 1 in the figure shows when the interval error occurs and how the buffer flush function operates.
When an interval error occurs during an IN transfer, the buffer flush function goes into operation; during an OUT transfer, the NRDY interrupt is generated.
Use the OVRN bit to determine whether an error is an NRDY interrupt, such as a receive packet error, or an overrun error.
Responses to the tokens in the shaded boxes are executed in accordance to the buffer memory state.

(1) IN direction:
  (a) If the buffer is ready for transfer, data is transferred as a normal response.
  (b) If the buffer is not ready for transfer, a zero-length packet is sent and an underrun error occurs.

(2) OUT direction:
  (a) If the buffer is ready for reception, data is received as a normal response.
  (b) If the buffer is not ready for reception, data is discarded and an overrun error occurs.

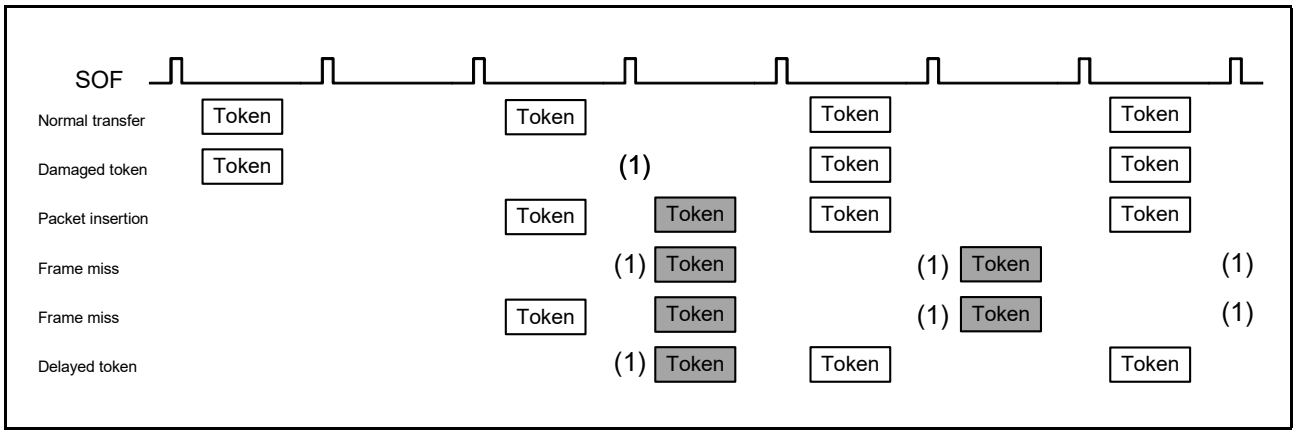


Figure 24.14 Example of Interval Error When IITV = 1

24.12 SOF Interpolation

If an SOF packet could not be received at intervals of 1 ms (in full-speed operation) or 125 µs (in high-speed operation) because of corruption or missing, this controller interpolates the SOF. The SOF interpolation operation begins when both the USBE and SUSPM bits have been set to 1 and an SOF packet is received. The interpolation is initialized under the following conditions.

(1) Hardware reset
(2) USB bus reset
(3) Suspended state detected

The SOF interpolation operates according to the following specifications.

(1) The frame interval (125 µs or 1 ms) is based on the results of the reset handshake protocol.
(2) The interpolation does not operate until the SOF packet is received.
(3) After a first SOF packet is received, interpolation of another SOF packet proceeds after a 125-µs or 1-ms interval counted in cycles of the internal clock running at 48 MHz clock has elapsed.
(4) Interpolation is performed in the previous reception intervals after the 2nd and subsequent SOF packets are received.
(5) Interpolation is not performed in the suspended state or during reception of a USB bus reset.
When the controller goes to the suspended state in high-speed operation, interpolation continues for 3 ms from the last packet.

The SOF interpolation works for the following.

(1) Updating of the frame number or micro-frame number
(2) SOFR interrupt and µSOF lock
(3) SOF pulse output
(4) Isochronous transfer interval count

When an SOF packet is lost during full-speed operation, the FRNM bit of the FRMNUM register is not updated.
When a µSOF packet is lost during high-speed operation, the UFRNM bit of the UFRMNUM register is updated. However, when a µSOF packet when µFRNM = 000b is lost, the FRNM bit is not updated. At this time, even if µSOF packets when µFRNM ≠ 000b are received successfully, the FRNM bit is not updated.

25. Serial Communications Interface with FIFO (SCIFA)

This LSI has five channels of serial communication interface (SCIFA) with FIFO that support both asynchronous and clock synchronous serial communication. The SCIFA has 16-stage FIFO buffers for transmission and reception, respectively, for each channel that enable this LSI to perform efficient high-speed continuous communication.

25.1 Overview

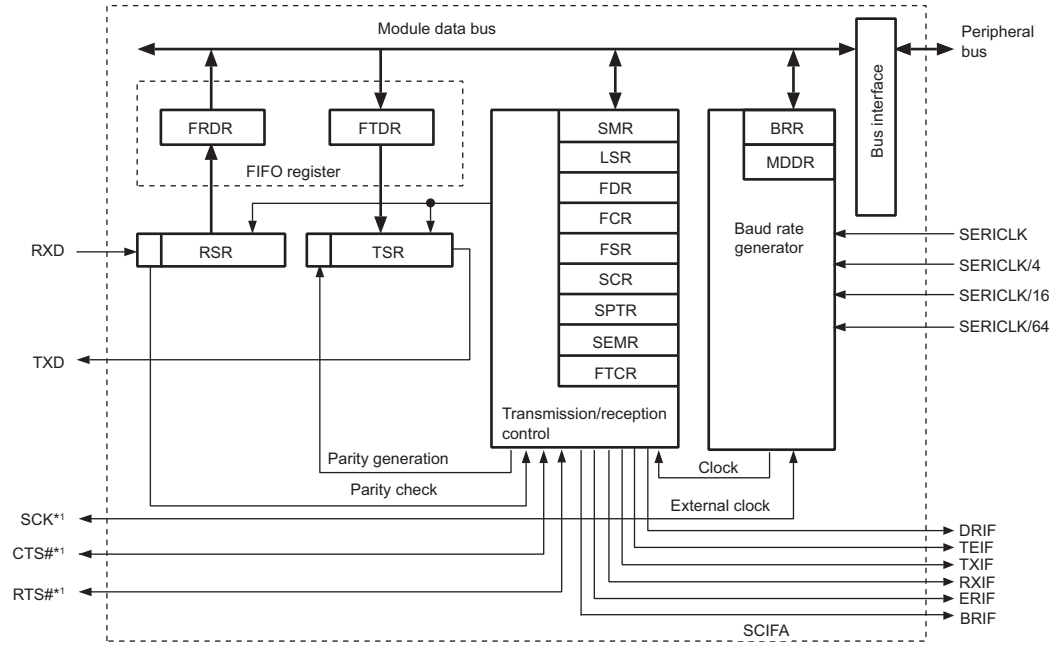
Table 25.1 lists the specifications of the SCIFA.

Table 25.1 Specifications of SCIFA

Item		Description
Channel		5 channels
Serial communication method		Asynchronous communication mode and clock synchronous communication mode*1
Transfer speed		Selectable bit rate with an on-chip baud rate generator
Full duplex communication		Transmitting section: realizes continuous data transmission using 16-stage FIFO buffer Receiving section: realizes continuous data reception using 16-stage FIFO buffer
Data transmission		Selectable either LSB-first or MSB-first transfer
Interrupt source		The following six sources: • Transmit-end (TEIF) • Transmit-FIFO-data-empty (TXIF) • Receive-FIFO-data-full (RXIF) • Receive-data-ready (DRIF)*2 • Framing error or parity error (ERIF) • Break or overrun (BRIF)
Asynchronous communication mode	Character length	7 or 8 bits
	Transmission stop bit length	1 or 2 bits
	Parity	Even, odd, or none
	Receive error detection	Detects following errors as receive error: parity error, overrun error, and framing error
	Hardware flow control	Controls data transmission and reception using the CTS# and RTS# pins.*1
	Break detection	Break signal detection function by hardware.
	Clock source	Selectable from internal or external clock
	Noise cancellation	Incorporates a digital noise filter in the RXD pin input path.
Clock synchronous communication mode*1	Character length	8 bits
	Receive error detection	Detects an overrun error as a receive error.
	Clock source	Selectable either internal or external clock
Bit rate modulation		Enables errors to be decreased by correcting the output of the on-chip baud rate generator.

Note 1. Setting is allowed for channels 0 to 2 only.
Note 2. Effective only for asynchronous communication mode.

Figure 25.1 shows a block diagram of the SCIFA.



RSR: Receive shift register
FRDR: Receive FIFO data register
TSR: Transmit shift register
FTDR: Transmit FIFO data register
SMR: Serial mode register
SCR: Serial control register
FSR: Serial status register
MDDR: Modulation duty register
BRR: Bit rate register
SPTR: Serial port register
FCR: FIFO control register
FDR: FIFO data count register
LSR: Line status register
SEMR: Serial extended mode register
FTCR: FIFO trigger control register

Note 1. Setting is allowed for channels 0 to 2 only.

Figure 25.1 Block Diagram of SCIFA

Table 25.2 lists the input/output pins of the SCIFA.

Table 25.2 Pin Configuration of the SCIFA

Item	Pin Name	I/O	Function
Serial clock pin	SCK*1	I/O	Transmission/reception clock input/output, general output
Receive data pin	RXD	Input	Receive data input
Transmit data pin	TXD	Output	Transmit data output
Transmission/reception start control pin	CTS#*1	I/O	Input for hardware flow control (transmission enable signal) / general output
	RTS#*1	Output	Output for hardware flow control (transmission request signal) / general output

Note: Channels of each pin is omitted.
Note 1. Setting is allowed for channels 0 to 2 only.

25.2 Register Descriptions

25.2.1 Receive Shift Register (RSR)

The RSR register receives serial data and temporally stores the data. The SCIFA stores the serial data input via the RXD pin into the RSR register. When one byte of data has been received, it is automatically transferred to the receive FIFO data register (FRDR).
The CPU cannot read from or write to the RSR register directly.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

25.2.2 Receive FIFO Data Register (FRDR)

The FRDR register is an 8-bit, 16-stage FIFO register that stores the received serial data. When the SCIFA receives one byte of serial data, it transfers the received data from the receive shift register (RSR) to the FRDR register and completes the receive operation. Continuous reception is possible until the received 16 bytes of data are stored. If the FRDR register is read when there is no received data in the FRDR register, an undefined value is read.
When the FRDR register is full of received data, subsequently received serial data is lost.
The CPU can read the FRDR register but cannot write to it.

Address(es): SCIFA0.FRDR A006 500Ah, SCIFA1.FRDR A006 540Ah, SCIFA2.FRDR A006 580Ah, SCIFA3.FRDR A006 5C0Ah, SCIFA4.FRDR A006 600Ah

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

25.2.3 Transmit Shift Register (TSR)

The SCIFA transfers the transmit data from the transmit FIFO data register (FTDR) to the TSR register, and then transmits the data serially to the TXD pin. After transmitting one byte of data, the SCIFA automatically transfers the next transmit data from the FTDR register into the TSR register and starts transmission.
The CPU cannot read from or write to the TSR register directly.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

25.2.4 Transmit FIFO Data Register (FTDR)

The FTDR register is an 8-bit, 16-stage FIFO register that stores serial transmission data. When the SCIFA detects that the transmit shift register (TSR) is empty, it transmits data written in the FTDR register to the TSR register and starts serial transmission. Continuous serial transmission is executed until there is no transmit data left in the FTDR register. Writing the transmit data to the FTDR register should be done when a transmit data empty interrupt (TXIF) request is generated.

When the FTDR register becomes full of transmit data (16 bytes), no more data can be written. Even if new data is written, the data is ignored.

CPU can read from the FTDR register but cannot write to it.

Address(es): SCIFA0.FTDR A006 5006h, SCIFA1.FTDR A006 5406h, SCIFA2.FTDR A006 5806h, SCIFA3.FTDR A006 5C06h, SCIFA4.FTDR A006 6006h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

25.2.5 Serial Mode Register (SMR)

The SMR register specifies the SCIFA serial communication format and selects the clock source for the baud rate generator.

The CPU can always read and write to the SMR register.

Address(es): SCIFA0.SMR A006 5000h, SCIFA1.SMR A006 5400h, SCIFA2.SMR A006 5800h, SCIFA3.SMR A006 5C00h, SCIFA4.SMR A006 6000h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CM	CHR	PE	PM	STOP	—	CKS[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1b0 0 0: 1 × SERICLK*1 0 1: 1/4 × SERICLK*1 1 0: 1/16 × SERICLK*1 1 1: 1/64 × SERICLK*1	R/W
b2	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b3	STOP	Stop Bit Length	0: One stop bit 1: Two stop bits	R/W
b4	PM	Parity Mode	0: Even parity 1: Odd parity	R/W
b5	PE	Parity Enable	0: Parity bit addition or check is disabled. 1: Parity bit addition or check is enabled.	R/W
b6	CHR	Character Length	0: 8-bit data 1: 7-bit data*2	R/W
b7	CM	Communication Mode	0: Asynchronous mode 1: Clock synchronous mode (This setting is allowed for channels 0 to 2 only. Setting this value to chennels 3 and 4 are not allowed.)	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. SERICLK: Peripheral clock
Note 2. When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted.

CKS[1:0] Bits (Clock Select)

Select an internal clock source for the on-chip baud rate generator. For further information on the clock source, bit rate register settings, and baud rates, see section 25.2.8, Bit Rate Register (BRR).

STOP Bit (Stop Bit Length)

Selects one bit or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added. When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character.

Note: When transmitting with one stop bit, a single 1 bit (stop bit) is added at the end of each transmission character.

Note: When transmitting with two stop bits, two 1 bits (stop bits) are added at the end of each transmission character.

PM Bit (Parity Mode)

Selects either the even or odd parity check. The setting of this bit is effective only when the parity enable (PE) bit of this register is set to 1 in asynchronous mode. The setting of this bit is ignored in clock synchronous mode, or when parity addition/check is disabled in asynchronous mode.

Note: If even parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s even in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is even.

Note: If odd parity is selected, the parity bit is added to data to be transmitted to make the total number of 1s odd in the transmission character and parity bit combined. When receiving, the SCIFA verifies that the total number of 1s in the received character and parity bit combined is odd.

PE Bit (Parity Enable)

Selects whether to add a parity bit on data transmission and whether to enable/disable the parity check on data reception in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the setting of this bit.

Note: When this bit is set to 1, an even or odd parity bit specified in the PM bit is added to data to be transmitted. The SCIFA verifies whether the parity bit of the received data is even or odd as specified in the PM bit when receiving.

CHR Bit (Character Length)

Selects 7- or 8-bit data length in asynchronous mode. In clock synchronous mode, the data length is always 8 bits, regardless of the CHR setting.

CM Bit (Communication Mode)

Selects whether the SCIFA operates in asynchronous or clock synchronous mode. Setting as clock synchronous mode is only allowed for channels 0 to 2.

25.2.6 Serial Control Register (SCR)

The SCR register enables or disables the SCIFA transmission/reception and interrupt requests, and selects the transmit/receive clock source. The CPU can always read from and write to the SCR register.

Address(es): SCIFA0.SCR A006 5004h, SCIFA1.SCR A006 5404h, SCIFA2.SCR A006 5804h, SCIFA3.SCR A006 5C04h, SCIFA4.SCR A006 6004h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	TIE	RIE	TE	RE	REIE	TEIE	CKE[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	In asynchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for input pin (input signal is ignored). The SCK pin state depends on the SCKIO and SCKDT bits in SPTR. 0 1: Internal clock or SCK pin is used for clock output (The output clock frequency is 16 or 8 times of the bit rate). 1 0: External clock or SCK pin is used for clock input (The input clock frequency is 16 or 8 times of the bit rate). 1 1: Setting prohibited In clock synchronous mode: b1 b0 0 0: Internal clock or SCK pin is used for synchronous clock output. 0 1: Internal clock or SCK pin is used for synchronous clock output. 1 0: External clock or SCK pin is used for synchronous clock input. 1 1: Setting prohibited	R/W
b2	TEIE*1	Transmit End Interrupt Enable	0: Transmit end interrupt (TEIF) request is disabled. 1: Transmit end interrupt (TEIF) request is enabled.	R/W
b3	REIE	Receive Error Interrupt Enable	0: Receive-error interrupt (ERIF) and break interrupt (BRIF) requests are disabled. 1: Receive-error interrupt (ERIF) and break interrupt (BRIF) requests are enabled.	R/W
b4	RE	Receive Enable	0: Data reception is disabled. 1: Data reception is enabled.	R/W
b5	TE	Transmit Enable	0: Data transmission is disabled. 1: Data transmission is enabled.	R/W
b6	RIE	Receive Interrupt Enable	0: Receive-FIFO-data-full interrupt (RXIF), receive-data ready interrupt (DRIF), receive-error interrupt (ERIF), and break interrupt (BRIF) requests are disabled. 1: Receive-FIFO-data-full interrupt (RXIF), receive-data ready interrupt (DRIF), receive-error interrupt (ERIF), and break interrupt (BRIF) requests are enabled.	R/W
b7	TIE	Transmit Interrupt Enable	0: Transmit-FIFO-data-empty interrupt request (TXIF) is disabled. 1: Transmit-FIFO-data-empty interrupt request (TXIF) is enabled.	R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. TEIF interrupt requests can be cleared by reading 1 from the TEND flag, and then clearing the setting to 0, or by setting the TEIE bit to 0.

CKE[1:0] Bits (Clock Enable)

Select the SCIFA clock source and enable or disable clock output from the SCK pin. Depending on the settings of these bits, the SCK pin can be used for serial clock output or serial clock input. If the SCK pin is set for the synchronous clock output in the clock synchronous mode, set the CM bit in the SMR register to 1, and then set the CKE[1:0] bits. The settings of the CKE[1:0] bits are listed in Table 25.15.

REIE Bit (Receive Error Interrupt Enable)

Specifies whether to enable or disable a receive-error interrupt (ERIF) request and a break interrupt (BRIF) request. The setting of this bit is only valid when the RIE bit is set to 0.

Note: ERIF interrupt requests can be cleared by reading 1 from the ER bit in the FSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0. BRIF interrupt requests can be cleared by reading 1 from the BRK bit in the FSR register, or from the ORER flag in the LSR register, and then clearing the setting to 0, or by clearing both the RIE and REIE bits in this register to 0.

RE Bit (Receive Enable)

Specifies whether to enable or disable the serial data reception.

Note: Setting this bit to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, and PER in the FSR register, and ORER in the LSR register). These flags retain their previous values.

Note: Serial reception starts when a start bit is detected in asynchronous mode, or a synchronous clock input is detected in clock synchronous mode. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the receive format and reset the receive FIFO.

TE Bit (Transmit Enable)

Specifies whether to enable or disable the serial data transmission.

Note: Serial transmission starts after writing of data to be transmitted into the FTDR register under this condition. Before setting this bit to 1, be sure to set the serial mode register (SMR) and the FIFO control register (FCR) to select the transmit format and reset the transmit FIFO.

RIE Bit (Receive Interrupt Enable)

Specifies whether to enable or disable a receive-FIFO-data-full (RXIF) interrupt request when the RDF flag in the serial status register (FSR) is set to 1, a receive-data ready (DRIF) interrupt request when the DR flag in the FSR register is set to 1, a receive-error (ERIF) interrupt request when the ER flag in the FSR register is set to 1, and a break (BRIF) interrupt request when the BRK flag in the FSR register or the ORER flag in the line status register (LSR) is set to 1.

Note: RXIF interrupt requests can be cleared by reading 1 from the DR or RDF flag in the FSR register, then clearing the flag to 0, or by clearing the RIE bit to 0. DRIF interrupt requests can be cleared by reading 1 from the DR flag in the FSR register, and then clearing the setting to 0, or by clearing the RIE bit in this register to 0. Receive error interrupt (ERIF) requests and break interrupt (BRIF) requests can be cleared by clearing both the RIE and REIE bits in this register to 0.

TIE Bit (Transmit Interrupt Enable)

Specifies whether to enable or disable a transmit-FIFO-data-empty interrupt (TXIF) request when the serial transmit data is transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the transmit FIFO data register falls below the specified trigger number for transmission, and the TDFE flag in the serial status register (FSR) is set to 1.

Note: TXIF interrupt requests can be cleared either by writing a greater quantity of transmit data than the specified transmission trigger number into the FTDR register, reading 1 from the TDFE flag, and then clearing the TDFE flag to 0, or by clearing this bit to 0.

25.2.7 Serial Status Register (FSR)

The FSR register is a 16-bit register. The 8 lower-order bits indicate the status flag representing the SCIFA operating state.

The CPU can always read and write to the FSR register, but cannot write 1 to the status flags (ER, TEND, TDFE, BRK, RDF, and DR bits) in this register. These flags can be only cleared to 0 when they have first been read (after being set to 1). b3 (FER) and b2 (PER) are read-only bits that cannot be written.

Address(es): SCIFA0.FSR A006 5008h, SCIFA1.FSR A006 5408h, SCIFA2.FSR A006 5808h, SCIFA3.FSR A006 5C08h, SCIFA4.FSR A006 6008h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Value after reset:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DR	Receive Data Ready Flag	0: Reception is in progress, or no received data has remained in the FRDR register after normally completed receiving. 1: Next receive data has not been received.	R/(W) *1
b1	RDF	Receive FIFO Data Full Flag	0: The quantity of receive data in the FRDR register falls below the specified reception trigger number. 1: The quantity of receive data written in the FRDR register is equal to or greater than the specified reception trigger number.	R/(W) *1
b2	PER	Parity Error Flag	0: No receive parity error occurred in the next receive data read from the FRDR register. 1: A receive parity error occurred in the next receive data read from the FRDR register.	R
b3	FER	Framing Error Flag	0: No receive framing error occurred in the next data read from the FRDR register. 1: A receive framing error occurred in the next data read from the FRDR register.	R
b4	BRK	Break Detect Flag	0: No break signal is received. 1: A break signal is received.*2	R/(W) *1
b5	TDFE	Transmit FIFO Data Empty Flag	0: The quantity of transmit data written in the FTDR register exceeds the specified transmission trigger number. 1: The quantity of transmit data written in the FTDR register is equal to or less than the specified transmission trigger number.*3	R/(W) *1
b6	TEND	Transmit End Flag	0: Transmission is in the waiting state or in progress. 1: Transmission is completed.	R/(W) *1
b7	ER	Receive Error Flag	0: Reception is in progress or has normally completed. 1: A framing error or parity error has occurred during reception.	R/(W) *1
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. 0 can be only written to clear the flag after 1 is read.

Note 2. When a break signal is detected, transfer of the receive data (00h) to the FRDR register stops after the detection. When the break ends and the receive signal becomes mark state (high level), the transfer of receive data resumes.

Note 3. Since the FTDR register is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is 1 is “16 minus the number of non-transmitted data units”. If additional data is written, the data is ignored. The quantity of data in the FTDR register is indicated by the 8 higher-order bits of the FDR register.

DR Bit (Receive Data Ready Flag)

Indicates that the quantity of data stored in the receive FIFO data register (FRDR) falls below the specified reception trigger number, and that no next data has been received yet after the elapse of 15 ETUs*1 from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1.

[Setting condition]

- DR is set to 1 when the FRDR register contains less data than the specified reception trigger number*2, and no next data has been received yet after the elapse of 15 ETUs*1 from the last stop bit.

[Clearing conditions]

When either of the following is satisfied:

- DR is cleared to 0 when DR = 1 is read and then 0 is written to the DR flag.
- DR is cleared to 0 when all received data in the FRDR register are read.

Note 1. This is equivalent to one and a half (1.5) frames in the 8-bit format with one stop bit (ETU: elementary time unit).
Note 2. The clearing condition takes precedence when all received data in the FRDR register are read.
Note: When the RE bit in SCR is cleared to 0, the DR bit is not affected and retains its previous value.

RDF Bit (Receive FIFO Data Full Flag)

Indicates that receive data has been transferred to the receive FIFO data register (FRDR), and the quantity of data in FRDR becomes equal to or greater than the specified reception trigger number.

[Setting condition]

- RDF is set to 1 when the quantity of receive data which is equal to or greater than the specified reception trigger number are stored in the FRDR register*1.

[Clearing condition]

- RDF is cleared to 0 when the FRDR register is read until the quantity of receive data in the FRDR register falls below the specified reception trigger number after 1 is read from RDF and then 0 is written to this bit.

Note 1. Since the FRDR register is a 16-byte FIFO register, the maximum quantity of data that can be read when this bit is 1 is equivalent to the specified reception trigger number. If an attempt is made to read after all the data in the FRDR register has been read, the read data is undefined. The quantity of receive data in the FRDR register is indicated by the 8 lower-order bits of the FDR register.

PER Bit (Parity Error Flag)

Indicates whether there is a parity error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- PER is set to 1 when a parity error is present in the next data read from the FRDR register.

[Clearing condition]

- PER is cleared to 0 when no parity error is present in the next data read from the FRDR register.

FER Bit (Framing Error Flag)

Indicates whether there is a framing error in the data read from the receive FIFO data register (FRDR) in asynchronous mode.

[Setting condition]

- FER is set to 1 when a framing error is present in the next data read from the FRDR register.

[Clearing condition]

- FER is cleared to 0 when no framing error is present in the next data read from the FRDR register.

BRK Bit (Break Detect Flag)

Indicates that a break signal has been detected in receive data.

[Setting condition]

- BRK is set to 1 when data including a framing error is received, and the framing error is followed by at least one frame of data received at the space 0 level (low level).

[Clearing condition]

- BRK is cleared to 0 when software reads BRK after it has been set to 1 and then writes 0 to BRK.

TDFE Bit (Transmit FIFO Data Empty Flag)

Indicates that data has been transferred from the transmit FIFO data register (FTDR) into the transmit shift register (TSR), the quantity of data in the FTDR register becomes equal to or less than the specified transmission trigger number, and writing of transmit data to the FTDR register is enabled.

[Setting conditions]

When either of the following is satisfied:

- TDFE is set to 1 when the TE bit in SCR is 0.
- TDFE is set to 1 when the quantity of transmit data written in the FTDR register is equal to or less than the specified transmission trigger number.

[Clearing condition]

- TDFE is cleared to 0 when 0 is written in the TDFE bit after reading TDFE = 1.

TEND Bit (Transmit End Flag)

Indicates that the FTDR register contains no more valid data and transmission is completed when transmitting the last bit of the transmit data.

[Setting condition]

- TEND is set to 1 when the FTDR register does not contain transmit data when the last bit of the serial transmission data is transmitted.

[Clearing conditions]

When either of the following is satisfied:

- When transmit data is written to the FTDR register
- When 0 is written to TEND after it has been read as 1

ER Bit (Receive Error Flag)

Indicates the occurrence of a framing error, or of a parity error when receiving the parity-added data*1.

[Setting conditions]

When either of the following is satisfied:

- ER is set to 1 when the stop bit is found to be 0 after checking whether the stop bit of the received data is 1 at the end of one data receive operation*2.
- ER is set to 1 when the total number of 1s in the received data and parity bit combined does not match the even or odd parity setting specified by the PM bit in the SMR register.

[Clearing condition]

- When 0 is written to ER after it has been read as 1

Note 1. Clearing the RE bit to 0 in the SCR register does not affect this bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to the FRDR register and the receive operation is continued. Whether the data read from the FRDR register includes a receive error can be detected by the FER and PER bits in the FSR register.

Note 2. In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.

25.2.8 Bit Rate Register (BRR)

The BRR register is an 8-bit register that, together with the baud rate generator clock source selected by the CKS[1:0] bits in the serial mode register (SMR), determines the serial transmit/receive bit rate.
This register is located in the same address as that of the MDDR register and selected when the MDDRS bit in SEMR is 0. The CPU can read and write to BRR. Writing to BRR should be executed when TE = RE = 0 in the SCR register.

Address(es): SCIFA0.BRR A006 5002h, SCIFA1.BRR A006 5402h, SCIFA2.BRR A006 5802h, SCIFA3.BRR A006 5C02h, SCIFA4.BRR A006 6002h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

The BRR setting is calculated using the following formulae.
[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$N = \frac{SERICLK}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$N = \frac{SERICLK}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$N = \frac{SERICLK}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$N = \frac{SERICLK}{16 \times 2^{2n-1} \times B} \times 10^6 - 1$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

[Clock synchronous mode]

$$N = \frac{SERICLK}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bit/s)
N: Setting of the BRR register (0 ≤ N ≤ 255) (The setting must satisfy the electrical characteristics).
SERICLK: Operating frequency for peripheral modules (MHz)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see Table 25.3).

Note: The MDDR register is used to adjust the bit rate. For details, see section 25.2.9, Modulation Duty Register (MDDR).

Table 25.3 SMR Register Setting

		Setting of SMR.CKS [1:0] Bits	
n	**Clock Source**	**b1**	**b0**
0	SERICLK	0	0
1	SERICLK/4	0	1
2	SERICLK/16	1	0
3	SERICLK/64	1	1

The bit rate error in asynchronous mode is calculated using the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{(N + 1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{(N + 1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{(N + 1) \times B \times 16 \times 2^{2n-1}} - 1 \right\} \times 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

Table 25.4 list the examples of the BRR register setting in asynchronous mode, and Table 25.5 list the examples of the BRR register setting in clock synchronous mode.

Table 25.4 Bit Rates and BRR Register Settings in Asynchronous Mode

Bit Rate (bps)	SERICLK (MHz)					
	120			150		
	n	N	Error (%)	n	N	Error (%)
150						
300	3	194	0.16	3	243	0.06
600	3	97	-0.35	3	121	0.06
1200	2	194	0.16	2	243	0.06
2400	2	97	-0.35	2	121	0.06
4800	1	194	0.16	1	243	0.06
9600	1	97	-0.35	1	121	0.06
14400	1	64	0.16	1	80	0.47
19200	0	194	0.16	0	243	0.06
28800	0	129	0.16	0	162	-0.15
31250	0	119	0	0	149	0
38400	0	97	-0.35	0	121	0.06
115200				0	40	-0.76
500000						

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled. Configure settings so the range of error is no greater than 1%.
Values for the blank cells in the table can be set using the MDDR register. For details, see section 25.2.9, Modulation Duty Register (MDDR) and the Table 25.10.

Table 25.5 Bit Rates and BRR Register Settings in Clock Synchronous Mode

Bit Rate (bps)	SERICLK (MHz)			
	120		150	
	n	N	n	N
250				
500				
1000				
2500	3	187	3	233
5000	3	93	3	116
10000	2	187	2	233
25000	2	74	2	93
50000	1	149	1	187
100000	1	74	1	93
250000	0	119	0	149
500000	0	59	0	74
1000000	0	29	0	37
2500000	0	11	0	14
5000000	0	5	0	7

Blank: Setting is prohibited.
Note: Set the BRR register so that the range of error can fall within 1% or less.

Table 25.6 lists the maximum bit rates for various frequencies in asynchronous mode when the baud rate generator is used. Table 25.7 lists the maximum bit rates for various frequencies in clock synchronous mode when the baud rate generator is used. Table 25.8 and Table 25.9 list the maximum rates for external clock inputs in asynchronous mode and clock synchronous mode, respectively.

Table 25.6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Asynchronous Mode)

SERICLK (MHz)	Maximum Bit Rate (bit/s)	Settings	
		n	N
120	15000000	0	0
150	18750000	0	0

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 1. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is 1/2. When bits SEMR.ABCS0 and SEMR.BGDM are both 0, the bit rate is 1/4.

Table 25.7 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (in Clock Synchronous Mode)

SERICLK (MHz)	Discontinuous Transmission/Reception			Continuous Transmission/Reception		
	Maximum Bit Rate (bit/s)	Settings		Maximum Bit Rate (bit/s)	Settings	
		n	N		n	N
120	30000000	0	0	15000000	0	1
150	37500000	0	0	18750000	0	1

Table 25.8 Maximum Bit Rates with External Clock Input (in Asynchronous Mode)

SERICLK (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
120	30	3750000
150	37.5	4687500

Note: This is an example when the SEMR.ABCS0 bit is 1. When the ABCS0 bit is set to 0, the bit rate is 1/2 .

Table 25.9 Maximum Bit Rates with External Clock Input (in Clock Synchronous Mode)

SERICLK (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bit/s)
120	10	10000000
150	12.5	12500000

25.2.9 Modulation Duty Register (MDDR)

The MDDR register corrects the bit rate adjusted by the BRR register. The value after reset of this register is FFh. When the BRME bit in SEMR is set to 1, the bit rate generated by the on-chip baud rate generator is evenly corrected according to the settings of MDDR (MDDR/256). The relationship between the MDDR register setting and the bit rate (B) is given by the following formula. The MDDR register is located in the same address as that of the BRR register and is selected when the MDDRS bit in SEMR is 1. This register is only writable when TE = RE = 0 in the SCR register. b7 in this register is fixed to 1.

Address(es): SCIFA0.MDDR A006 5002h, SCIFA1.MDDR A006 5402h, SCIFA2.MDDR A006 5802h, SCIFA3.MDDR A006 5C02h, SCIFA4.MDDR A006 6002h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

The formulae below show the relationships between the MDDR setting and the bit rate (B) when the bit rate modulation function is used.

[Asynchronous mode]

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$B = \frac{SERICLK \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{SERICLK \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$B = \frac{SERICLK \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$B = \frac{SERICLK \times 10^6}{16 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

[Clock synchronous mode]

$$B = \frac{SERICLK \times 10^6}{8 \times 2^{2n-1} \times (256/MDDR) \times (N + 1)}$$

When the bit rate modulation is used, the bit rate average error is given by the following formulae.

- When the baud rate generator is in normal mode (SEMR.BGDM = 0):

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{B \times 64 \times 2^{2n-1} \times (256/\text{MDDR}) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/\text{MDDR}) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

- When the baud rate generator is in double-speed mode (SEMR.BGDM = 1):

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/\text{MDDR}) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 16 times the bit rate (SEMR.ABCS0 = 0))

$$\text{Error (\%)} = \left\{ \frac{\text{SERICLK} \times 10^6}{B \times 16 \times 2^{2n-1} \times (256/\text{MDDR}) \times (N+1)} - 1 \right\} 100$$

(When operating on the base clock with a frequency 8 times the bit rate (SEMR.ABCS0 = 1))

B:	Bit rate (bits/s)
N:	BRR register setting (0 ≤ N ≤ 255) (The setting must satisfy the electrical characteristics).
SERICLK:	Operating frequency for peripheral modules (MHz)
MDDR:	MDDR setting (128 ≤ MDDR ≤ 255)
n:	Baud rate generator clock source (n = 0, 1, 2, 3) (For the clock sources and values of n, see Table 25.3).

Table 25.10 Bit Rates and Settings of BRR and MDDR Registers in Asynchronous Mode

Bit Rate (bps)	SERICLK (MHz)							
	120				150			
	n	N	MDDR	Error (%)	n	N	MDDR	Error (%)
150	3	205	135	-0.003	3	247	130	-0.018
300	3	176	232	0.001	3	205	216	-0.003
600	2	205	135	-0.003	3	102	216	-0.003
1200	2	176	232	0.001	2	205	216	-0.003
2400	1	205	135	-0.003	2	102	216	-0.003
4800	1	176	232	0.001	1	205	216	-0.003
9600	0	205	135	-0.003	1	102	216	-0.003
14400	0	176	174	0.001	0	205	162	-0.003
19200	0	176	232	0.001	0	205	216	-0.003
28800	0	117	232	0.001	0	102	162	-0.003
31250	0	59	128	0.000	0	74	128	0.000
38400	0	73	194	0.007	0	102	216	-0.003
115200	0	21	173	-0.009	0	23	151	0.003
500000	0	6	239	0.028	0	6	151	-0.077

Note: These values assume bits SEMR.ABCS0 and SEMR.BGDM are both 0. When either the SEMR.ABCS0 bit or SEMR.BGDM bit is set to 1, the bit rate is doubled. When bits SEMR.ABCS0 and SEMR.BGDM are both 1, the bit rate is quadrupled. Configure settings so the range of error is no greater than 1%.

25.2.10 FIFO Control Register (FCR)

The FCR register resets the quantity of data in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR) and specifies the number of triggers. This register also specifies whether to enable the loop-back test.
The CPU can always read and write to the FCR register.

Address(es): SCIFA0.FCR A006 500Ch, SCIFA1.FCR A006 540Ch, SCIFA2.FCR A006 580Ch, SCIFA3.FCR A006 5C0Ch, SCIFA4.FCR A006 600Ch

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	LOOP	Loop-Back Test	0: Loop back test is disabled. 1: Loop back test is enabled.	R/W
b1	RFRST	Receive FIFO Data Register Reset	0: Normal operation 1: Resets the FRDR register.	R/W
b2	TFRST	Transmit FIFO Data Register Reset	0: Normal operation 1: Resets the FTDR register.	R/W
b3	MCE	Modem Control Enable	0: Modem signal is disabled.*1 1: Modem signal is enabled.	R/W
b5, b4	TTRG[1:0]	Transmit FIFO Data Trigger Number Select	b5 b4 0 0: 8 (8)*2 0 1: 4 (12)*2 1 0: 2 (14)*2 1 1: 0 (16)*2	R/W
b7, b6	RTRG[1:0]	Receive FIFO Data Trigger Number Select	In asynchronous mode: b7 b6 0 0: 1 0 1: 4 1 0: 8 1 1: 14 In clock synchronous mode: b7 b6 0 0: 1 0 1: 2 1 0: 8 1 1: 14	R/W
b10 to b8	RSTRG[2:0]	RTS# Output Active Trigger Number Select	b10 b8 0 0 0: 15 0 0 1: 1 0 1 0: 4 0 1 1: 6 1 0 0: 8 1 0 1: 10 1 1 0: 12 1 1 1: 14	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. The CTS# input level does not affect the transmit operation. Similarly, the RTS# input level does not affect the receive operation.
Note 2. Values in parentheses mean the number of empty bytes in the FTDR register when the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXIF) request is generated.

LOOP Bit (Loop-Back Test)

Internally connects between the transmit output pin (TXD) and the receive input pin (RXD) and between the RTS# pin and the CTS# pin, to perform loop-back testing.

RFRST Bit (Receive FIFO Data Register Reset)

Disables the receive data in the receive FIFO data register (FRDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

TFRST Bit (Transmit FIFO Data Register Reset)

Disables the transmit data in the transmit FIFO data register (FTDR) and makes the data to the empty state. If you set this bit to 1, be sure to clear it to 0 afterward.

MCE Bit (Modem Control Enable)

Specifies whether to enable or disable the modem control signals, CTS# and RTS#.
In clock synchronous mode, this bit should always be set to 0.

TTRG[1:0] Bits (Transmit FIFO Data Trigger Number Select)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR). When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the number specified by the TTRG[1:0] bits, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXIF) request is generated.
The setting in these bits is valid when the TTRGS bit in the FTCR register is 0. When the TTRGS bit in the FTCR register is 1, the setting of the TFTC[4:0] bits in the FTCR register is valid.

RTRG[1:0] Bits (Receive FIFO Data Trigger Number Select)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR). When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the number specified by the RTRG[1:0] bits, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXIF) request is generated.
The setting in these bits is valid when the RTRGS bit in the FTCR register is 0. When the RTRGS bit in the FTCR register is 1, the setting of the RFTC[4:0] bits in the FTCR register is valid.

RSTRG[2:0] Bits (RTS# Output Active Trigger Number Select)

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the trigger number specified by the RSTRG[2:0] bits, the RTS# signal is in the high state.
The setting in these bits is only valid when a modem signal is enabled by the MCE bit in this register in asynchronous mode.

25.2.11 FIFO Data Count Register (FDR)

The FDR register indicates the quantity of data stored in the transmit FIFO data register (FTDR) and the receive FIFO data register (FRDR).

This register indicates the quantity of transmit data in the FTDR register with the 8 higher-order bits, and the quantity of receive data in the FRDR register with the 8 lower-order bits. The CPU can always read the FDR register but cannot write to it.

Address(es): SCIFA0.FDR A006 500Eh, SCIFA1.FDR A006 540Eh, SCIFA2.FDR A006 580Eh, SCIFA3.FDR A006 5C0Eh, SCIFA4.FDR A006 600Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	T[4:0]					—	—	—	R[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	R[4:0]	Receive Data Quantity in FRDR	Indicate the quantity of receive data stored in the FRDR register.	R
b7 to b5	—	Reserved	These bits are read as 0.	R
b12 to b8	T[4:0]	Non-Transmitted Data Quantity in FTDR	Indicate the quantity of non-transmitted data stored in the FTDR register.	R
b15 to b13	—	Reserved	These bits are read as 0.	R

R[4:0] Bits

Indicate the quantity of receive data stored in the FRDR register.

00h means no received data, and 10h means that all of the received data is stored in the FRDR register.

T[4:0] Bits

Indicate the quantity of non-transmitted data stored in the FTDR register.

00h means no transmit data, and 10h means that all of the data for transmission is stored in the FTDR register.

25.2.12 Serial Port Register (SPTR)

The SPTR register controls input/output and data of the pins multiplexed to SCIFA function.
The CPU can always read and write to the SPTR register.

Note: b6, b4, b2, and b0 of this register respectively indicate the input status of their corresponding pins. See the descriptions for each bit for details. Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Address(es): SCIFA0.SPTR A006 5010h, SCIFA1.SPTR A006 5410h, SCIFA2.SPTR A006 5810h, SCIFA3.SPTR A006 5C10h, SCIFA4.SPTR A006 6010h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	RTS2IO	RTS2DT	CTS2IO	CTS2DT	SCKIO	SCKDT	SPB2IO	SPB2DT
Value after reset:	0	0	0	0	0	0	0	0	0	x	0	x	0	x	0	x

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	SPB2DT	Serial Port Break Data Select	Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2IO bit. See Table 25.13.	R/W
b1	SPB2IO	Serial Port Break Input/Output	Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.	R/W
b2	SCKDT	SCK Port Data Select	Controls the SCK pin in combination with the CM bit in the SMR register, the SCKIO bit, and the CKE1 and CKE0 bits in the SCR register. See Table 25.15. Setting of this bit is valid for channels 0 to 2 only.	R/W
b3	SCKIO	SCK Port Input/Output	Controls the SCK pin in combination with the CM bit in the SMR register, the SCKDT bit, and the CKE1 and CKE0 bits in the SCR register. See Table 25.15. Setting of this bit is valid for channels 0 to 2 only.	R/W
b4	CTS2DT	CTS# Port Data Select	Controls the CTS# pin in combination with MCE bit in FCR and CTS2IO bit. See Table 25.12. Setting of this bit is valid for channels 0 to 2 only.	R/W
b5	CTS2IO	CTS# Port Output Specify		R/W
b6	RTS2DT	RTS# Port Data Select	Controls the RTS# pin in combination with MCE bit in FCR and RTS2IO bit. See Table 25.11. Setting of this bit is valid for channels 0 to 2 only.	R/W
b7	RTS2IO	RTS# Port Output Specify		R/W
b15 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPB2DT Bit (Serial Port Break Data Select)

This bit specifies the output level of the TXD pin when the setting of the SCR.TE bit is 0. The RXD pin input status can be read from this bit regardless of the SPB2IO bit setting. However, the RXD pin function must have been selected with the multi-function pin controller (MPC).

SPB2IO Bit (Serial Port Break Input/Output)

Controls the TXD pin in combination with the TE bit in the SCR register and the SPB2DT bit.

SCKDT Bit (SCK Port Data Select)

The SCK pin status can be read from this bit regardless of the SCKIO bit setting. (When the SCK pin is used for input, the input signal is invalid (has no means) but the pin status can be read.) However, the SCK pin function must have been selected with the multi-function pin controller (MPC). The setting of this bit is only valid for channels 0 to 2.

SCKIO Bit (SCK Port Input/Output)

Specifies input or output status of the SCK pin. This bit controls the SCK pin in combination with the SCKDT bit, the CM bit in the SMR register, and the CKE1 and CKE0 bits in the SCR register. The setting of this bit is only valid for channels 0 to 2.

CTS2DT Bit (CTS# Port Data Select)

The status of the CTS# pin can be read from this bit regardless of the CTS2IO bit setting. However, the CTS# pin function must have been selected with the multi-function pin controller (MPC). The setting of this bit is only valid for channels 0 to 2.

RTS2DT Bit (RTS# Port Data Select)

The status of the RTS# pin can be read from this bit regardless of the RTS2IO bit setting. However, the RTS# pin function must have been selected with the multi-function pin controller (MPC). The setting of this bit is only valid for channels 0 to 2.

Table 25.11 RTS# Pin Status

FCR.MCE Bit Setting	RTS2IO Bit Setting	RTS2DT Bit Setting	RTS# Pin Status
0	0	×	Setting prohibited*1
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control output

×: Don't care

Note 1. There is no problem with the initial setting if the RTS# pin is not used.

Table 25.12 CTS# Pin Status

FCR.MCE Bit Setting	CTS2IO Bit Setting	CTS2DT Bit Setting	CTS# Pin Status
0	0	×	Setting prohibited*1
0	1	0	Low output
0	1	1	High output
1	×	×	Modem control input

×: Don't care

Note 1. There is no problem with the initial setting if the CTS# pin is not used.

Table 25.13 TXD Pin Status

SCR.TE Bit Setting	SPB2IO Bit Setting	SPB2DT Bit Setting	TXD Pin Status
0	0	×	Setting prohibited
0	1	0	Low output
0	1	1	High output
1	×	×	Transmit data output

×: Don't care

25.2.13 Line Status Register (LSR)

The LSR register is a 16-bit register. The PER and FER bits indicate the number of receive errors in the receive FIFO data register. 1 cannot be written to the ORER status flag. The flag should be read as 1 prior to clearing it to 0.

Address(es): SCIFA0.LSR A006 5012h, SCIFA1.LSR A006 5412h, SCIFA2.LSR A006 5812h, SCIFA3.LSR A006 5C12h, SCIFA4.LSR A006 6012h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PER[3:0]				—	—	FER[3:0]				—	ORER
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ORER	Overrun Error Flag	0: Reception is in progress or has normally completed. 1: An overrun error has occurred during reception.	R/(W) *1
b1	—	Reserved	This bit is read as 0.	R
b5 to b2	FER[3:0]	Framing Error Count	Indicates the quantity of data with a framing error among the receive data stored in the receive FIFO data register (FRDR).	R
b7, b6	—	Reserved	These bits are read as 0.	R
b11 to b8	PER[3:0]	Parity Error Count	Indicates the quantity of data with a parity error among the receive data stored in the receive FIFO data register (FRDR).	R
b15 to b12	—	Reserved	These bits are read as 0.	R

Note 1. To clear the flag, 0 can be only written after 1 is read.

ORER Bit (Overrun Error Flag)

Indicates that receive operation abnormally stops due to occurrence of an overrun error. This flag is not affected and retains its previous state if the RE bit in the serial control register (SCR) is cleared to 0. The receive FIFO data register (FRDR) retains the data before an overrun error occurred, and newly received data is lost. When the ORER bit is set to 1, the SCIFA cannot continue subsequent serial reception.

[Setting condition]

- When the next serial reception is completed with the receive FIFO in full state (16-byte data is received)

[Clearing condition]

- When 0 is written to ORER after being read as 1.

Note: When the internal clock is selected while the SCIFA is in clock synchronous mode, the amount of receive data can be controlled, so no overrun occurs.

FER[3:0] Bits (Framing Error Count)

The values of bits 5 to 2 indicate the quantity of data with a framing error after the ER bit in the FSR register is set. Reading 0000 from the FER[3:0] bits means all 16-byte receive data in the FRDR register have a framing error.

PER[3:0] Bits (Parity Error Count)

The values of bits 11 to 8 indicate the quantity of data with a parity error after the ER bit in the FSR register is set. Reading 0000 from the PER[3:0] bits means all 16-byte receive data in the FRDR register have a parity error.

25.2.14 Serial Extended Mode Register (SEMR)

The SEMR register specifies either LSB or MSB first, enables the noise cancellation, operation in normal or double-speed mode of the baud rate generator, and bit rate modulation, and selects the modulation register and the sampling count (either 8 or 16 times).

Address(es): SCIFA0.SEMR A006 5014h, SCIFA1.SEMR A006 5414h, SCIFA2.SEMR A006 5814h, SCIFA3.SEMR A006 5C14h, SCIFA4.SEMR A006 6014h

	b7	b6	b5	b4	b3	b2	b1	b0
	BGDM	—	BRME	MDDRS	DIR	NFEN	—	ABCS0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ABCS0	Asynchronous Base Clock Select	0: Operates on a frequency 16 times the transfer rate as the base clock. 1: Operates on a frequency 8 times the transfer rate as the base clock.	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	NFEN	Noise Cancellation Enable	0: Noise cancellation for the RxD pin is disabled. 1: Noise cancellation for the RxD pin is enabled.	R/W
b3	DIR	Data Transfer Direction Select	0: Transmits the data in the FTDR register by the LSB-first method. The received data is stored in the FRDR register by the LSB-first method. 1: Transmits the data in the FTDR register by the MSB-first method. The received data is stored in the FRDR register by the MSB-first method.	R/W
b4	MDDRS	Modulation Duty Register Select	0: BRR register is accessible. 1: MDDR register is accessible.	R/W
b5	BRME	Bit Rate Modulation Enable	0: Bit rate modulation is disabled. 1: Bit rate modulation is enabled.	R/W
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b7	BGDM	Baud Rate Generator Double-Speed Mode Select	0: Baud rate generator normal mode: Baud rate generator operates on the clock signal produced by dividing the clock source by two. 1: Baud rate generator double-speed mode: Baud rate generator operates on the clock signal produced by the clock source (no frequency division).	R/W

ABCS0 Bit (Asynchronous Base Clock Select)

Selects the base clock for 1-bit period in asynchronous mode.

This bit setting is valid only in asynchronous mode (i.e., when the CM bit in the SMR register is 0).

NFEN Bit (Noise Cancellation Enable)

Reduces noise of the input to the RxD pin. This function is only valid in asynchronous mode. For details, see section 25.7, Noise Cancellation.

In clock synchronous mode, this bit should always be set to 0.

DIR Bit (Data Transfer Direction Select)

Selects the serial communication format. This bit is valid only when the transmit/receive data is in 8-bit formats.*1

Note 1. Asynchronous mode or clock synchronous mode with the 8-bit data length

MDDRS Bit (Modulation Duty Register Select)

Selects the register to be enabled access to it.

BRME Bit (Bit Rate Modulation Enable)

Specifies whether to enable or disable the bit rate modulation.

BGDM Bit (Baud Rate Generator Double-Speed Mode Select)

Selects operating mode of the baud rate generator. When setting 1 in this bit, the baud rate generator included in the SCIFA operates in double-speed mode. The setting of this bit is only effective in asynchronous mode (SMR.CM bit = 0) when the internal clock is selected as the clock source (SCR.CKE[1:0] = 00b). Use normal mode under any other settings.

25.2.15 FIFO Trigger Control Register (FTCR)

The FTCR register is a 16-bit register that specifies FIFO trigger conditions. The CPU can always read from and write to the FTCR register.

Address(es): SCIFA0.FTCR A006 5016h, SCIFA1.FTCR A006 5416h, SCIFA2.FTCR A006 5816h, SCIFA3.FTCR A006 5C16h, SCIFA4.FTCR A006 6016h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RTRGS	—	—	RFTC[4:0]					TTRGS	—	—	TFTC[4:0]				
Value after reset:	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	TFTC[4:0]	Transmit FIFO Data Trigger Number	00h: Transmit data trigger number is 0. 0Fh: Transmit data trigger number is 15. Do not set 10h to 1Fh in these bits.	R/W
b6, b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b7	TTRGS	Transmit Trigger Select	0: TTRG[1:0] bits in FCR are valid. 1: TFTC[4:0] bits in FTCR are valid.	R/W
b12 to b8	RFTC[4:0]	Receive FIFO Data Trigger Number	01h: Receive data trigger number is 1. 10h: Receive data trigger number is 16. Do not set 00h and 11h to 1Fh in these bits.	R/W
b14, b13	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b15	RTRGS	Receive Trigger Select	0: RTRG[1:0] bits in FCR are valid. 1: RFTC[4:0] bits in FTCR are valid.	R/W

TFTC[4:0] Bits (Transmit FIFO Data Trigger Number)

Specify the reference quantity of data for transmission (i.e., the threshold number of entries to trigger the writing of further data for transmission) for setting of the TDFE flag in the serial status register (FSR).

When the number of entries for transmission in the transmission FIFO, i.e. the number of entries written to the transmit FIFO data register (FTDR) that are yet to be transmitted, falls to or below the specified trigger number for transmission, the TDFE flag is set to 1 and a transmit FIFO data empty interrupt (TXIF) request is generated.

RFTC[4:0] Bits (Receive FIFO Data Trigger Number)

Specify the reference quantity of receive data (i.e., the threshold number of entries to trigger the reading of received data) for setting of the RDF flag in the serial status register (FSR).

When the number of entries in the reception FIFO, i.e. the number of entries yet to be read from the receive FIFO data register (FRDR), rises to or above the specified trigger number for reception, the RDF flag is set to 1 and a receive FIFO data full interrupt (RXIF) request is generated.

25.3 Operation

25.3.1 Overview

For serial communication, the SCIFA can select either asynchronous mode in which characters are synchronized individually or a clock synchronous mode in which communication is synchronized with clock pulses.
The SCIFA has a 16-stage FIFO buffer for both transmission and receptions, reducing the overhead of the CPU and enabling continuous high-speed communication. The RTS# and CTS# signals are provided as modem control signals.
Selection of a transmission/reception format is enabled with the serial mode register (SMR). Table 25.14 shows the transmission format which can be selected in the serial mode register (SMR). As shown in Table 25.15, the SCIFA clock source can be set by using the CKE[1:0] bits of the serial control register (SCR).

(1) Asynchronous Mode

- Data length is selectable either 7 or 8 bits
- Parity addition and 1- or 2-bit stop bit addition are selectable.
  (The combination of the preceding selections determines the transmission/reception format and character length).
- In reception, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The stored data quantities are indicated in the FIFO data count register (FDR), respectively for transmit and receive FIFO data.
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of on-chip baud rate generator and can output the clock with a frequency 16 (or 8) times the bit rate.
  When an external clock is selected, the external clock input must have a frequency 16 (or 8) times the bit rate. (The on-chip baud rate generator is not used.)

(2) Clock Synchronous Mode

- The transmission/reception format is fixed to the 8-bit data length.
- In reception, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the SCIFA clock source.
  When an internal clock is selected, the SCIFA operates using the clock of the on-chip baud rate generator, and outputs this clock to external devices as the synchronous clock.
  When an external clock is selected, the SCIFA operates on the input synchronous clock not using the on-chip baud rate generator.

Table 25.14 SMR Register Settings and SCIFA Communication Formats

SMR Register					SCIFA Transmission/Reception Format		
b7	b6	b5	b3				
CM	CHR	PE	STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous mode	8 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
	1	0	0		7 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
1	×	×	×	Clock synchronous mode	8 bits	Not set	None

×: Don't care

Table 25.15 SMR, SCR, and SPTR Register Settings and SCIFA Clock Source Selection

SMR Register	SCR Register		SPTR Register				
b7	b1	b0	b3	b2			
CM	CKE[1:0]		SCKIO	SCKDT	Mode	Clock Source	SCK Pin Function
0	0	0	0	×	Asynchronous mode	Internal	Input pin (input signal invalid) (Initial state)
			1	0			SCK pin state: Low
			1	1			SCK pin state: High
		1	×	×			Outputs a clock with frequency 16/8 times the bit rate*1
	1	0	×	×		External	Inputs a clock with frequency 16/8 times the bit rate*2
		1	×	×		Setting prohibited	
1	0	×	×	×	Clock synchronous mode	Internal	Outputs the synchronous clock
	1	0	×	×		External	Inputs the synchronous clock
		1	×	×		Setting prohibited	

×: Don't care

Note 1. SEMR.ABCS0 = 0: Output a clock that has a frequency 16 times the bit rate.
SEMR.ABCS0 = 1: Output a clock that has a frequency 8 times the bit rate.

Note 2. SEMR.ABCS0 = 0: Input a clock that has a frequency 16 times the bit rate.
SEMR.ABCS0 = 1: Input a clock that has a frequency 8 times the bit rate.

25.3.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIFA are independent, so full duplex communication is possible. The transmitter and receiver are 16-stage FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmission and reception.

Figure 25.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIFA monitors the line and starts serial communication when the line goes to the space (low) state, considered as a start bit. One serial character consists of a start bit (low), data (LSB first when LSB-first transfer is selected), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCIFA synchronizes at the falling edge of the start bit. The SCIFA samples each data bit on the eighth pulse of a clock with a frequency 16 or 8 times the bit rate*1. Receive data is latched at the center of each bit.

Note 1. When the SEMR.ABCS0 bit = 0, data is sampled on the eighth pulse of a clock with a frequency 16 times the bit rate.
When the SEMR.ABCS0 bit = 1, data is sampled on the fourth pulse of a clock with a frequency 8 times the bit rate.

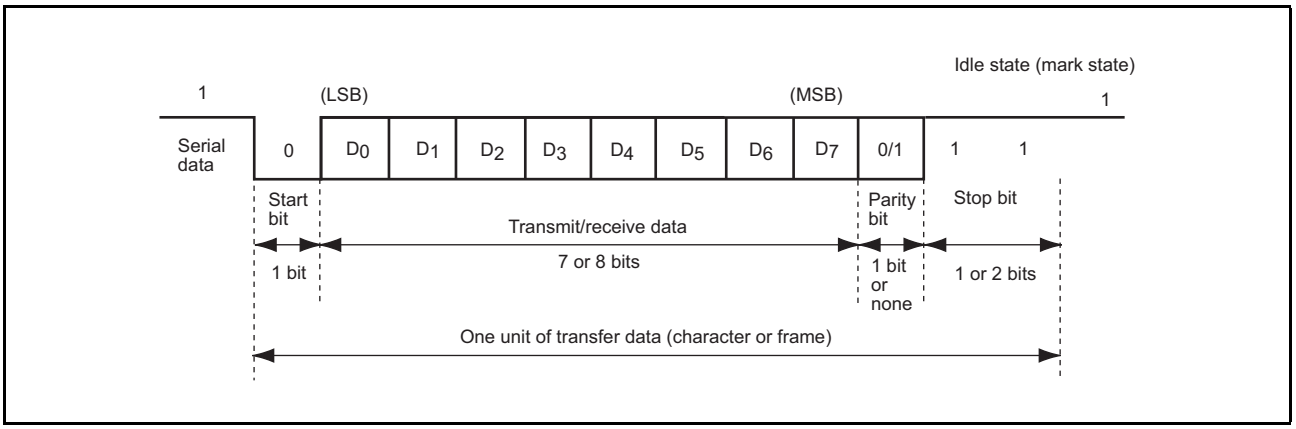


Figure 25.2 Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits when LSB-First Transfer is Selected)

(1) Transmit/Receive Formats

Table 25.16 lists the eight communications formats that can be selected in asynchronous mode. The format is selected by setting in the serial mode register (SMR).

Table 25.16 Serial Communications Formats (in Asynchronous Mode)

SMR Setting			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-bit data								STOP		
		1	START	8-bit data								STOP	STOP	
	1	0	START	8-bit data								P	STOP	
		1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
		1	START	7-bit data							STOP	STOP		
	1	0	START	7-bit data							P	STOP		
		1	START	7-bit data							P	STOP	STOP	

START: Start bit
STOP: Stop bit
P: Parity bit

(2) Clock

An SCIFA transmit/receive clock can be selected from two types of clock sources: the internal clock generated by the on-chip baud rate generator or the external clock input from the SCK pin. The clock source is selected by the settings of the CM bit in the serial mode register (SMR), the CKE[1:0] bits in the serial control register (SCR), and the ACS0 bit in the serial extended mode register (SEMR). For clock source selection, refer to Table 25.15.

When an external clock is input at the SCK pin, it must have a frequency equal to 16/8 times the desired bit rate.

When the SCIFA operates on an internal clock, it can output a clock signal on the SCK pin. The frequency of this output clock is 16/8 times the desired bit rate.

(3) Transmitting and Receiving Data

- SCIFA Initialization (in Asynchronous Mode)

Before transmitting or receiving data, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA as follows.

When changing operating mode or communication format, always clear the TE and RE bits in the SCR register to 0 before following the procedure given below. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing TE and RE to 0, however, does not initialize the serial status register (FSR), transmit FIFO data register (FTDR), or receive FIFO data register (FRDR), which retain their previous contents. Clear TE to 0 after all transmit data has been transmitted and the TEND flag in the FSR register is set. The TE bit can be cleared to 0 during transmission, but the transmit data (the TXD pin output level) after the TE bit is cleared to 0 depends on the settings of the SPB2IO and SPB2DT bits in the SPTR register. Set the TFRST bit in the FCR register to 1 and reset the FTDR register before TE is set to 1 again to start transmission.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. SCIFA operation becomes unreliable if the clock is stopped. Figure 25.3 shows a sample flowchart for initializing the SCIFA in asynchronous mode.

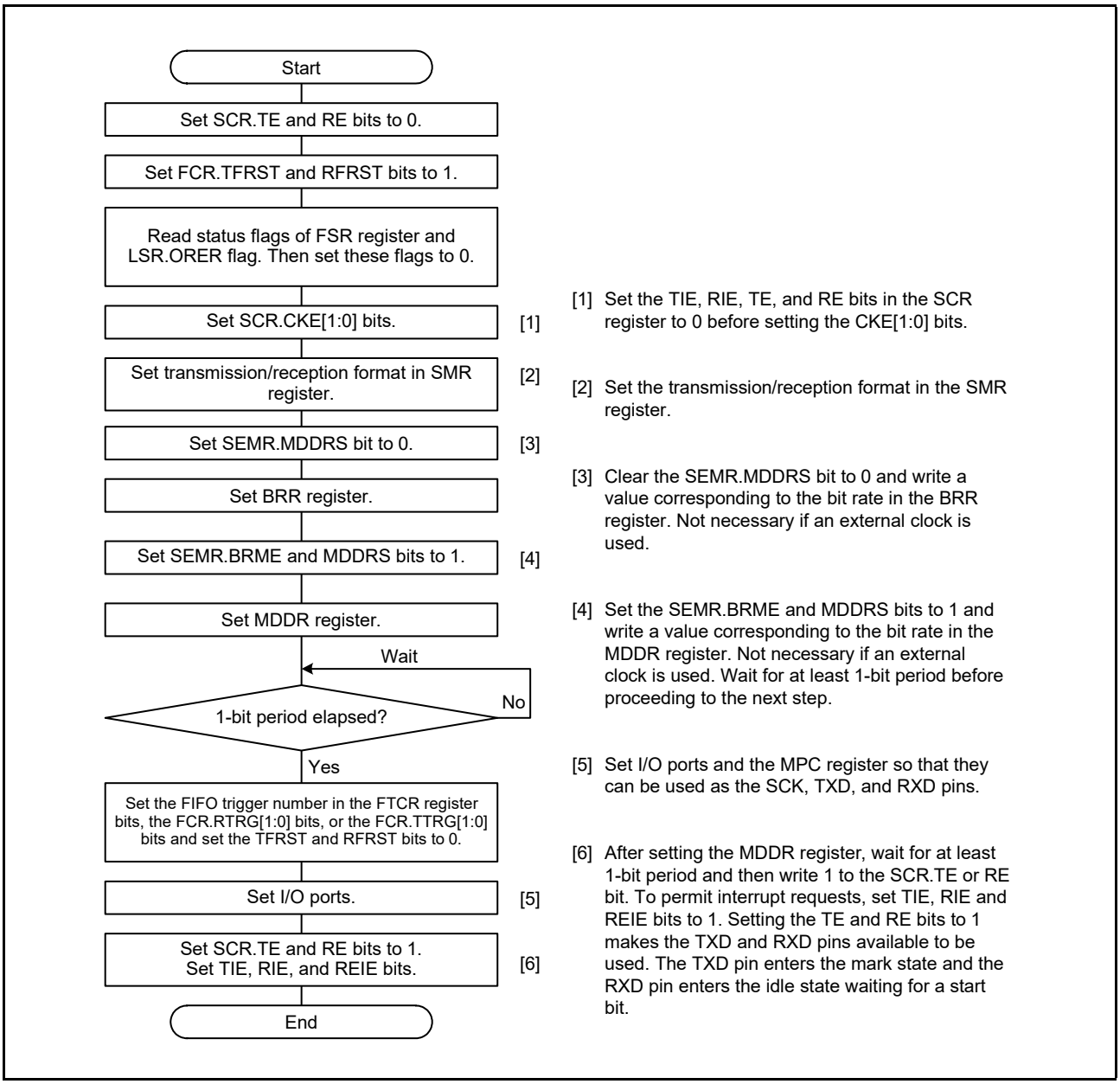
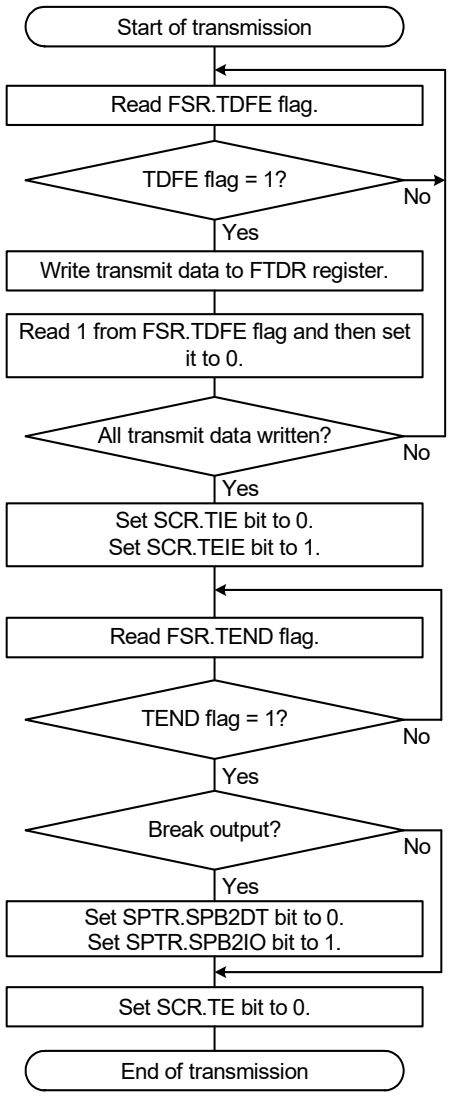


Figure 25.3 Sample Flowchart for SCIFA Initialization in Asynchronous Mode

- Transmitting Serial Data (in Asynchronous Mode)

Figure 25.4 shows a sample flowchart for serial transmission in asynchronous mode.
Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data to the FTDR register. The transmit data quantity that can be written is "16 - FDR.T[4:0]".
A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure (when the FTDR register is written by the CPU):
To continue serial transmission, confirm that the TDFE flag in the FSR register is set to 1, and then write 0 to the flag.

[3] Serial transmission end procedure (when TXIF or TEIF interrupt is used):
After the last transmission data is written to the FTDR register, set the TIE bit to 0 and the TEIE bit to 1 in the SCR register. If the DMAC is activated to write the transmit data to the FTDR register, enable the transmit end interrupt. This will generate a CPU interrupt request when the last transmit data is written in the FTDR register. The TIE and TEIE bit settings will be changed by the CPU interrupt process.
This procedure is unnecessary if the TXIF or TEIF interrupt is not used.

[4] Serial transmission end confirmation:
Read the FSR register and confirm that the TEND flag has changed to 1.
A change of the TEND flag from 0 to 1 can be known by a TEIF interrupt request.

[5] Break output during serial transmission:
To cause a break to be output during serial transmission, set the SPB2DT bit to 0 and the SPB2IO bit to 1 in the SPTR register. Then set the SCR.TE bit to 0.*1

In Step [1], the number of data bytes that can be written can be checked by the FDR.T[4:0] flag (non-transmitted data quantity in the FTDR register).

Note 1. The SPB2DT bit of the SPTR register indicates the input status of the RXD pin. Writings to the SPB2DT and SPB2IO bits of this register in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written to the SPB2DT bit. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

Figure 25.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode

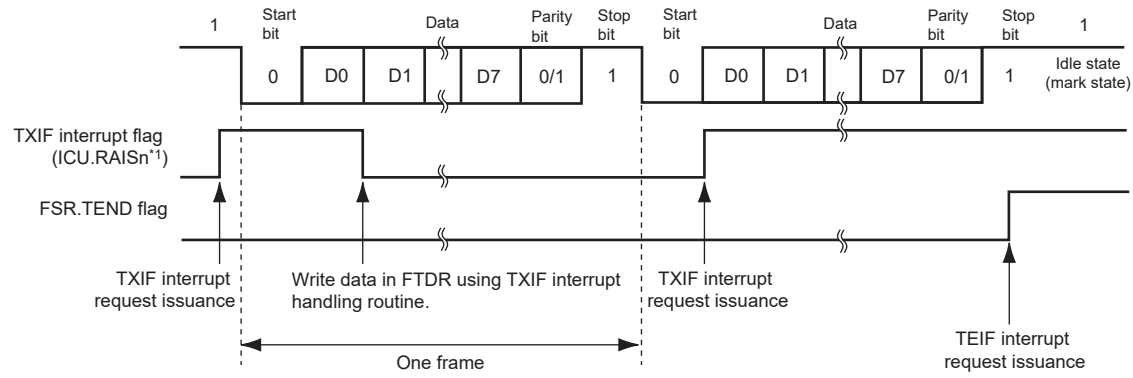
In asynchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXIF interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number specified in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag is set. If the TIE bit in the serial control register (SCR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXIF) request is generated.

   The serial transmit data is output from the TXD pin in the following order.

   (a) Start bit: One-bit 0 is output.
   (b) Transmit data: 8- or 7-bit data is output in LSB-first order (when LSB-first transfer is selected).
   (c) Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
   (d) Stop bit(s): One or two 1 bits (stop bits) are output.
   (e) Mark state: 1 is output continuously until the start bit that starts the next transmission is output.
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the stop bit. If data is present, the data is transferred from the FTDR register to the TSR register, the stop bit is output, and then serial transmission of the next frame is started. If there is no data to be transmitted, the TEND flag in the FSR register is set to 1, the stop bit is output, and then the SCIFA enters the mark state (high level) in which 1 is output continuously.

Figure 25.5 shows an example of the operation for transmission in asynchronous mode.



Note 1. For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

Figure 25.5 Example of Transmit Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

4. When modem control is enabled, transmission can be stopped/resumed by the input level to the CTS# pin. When a high level is input to the CTS# pin during transmission, the SCIFA enters the mark state (high level) after completion of one-frame data transmission. When a low level is input to the CTS# pin, output of the next data to be transmitted begins with a start bit. Figure 25.6 shows an example of the operation for transmission when using the modem control function.

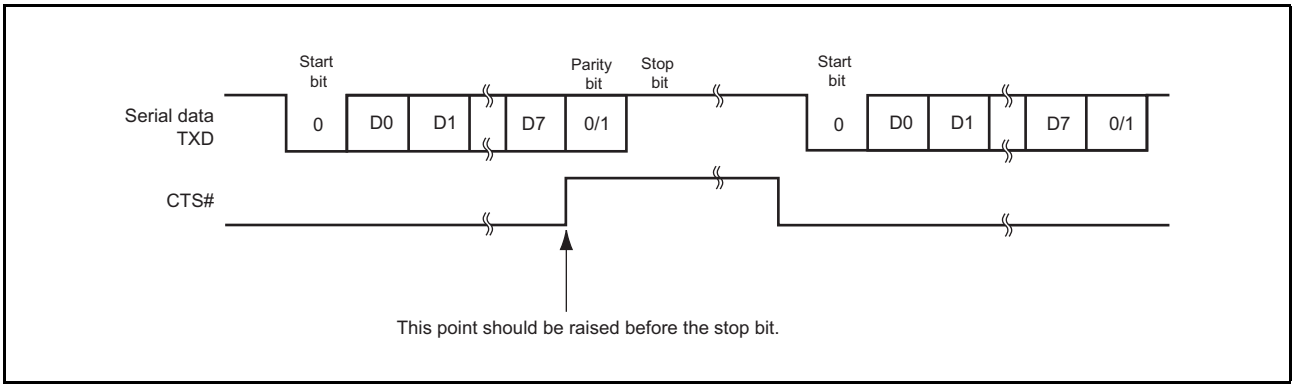
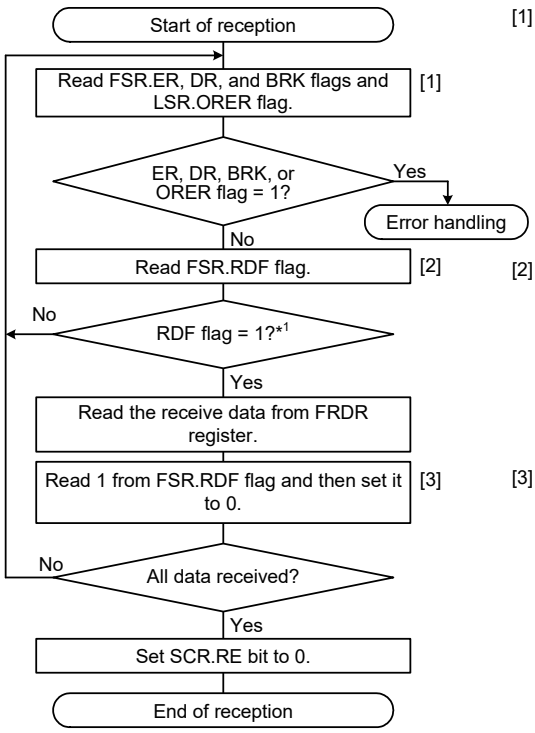


Figure 25.6 Example of Transmit Operation in Asynchronous Mode Using Modem Control Function (CTS#)

- Receiving Serial Data (in Asynchronous Mode)

Figure 25.7 and Figure 25.8 show sample flowcharts for serial reception in asynchronous mode.
Follow the procedure given below for serial data reception after enabling the SCIFA for reception.



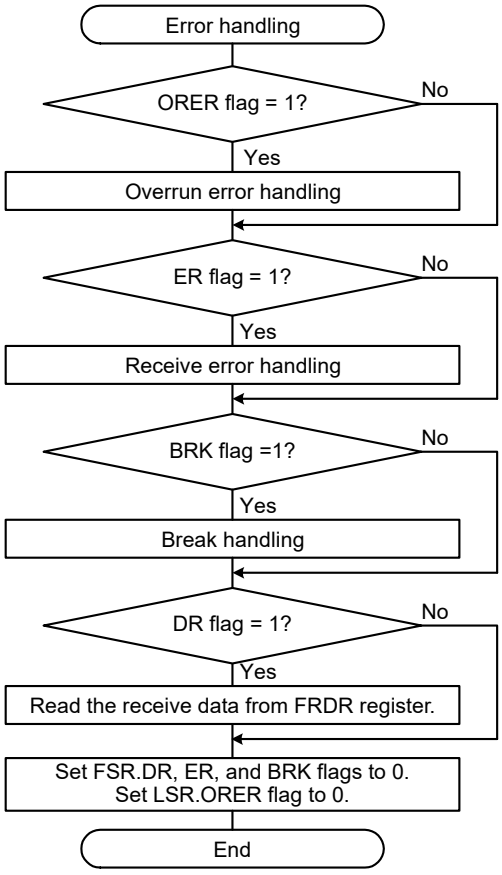
[1] Receive error handling and break detection:
When a receive error occurs, read the ER, DR, and BRK flags in the FSR register and the LSR.ORER flag to identify the error. After handling the error as specified, clear the ER, DR, BRK, and ORER flags to 0.
In the case where a framing error occurs, a break can be detected by reading the value of the RXD pin.
Error occurrence can be known by a BRIF, ERIF, or DRIF interrupt request.

[2] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data at least as much as the amount specified as the receive data trigger number from those stored in the FRDR register. The receive data quantity that can be read can be checked by the FDR.R[4:0] flag.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[3] Serial reception continuation procedure (when the FRDR register is read by the CPU):
To continue serial reception, confirm that the RDF flag in the FSR register is set to 1, and then write 0 to the flag.

Note 1. The number of all the received data units should be an integral multiple of the receive data trigger number. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 25.7 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (1)



- Whether a framing error or parity error has occurred in the received data that is read from the FRDR register can be checked with the FER and PER flags in the FSR register.
- When a break signal is received, the received data is not transferred to the FRDR register while the BRK flag is set to 1. However, note that the last data in the FRDR register is 00h and the break data in which a framing error occurred is stored.

Figure 25.8 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (2)

In asynchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA monitors the communication line, and if a 0 start bit is detected, it performs internal synchronization to start reception.
2. The received data is stored into the RSR register in LSB-to-MSB order (when LSB-first transfer is selected).
3. The parity bit and stop bit are received.
   After receiving these bits, the SCIFA carries out the following checks.
   (a) Stop bit check: The SCIFA checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
   (b) The SCIFA checks whether receive data can be transferred from the receive shift register (RSR) to the receive FIFO data register (FRDR).
   (c) Parity bit check: The SCIFA checks whether the parity bit is an expected value.
   (d) Overrun error check: The SCIFA checks whether the ORER flag is 0, indicating that the overrun error has not occurred.
   (e) Break check: The SCIFA checks whether the BRK flag is 0, indicating that the break state is not set.

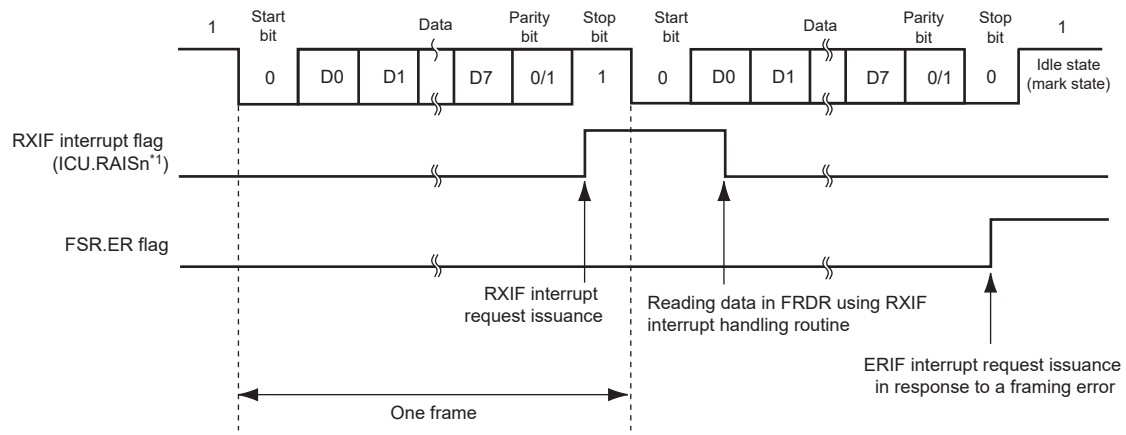
If all the above checks are passed, the receive data is stored in the FRDR register.

Note: When a parity error or a framing error occurs, reception is not suspended.

4. When receive data units equaling or exceeding the specified reception trigger number are stored in the receive FIFO data register (FRDR) and the RDF flag is changed to 1, a receive FIFO data full interrupt (RXIF) request is generated while the RIE bit in the SCR register is set to 1. When the quantity of data in the FRDR register falls below the specified reception trigger number and the RIE bit in the SCR register is set to 1, a receive data ready interrupt (DRIF) request is generated if no next data is received after the elapse of 15 ETUs*1 from the last stop bit (the DR flag in the FSR register is 1). When the ER flag in the FSR register is changed to 1, a receive error interrupt (ERIF) request is generated while the RIE or REIE bit in the SCR register is set to 1. When the BRK or ORER flag is changed to 1 in the FSR register, a break reception interrupt (BRIF) request is generated while the RIE or REIE bit in the SCR register is set to 1.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

Figure 25.9 shows an example of the operation for reception in asynchronous mode.



Note 1. For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

Figure 25.9 Example of SCIFA Receive Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected)

5. When modem control is enabled, the RTS# signal that indicates the FRDR register has space is output. When the RTS# pin is at low level, reception is possible. The RTS# pin being at the high level indicates that the number of entries in the FRDR register is equal to or greater than the threshold for output of the active level of the RTS# signal and that the transmission of further data needs to be suspended until the FRDR register has enough space. Figure 25.10 shows an example of the operation for reception in asynchronous mode when using the modem control function.

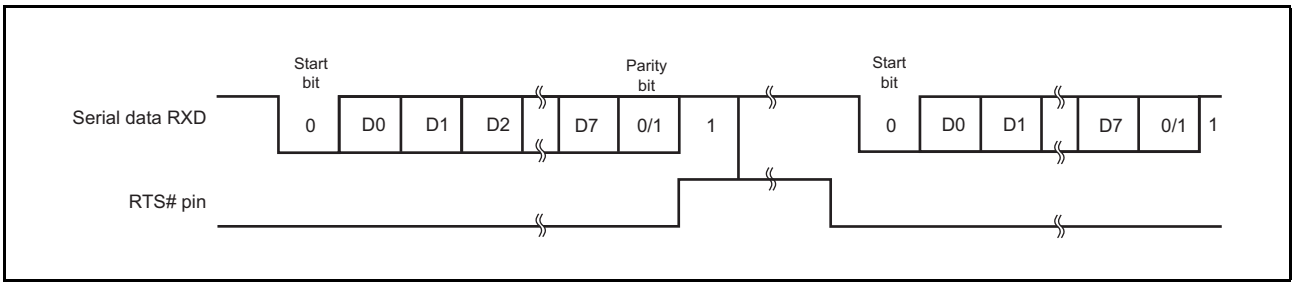
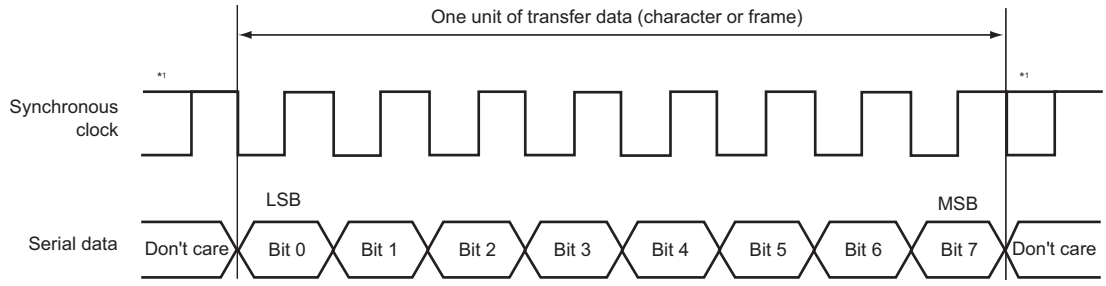


Figure 25.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Modem Control Function (RTS#)

25.3.3 Operation in Clock Synchronous Mode

In clock synchronous mode, the SCIFA transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.
Full-duplex communication is possible because the SCIFA transmitter and receiver are independent and share the same clock. Since the transmitter and the receiver have 16-stage FIFO buffers, respectively, continuous transmission or reception is possible by reading or writing data while transmission or reception is in progress.
Figure 25.11 shows the general format in clock synchronous serial communication.



Note 1. High level except in continuous transmission/reception

Figure 25.11 Data Format in Clock Synchronous Communication (when LSB-First Transfer is Selected)

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the synchronous clock to the next. Data is guaranteed valid at the rising edge of the synchronous clock.
In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB (when LSB-first transfer is selected).
In clock synchronous mode, the SCIFA receives data by synchronizing with the rising edge of the synchronous clock.

(1) Transmit/Receive Formats

The data length is fixed at eight bits.
No parity bit can be added.

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIFA transmit/receive clock according to the settings of the CM bit in the serial mode register (SMR), the CKE[1:0] bits in the serial control register (SCR), and the ACS0 bit in the serial extended mode register (SEMR).
When the SCIFA operates on an internal clock, it outputs the synchronous clock signal at the SCK pin. Eight synchronous clock pulses are output per transmitted or received character. Unless the SCIFA is transmitting or receiving, the synchronous clock signal remains in the high state. When the SCIFA only receives data on an internal clock, the internal clock signal outputs while the RE bit in the SCR register is 1 until the number of data units in the receive FIFO reaches the specified reception trigger number.

(3) Transmitting and Receiving Data

- SCIFA Initialization (in Clock Synchronous Mode)

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCR), and then initialize the SCIFA by performing the following procedure.
Similarly, before changing the mode or communication format, clear the TE and RE bits to 0, and then change it by performing the following procedure. Clearing TE to 0 initializes the transmit shift register (TSR). Clearing RE to 0, however, does not initialize the RDF, PER, FER, and ORER flags and the receive FIFO data register (FRDR), which retain their previous contents.

Figure 25.12 shows a sample flowchart for initializing the SCIFA in clock synchronous mode.

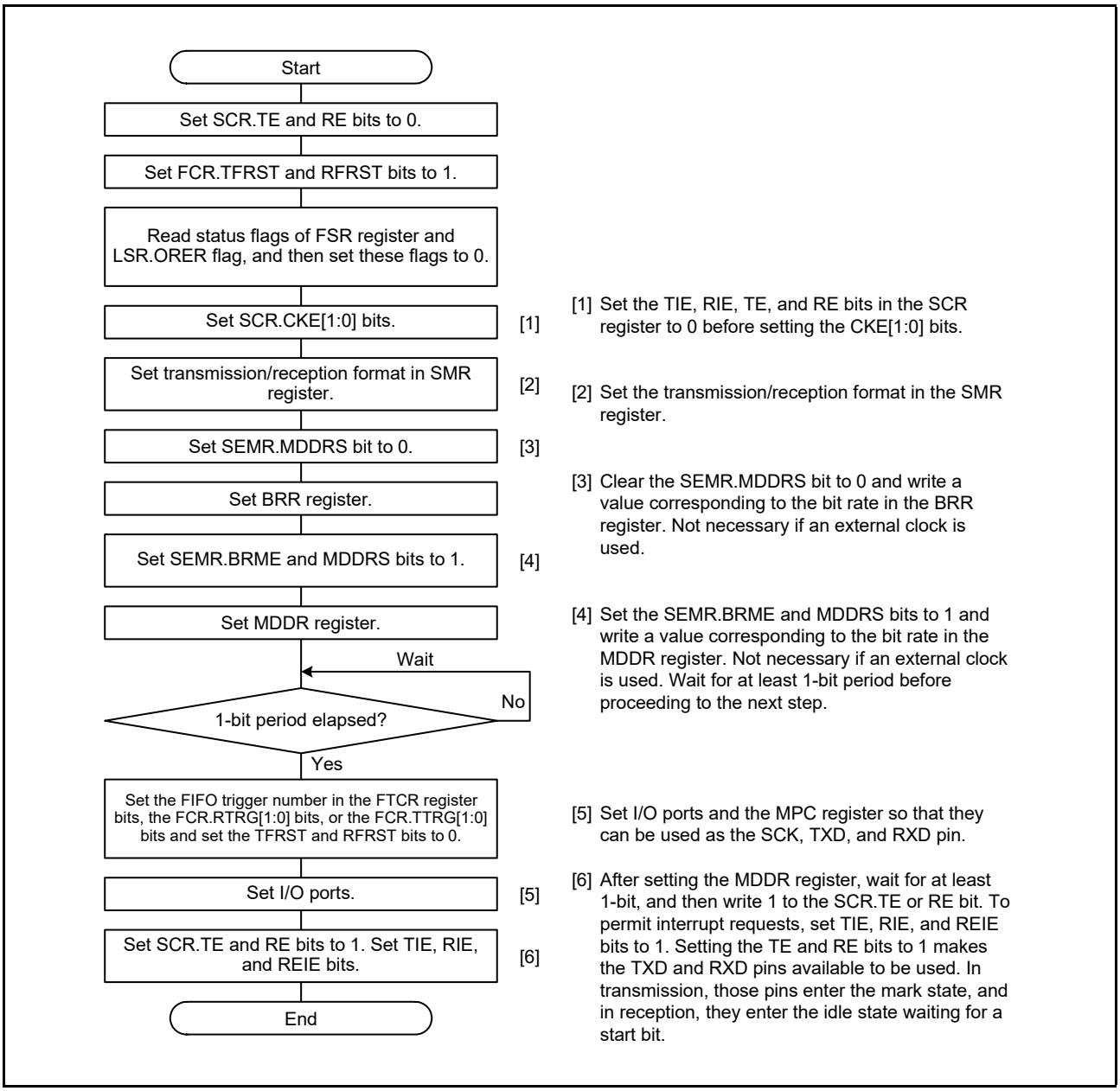


Figure 25.12 Sample Flowchart for SCIFA Initialization in Clock Synchronous Mode

- Transmitting Serial Data (in Clock Synchronous Mode)

Figure 25.13 shows a sample flowchart for transmitting serial data in clock synchronous mode.
Follow the procedure given below for serial data transmission after enabling the SCIFA for transmission.

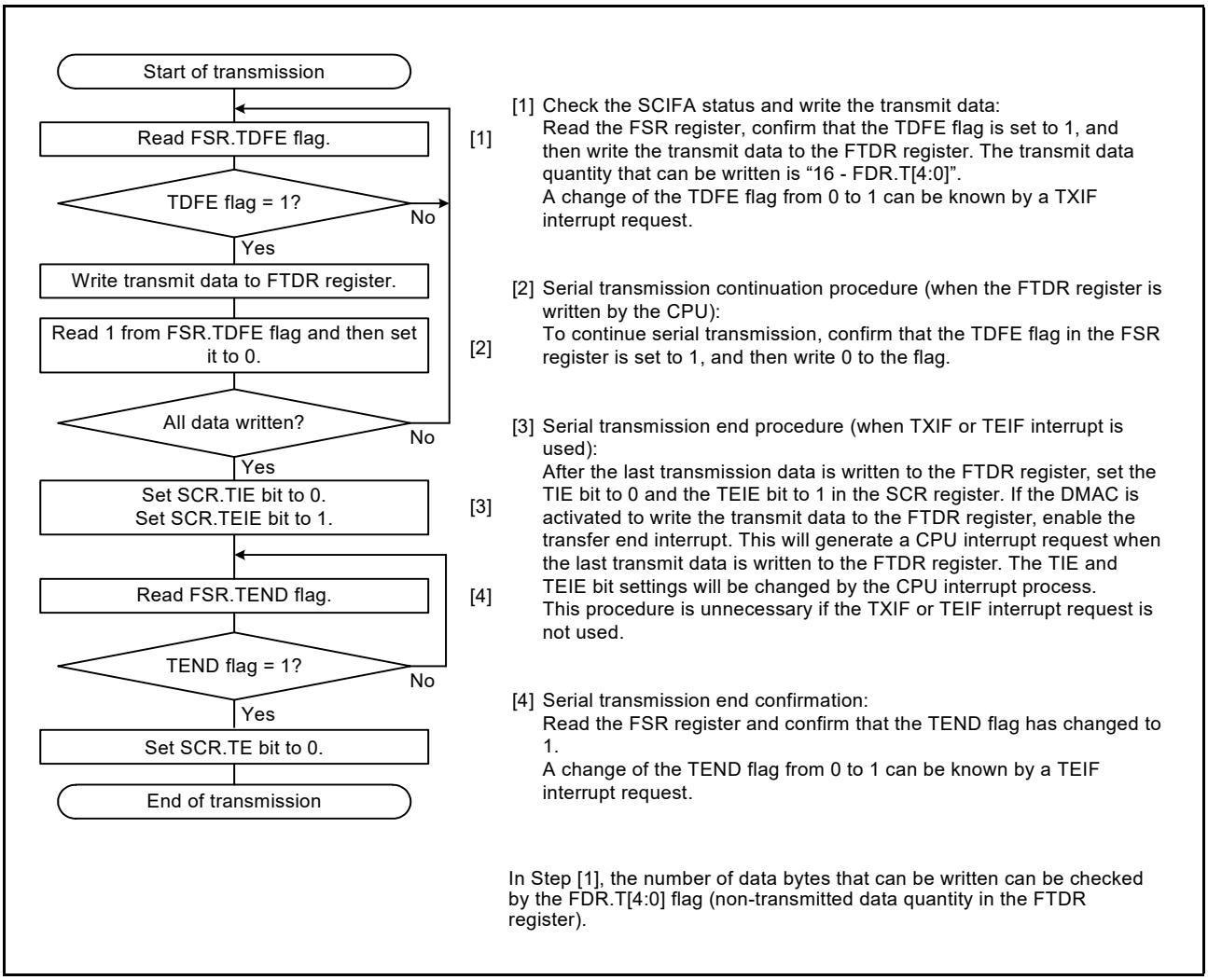
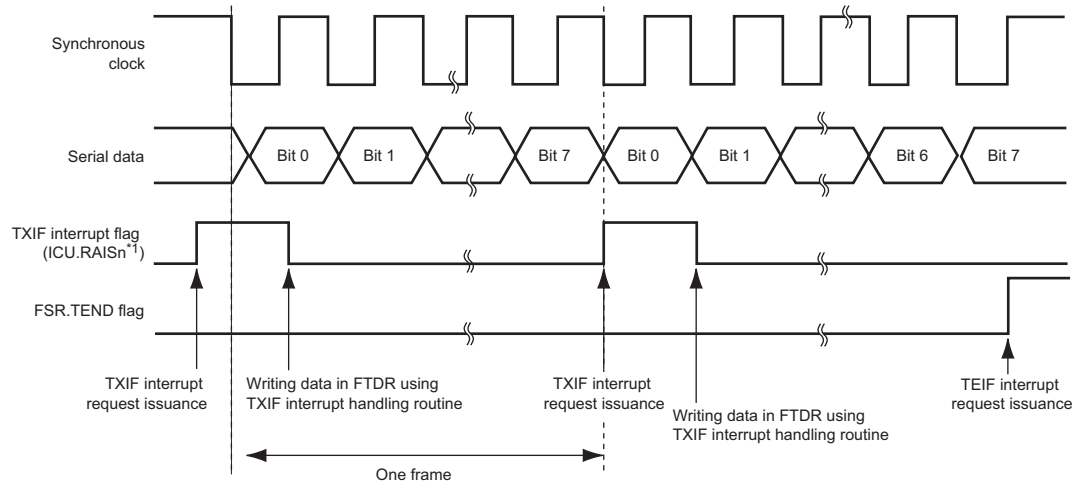


Figure 25.13 Sample Flowchart for Transmitting Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial transmission as described below.

1. When data is written into the transmit FIFO data register (FTDR) by the TXIF interrupt processing routine, the SCIFA transfers the data from the FTDR register to the transmit shift register (TSR) and starts transmission. Confirm that the TDFE flag in the serial status register (FSR) is set to 1 before writing transmit data to the FTDR register. The number of data bytes that can be written is “16 minus the specified number of non-transmitted data units”.
2. When data is transferred from the FTDR register to the TSR register and transmission is started, consecutive transmit operations are performed until there is no transmit data left in the FTDR register. When the number of transmit data bytes in the FTDR register becomes equal to or less than the transmission trigger number set in the FIFO control register (FCR) or FIFO trigger control register (FTCR), the TDFE flag in the FSR register is set. If the TIE bit in the serial control register (SCR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXIF) request is generated.
   If clock output mode is selected, the SCIFA outputs eight synchronous clock pulses. If an external clock source is selected, the SCIFA outputs data in synchronization with the input clock. Data is output from the TXD pin in order from the LSB (b0) to the MSB (b7) (when LSB-first transfer is selected).
3. The SCIFA checks the transmit data of the FTDR register at the timing for sending the MSB (bit 7). If data is present, the data is transferred from the FTDR register to the TSR register, and then serial transmission of the next frame is started. If there is no data, the TXD pin holds the output level of the last data after the TEND flag in the FSR register is set to 1 and the MSB (bit 7) is output.
4. After the end of serial transmission, the SCK pin is held in the high state.

Figure 25.14 shows an example of SCIFA transmit operation in clock synchronous mode.

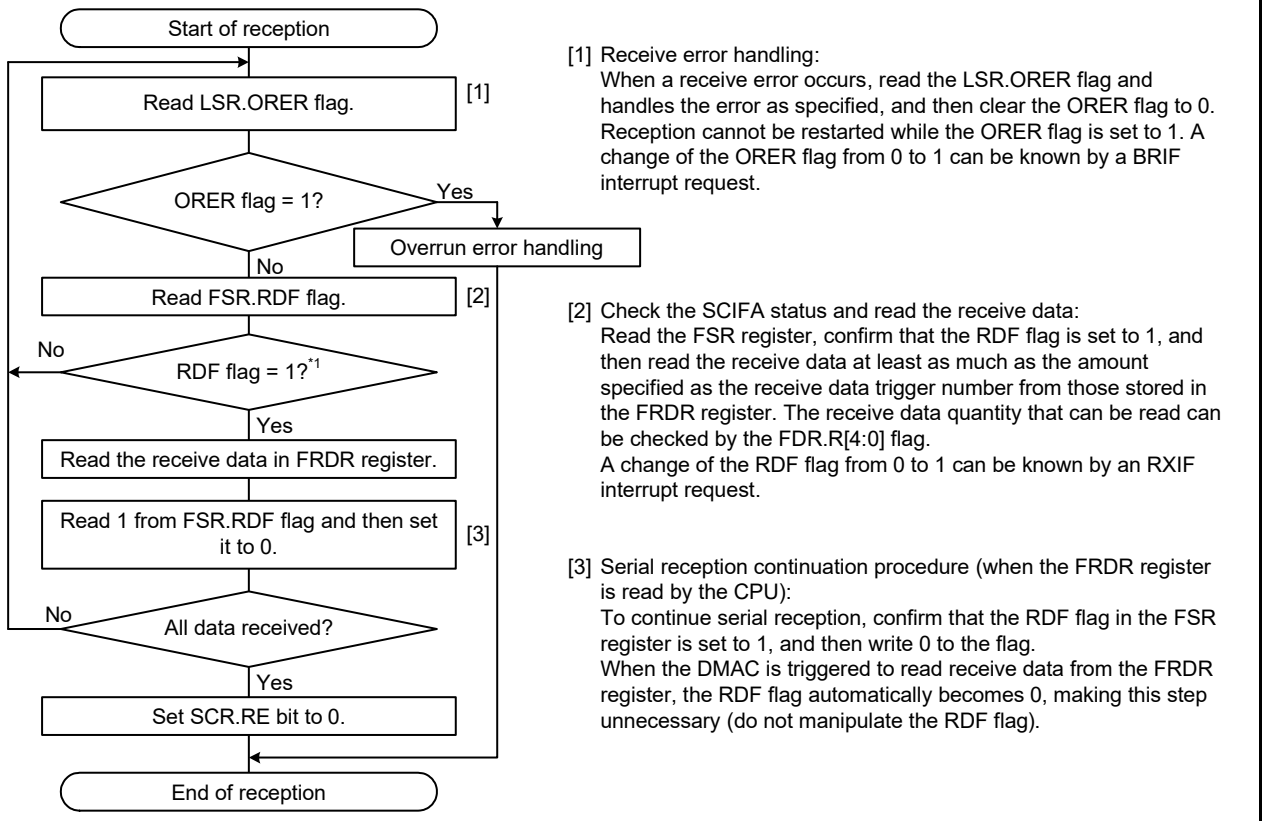


Note 1. For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

Figure 25.14 Example of SCIFA Transmit Operation in Clock Synchronous Mode (when LSB-First Transfer is Selected)

- Receiving Serial Data (in Clock Synchronous Mode)

Figure 25.15 shows sample flowcharts for receiving serial data in clock synchronous mode.
Follow the procedure given below for serial data reception after enabling the SCIFA for reception. When switching from asynchronous mode to clock synchronous mode without SCIFA initialization, make sure that the ORER, PER, and FER flags in the line status register (LSR) are cleared to 0.



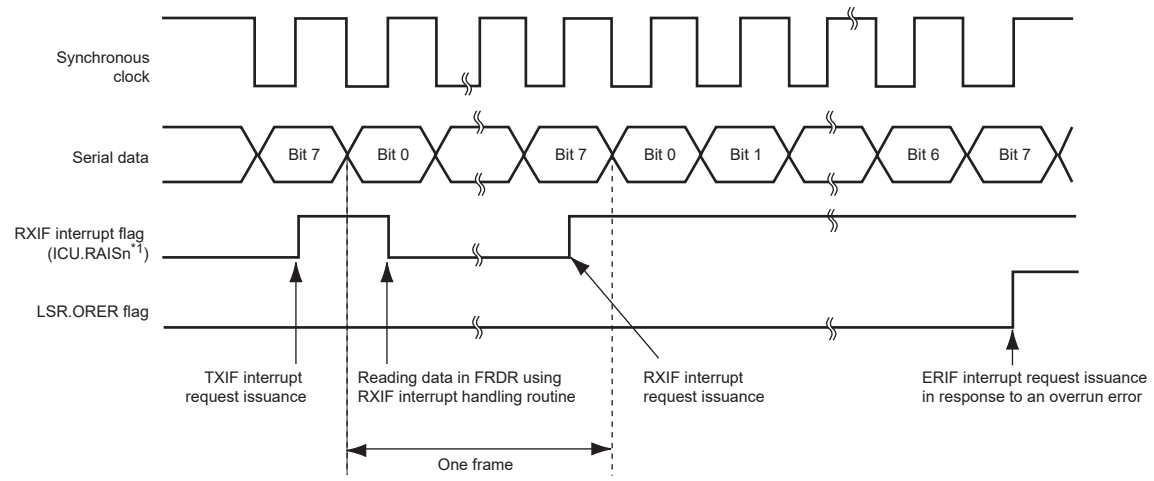
Note 1. The number of all the received data units should be an integral multiple of the receive data trigger number. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 25.15 Sample Flowchart for Receiving Serial Data in Clock Synchronous Mode

In clock synchronous mode, the SCIFA performs serial reception as described below.

1. The SCIFA synchronizes with the synchronous clock input or output and starts reception.
2. Receive data is stored into the receive shift register (RSR) in order from the LSB to the MSB (when LSB-first transfer is selected). After receiving the data, the SCIFA checks whether the receive data can be transferred from the RSR register to the FRDR register. If data can be transferred, the SCIFA stores the received data in the FRDR register. If an overrun error is detected during the error check, further reception is not performed.
3. After the received data units equaling or exceeding the specified reception trigger number are stored in the FRDR register and the RDF flag is set to 1, a receive-data-full interrupt (RXIF) request is generated when the RIE bit in the serial control register (SCR) is set to 1. When the ORER flag in the line status register (LSR) is set to 1 and the RIE or REIE bit in the SCR register is also set to 1, a break interrupt (BRIF) request is generated.

Figure 25.16 shows an example of SCIFA receive operation in clock synchronous mode.



Note 1. For the corresponding interrupt number, see section 12, Interrupt Controller (ICUA).

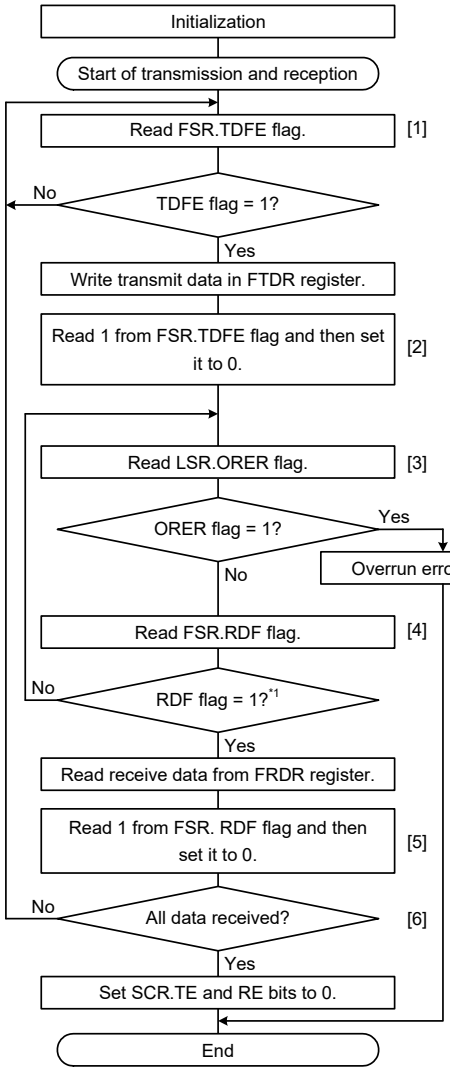
Figure 25.16 Example of SCIFA Receive Operation (when LSB-First Transfer is Selected)

- Transmitting and Receiving Serial Data Simultaneously (in Clock Synchronous Mode)

Figure 25.17 shows a sample flowchart for transmitting and receiving serial data simultaneously in clock synchronous mode.

In simultaneous transmission/reception of serial data, number of receive data = number of transmit data = number of transmit data to be written to the FTDR register.

Follow the procedure given below for the simultaneous transmission/reception of serial data, after enabling the SCIFA for transmission/reception.



[1] Check the SCIFA status and write the transmit data:
Read the FSR register, confirm that the TDFE flag is set to 1, and then write the transmit data in the FTDR register.
A change of the TDFE flag from 0 to 1 can be known by a TXIF interrupt request.

[2] Serial transmission continuation procedure (when the FTDR register is written by the CPU):
To continue serial transmission, confirm that the FSR.TDFE flag is set to 1, and then write 0 to the flag.

[3] Receive error handling:
When a receive error occurs, read the LSR.ORER flag and handles the error as specified, and then clear the ORER flag to 0.
Reception cannot be restarted while the ORER flag is set to 1.
A change of the ORER flag from 0 to 1 can be known by a BRIF interrupt request.

[4] Check the SCIFA status and read the receive data:
Read the FSR register, confirm that the RDF flag is set to 1, and then read the receive data stored in the FRDR register.
A change of the RDF flag from 0 to 1 can be known by an RXIF interrupt request.

[5] Serial reception continuation procedure (when the FRDR register is read by the CPU):
To continue serial reception, confirm that the FSR.RDF flag is set to 1, and then write 0 to the flag.
When the DMAC is triggered to read receive data from the FRDR register, the RDF flag automatically becomes 0, making this step unnecessary (do not manipulate the RDF flag).

[6] Serial transmission/reception continuation procedure (supplementary):
To continue serial transmission/reception, read the FRDR register and clear the RDF flag to 0 before the FRDR register overflows and the ORER flag changes to 1. In addition, before the last bit of the last data written to the FTDR register is transmitted, write the data in the FTDR register and clear the TDFE flag to 0.
When the DMAC is triggered to write data to the FTDR register or read data from the FRDR register, the TDFE and RDF flags automatically become 0, making this step unnecessary (do not manipulate the TDFE or RDF flag).

Note: When switching the operation from transmission or reception to the simultaneous transmission and reception, clear the TE and RE bits to 0 and then simultaneously set these bits to 1.

Note 1. The number of all the received data units should be an integral multiple of the receive data trigger number. If not, process returns to Step [1] and enters an infinite loop. In this case, it is necessary to set the timeout processing using a timer.

Figure 25.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode

25.4 Bit Modulation

Using the bit rate modulation, the bit rate can be corrected by skipping the specified number of clock pulses input to the baud rate generator. To correct the bit rate, only the number of clock pulses specified in the MDDR register are enabled among 256 internal clock pulses specified by the CKS[1:0] bits in the SMR register in a way that forms average intervals.

Figure 25.18 shows an example where SERICLK is selected by the CKS[1:0] bits in SMR and BRR and MDDR are set to 0 and 160, respectively, in asynchronous mode. In this example, the cycle of the base clock is evenly corrected (256/160) and the bit rate is also corrected (160/256). Note that skipping an internal clock causes bias and expansion or contraction is generated in the pulse width of the base clock.

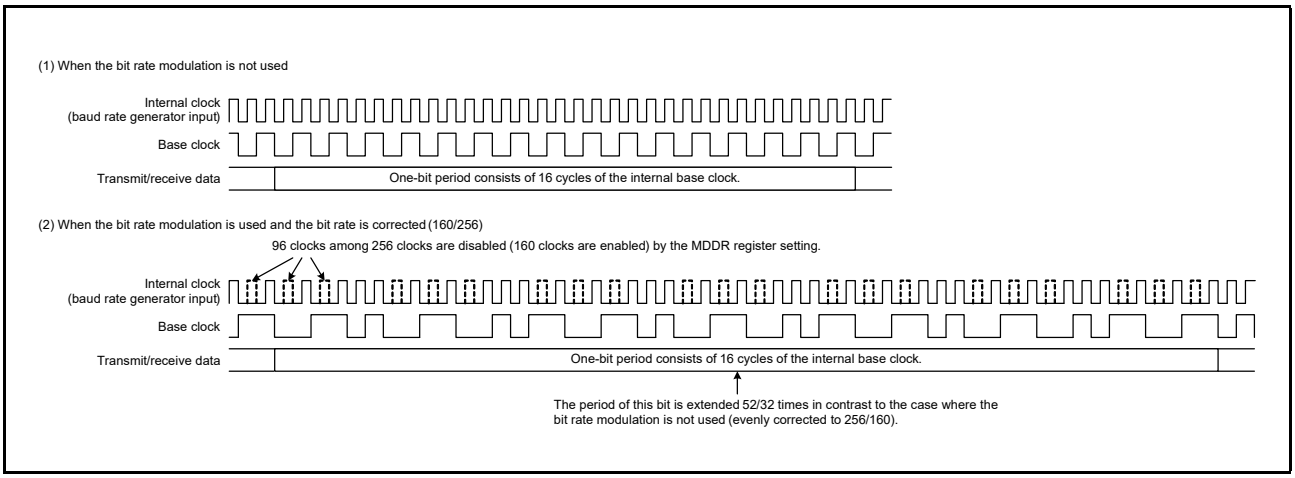


Figure 25.18 Example of Internal Base Clock when Bit Modulation is Used

25.5 Interrupt Sources

The SCIFA has six interrupt sources: transmit-FIFO-data-empty (TXIF), receive-error (ERIF), receive-FIFO-data-full (RXIF), break (BRIF), transmit-end (TEIF), receive-data-ready (DRIF). The TEIF and DRIF interrupts, the ERIF and BRIF interrupts share the same vector numbers, respectively.

Table 25.17 shows the interrupt sources and priority. The interrupt sources can be enabled or disabled using the TIE, RIE, REIE, TEIE bits in the SCR register and are separately input to the interrupt controller.

When the quantity of transmit data written in the FTDR register as a result of transmission is equal to or less than the specified transmission trigger number, the TDFE flag in the serial status register (FSR) is set to 1 and a TXIF interrupt request is generated.

When the data units equaling or exceeding the specified transmission trigger number are stored in the receive FIFO register (FRDR) and the RDF flag in the FSR register is set to 1, a receive data full interrupt (RXIF) request is generated.

When the quantity of received data in the FRDR register is below the specified reception trigger number and no next data has been received yet even after the period of 15 ETUs elapsed*1 from the last stop bit, the DR flag in the FSR register is set to 1 and a receive data ready interrupt (DRIF) request is generated. In clock synchronous mode, a DRIF interrupt request is not generated.

When the BRK flag in the FSR register or the ORER flag in the LSR register is set to 1, a BRIF interrupt request is issued.

When the ER flag in the FSR register is set to 1, an ERIF interrupt request is issued.

When the TEND flag in the FSR register is set to 1, a TEIF interrupt request is issued.

When the RIE bit is cleared to 0 and the REIE bit in the SCR register is set to 1, an ERIF and BRIF interrupt requests are issued but an RXIF and a DRIF interrupt requests are not.

An TXIF interrupt indicates that transmit data can be written and an RXIF interrupt indicates that receive data is stored in the FRDR register.

Note 1. It is equivalent to 1 and half frames of 8-bit format with one stop bit (ETU: Element Time Unit).

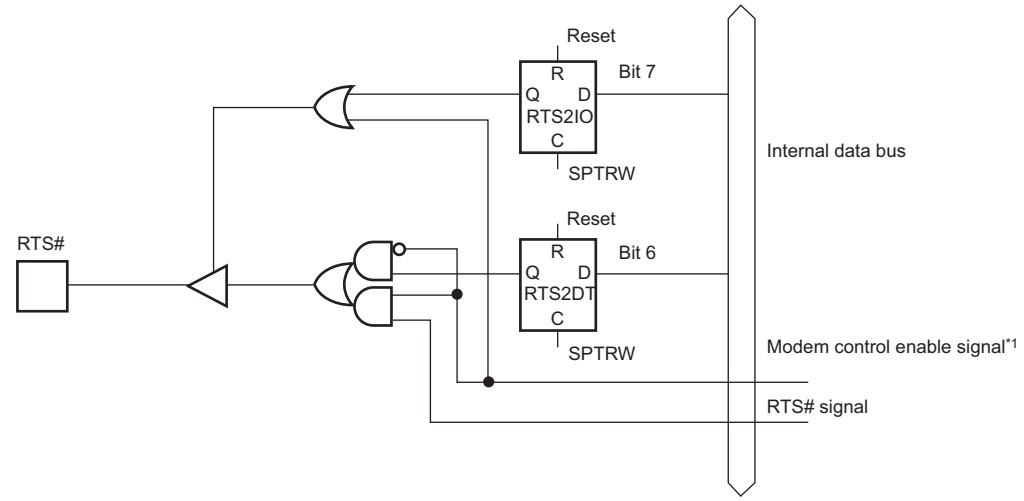
Table 25.17 SCIFA Interrupt Sources

Name	Level/ Edge	Interrupt Source	Interrupt Enable Bit	DMAC Activation	Priority
BRIF	Level	Interrupt caused by break (BRK) or overrun (ORER).	RIE or REIE	Impossible	High
ERIF	Level	Interrupt caused by framing or parity (ER).	RIE or REIE	Impossible	
RXIF	Level	Interrupt caused by receive FIFO data full (RDF).	RIE	Possible	
TXIF	Level	Interrupt caused by transmit FIFO data empty (TDFE).	TIE	Possible	
TEIF	Level	Interrupt caused by transmit end (TEND).	TEIE	Impossible	
DRIF	Level	Interrupt caused by receive data ready (DR).	RIE	Impossible	Low

Note: The TEIF and DRIF interrupts share the same vector number.

25.6 Serial Port Register (SPTR) and SCIFA-Related Pins

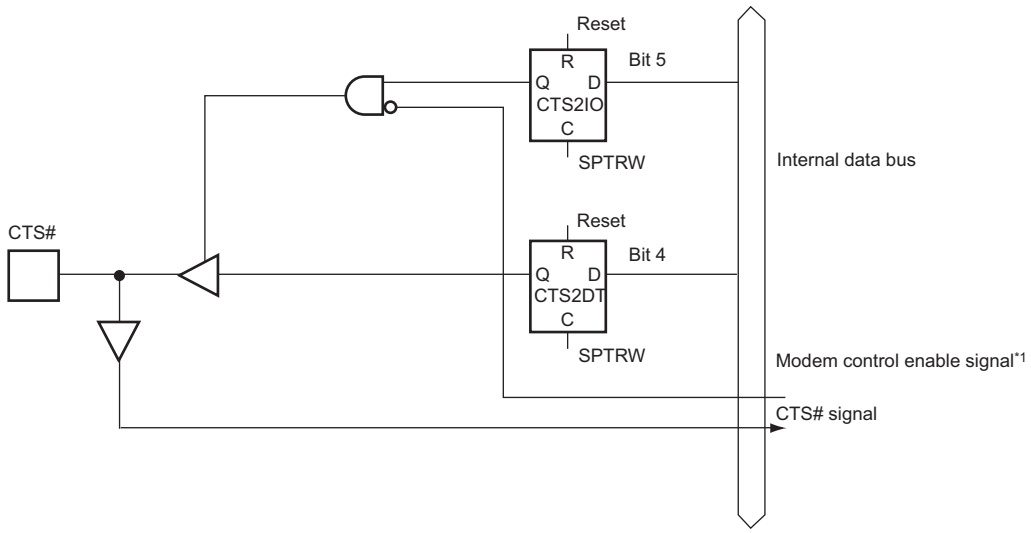
Figure 25.19 to Figure 25.22 show the relationships between the SPTR register and the SCIFA-related pins.



SPTRW: SPTR write

Note 1. These signals allow the RTS# pin to enter the modem control state with the MCE bits in FCR.

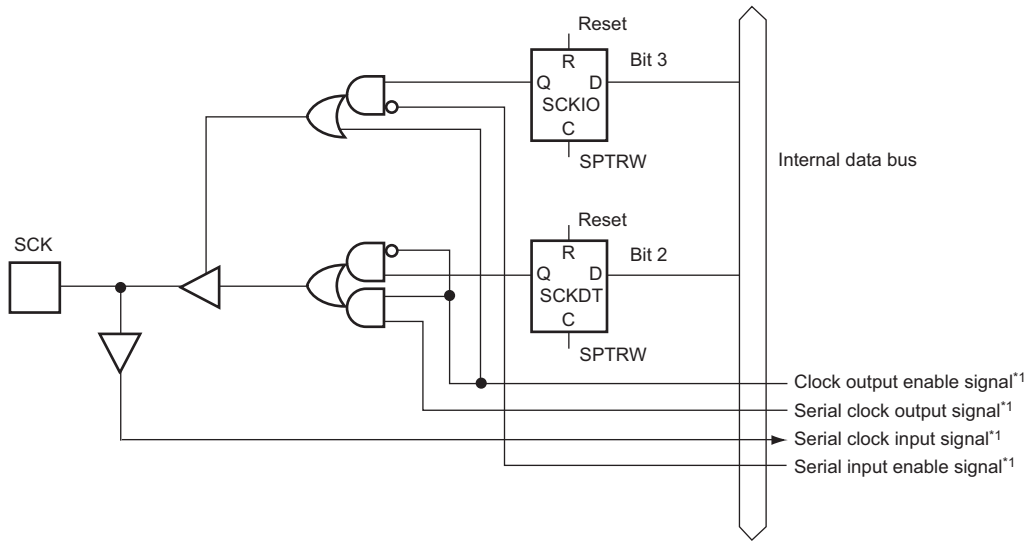
Figure 25.19 RTS2IO Bit and RTS2DT Bit in the SPTR Register, and RTS# Pin



SPTRW: SPTR write

Note 1. These signals allow the CTS# pin to enter the modem control state with the MCE bits in FCR.

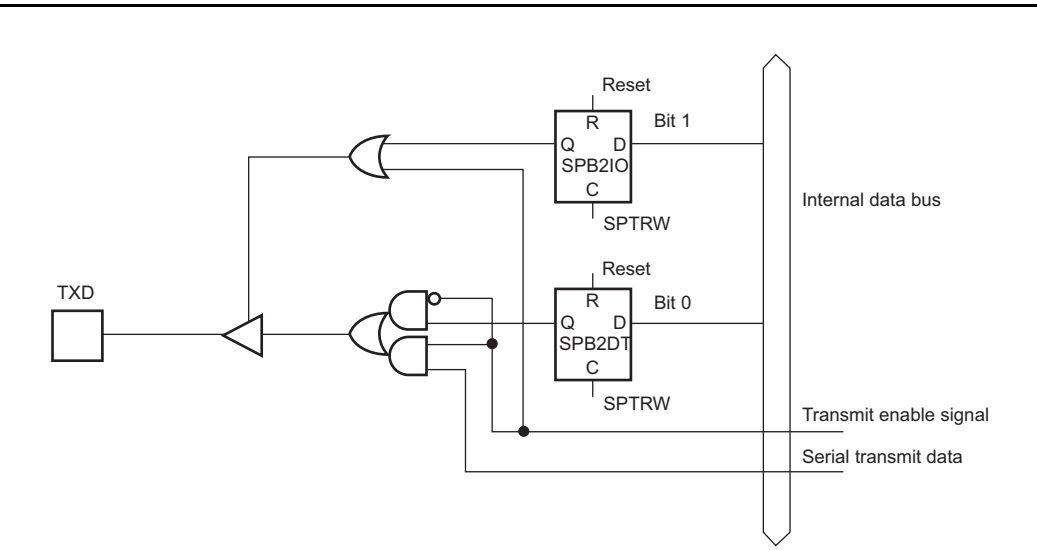
Figure 25.20 CTS2IO Bit and CTS2DT Bit in the SPTR Register, and CTS# Pin



SPTRW: SPTR write

Note 1. These signals control the SCK pin according to the settings of the CM bit in SMR and CKE[1:0] bits in SCR.

Figure 25.21 SCKIO Bit and SCKDT Bit in the SPTR Register, and SCK Pin



SPTRW: SPTR write

Figure 25.22 SPB2IO Bit and SPB2DT Bit in the SPTR Register, and TXD Pin

25.7 Noise Cancellation

Figure 25.23 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of a two-stage flip-flop circuits and a match detection circuit. When the input signals of the noise filter and the output signals of the two-stage flip-flop circuits completely match, the matched level is conveyed as an internal signal. Unless otherwise matched, the previous value is retained. (When the levels sampled on three consecutive cycles of the sampling clock of the noise filter match, the signal is considered valid. If three consecutive sampled values do not match, the signal is considered to be noise rather than a received signal).

In asynchronous mode, the noise cancellation can be applied to the receive signal input to the RXDn pin. The receive level of the RXDn pin is taken in the flip-flop circuit of the noise filter on the base clock (the clock with a frequency 16, 8, or 4 times the transfer rate*1).

If the base clock is stopped once with the noise filter enabled and then the base clock input is restarted again, the noise filter operation resumes from the state where the clock was stopped. When SCR.RE is set to 0 during input of the base clock, the noise filter outputs 0 as the internal RxDn signal. The internal match detector continues operating even while operations for reception are stopped, and the result from the last time previous consecutive samples matched is output at the same time as operations for reception are resumed.

Note 1. A frequency 16 times bit rate when the SEMR.ABCS0 bit and the SEMR.BGDM bit are both 0, a frequency 8 times bit rate when either the SEMR.ABCS0 bit or the SEMR.BGDM bit is 1, and a frequency 4 times bit rate when the SEMR.ABCS0 bit and the SEMR.BGDM bit are both 1.

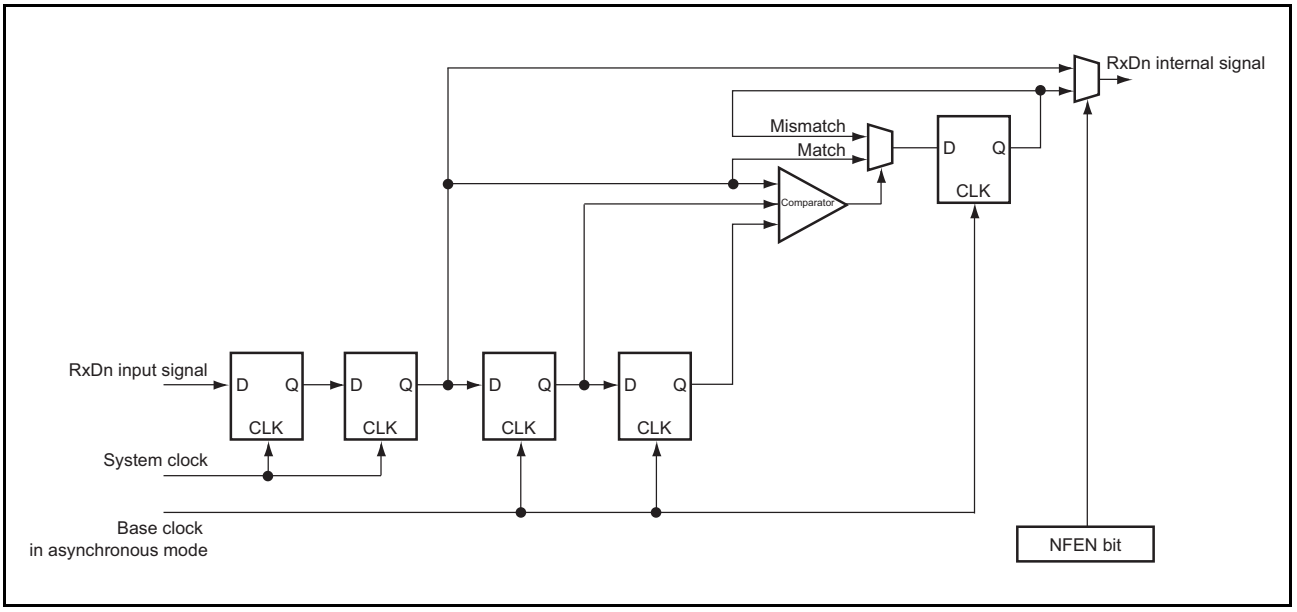


Figure 25.23 Block Diagram of Digital Noise Filter Circuit

25.8 Usage Notes

The following is the notes on using the SCIFA.

25.8.1 FTDR Register Writing and TDFE Flag

The TDFE flag in the serial status register (FSR) is set when the number of transmit data bytes written in the transmit FIFO data register (FTDR) has fallen below the transmission trigger number set by bits TTRG[1:0] in the FIFO control register (FCR) or bits TFTC[4:0] in the FIFO trigger control register (FTCR). After the TDFE flag is set, transmit data up to the number of empty bytes in the FTDR register can be written, allowing efficient continuous transmission.
However, if the number of data bytes written in the FTDR register is equal to or less than the specified transmission trigger number, the TDFE flag will be set to 1 again even after being read as 1 and cleared to 0.
In case of data writing to the FTDR register by the DMAC, the FSR.TDFE flag remains 1 after the DMAC transfer completes. However, DMAC transfer is possible regardless of the FSR.TDFE flag setting.
The number of transmit data bytes in the FTDR register can be checked by the 8 higher-order bits of the FIFO data count register (FDR).

25.8.2 FRDR Register Reading and RDF Flag

The RDF flag in the serial status register (FSR) is set when the number of receive data bytes in the receive FIFO data register (FRDR) has become equal to or greater than the reception trigger number set by bits RTRG[1:0] in the FIFO control register (FCR) or bits RFTC[4:0] in the FIFO trigger control register (FTCR). After the RDF flag is set, receive data equivalent to the trigger number can be read from the FRDR register, allowing efficient continuous reception.
However, if the number of data bytes in the FRDR register exceeds the reception trigger number, the RDF flag will be set to 1 again even after being read as 1 and cleared to 0.
In case of data reading from the FRDR register by the DMAC, the FSR.RDF flag remains 1 after the DMAC transfer completes. However, DMAC transfer is possible regardless of the FSR.RDF flag setting.
The number of receive data bytes in the FRDR register can be checked by the 8 lower-order bits of the FIFO data count register (FDR).

25.8.3 Break Detection and Processing

When a framing error (FER) is detected, a break signal can be detected by reading the RXD pin value directly. In a break, the input from the RXD pin becomes all low. Therefore, the FER flag in the serial status register (FSR) is set to 1 and the parity error flag (PER) may also be set to 1.
Upon detection of a break signal, the SCIFA stops the received data transfer to the FRDR register but continues the receive operation.

25.8.4 Writing to the SPTR Register

b6, b4, b2, and b0 of the SPTR register respectively indicate the input status of their corresponding pins. (See the description of each bit of section 25.2.12, Serial Port Register (SPTR) for details.)
Writings to these bits in 1-bit unit are handled as read-modify-write, which may lead to undesired values to be written. To avoid this, when modifying the SPB2DT or SPB2IO bit, for example, write the other bit (the bit used in combination) at the same time.

25.8.5 Break Signal Transmission

The output signal from the TXD pin is determined by the SPB2IO bit and the SPB2DT bit in the serial port register (SPTR). The break signal can be sent by using these bits.
The TXD pin does not function as a transmit data output pin during the period from when the SCIFA is initialized to when the TE bit in the SCR register is set to 1 (transmission possible). The TXD pin status during this period is replaced by the SPB2DT bit value. Therefore, the SPB2IO and SPB2DT bits in the SPTR register must have been set to 1 (high output) at first (mark (high) status).

To transmit the break signal during serial transmission, set the SPB2IO bit in the SPTR register to 1, clear the SPB2DT bit to 0 (specify a low level), and then clear the TE bit in the SCR register to 0 (transmission stop). Clearing the TE bit to 0 initializes the transmitter regardless of the current transmission status, and outputs a low level from the TXD pin.

25.8.6 Receive Data Sampling Timing and Receive Margin in Asynchronous Mode

The SCIFA operates on a base clock with a frequency 16 times the transfer rate*1. In reception, the SCIFA internally latches the received data at the rising edge of the eighth base clock pulse*1. The timing is shown in Figure 25.24.

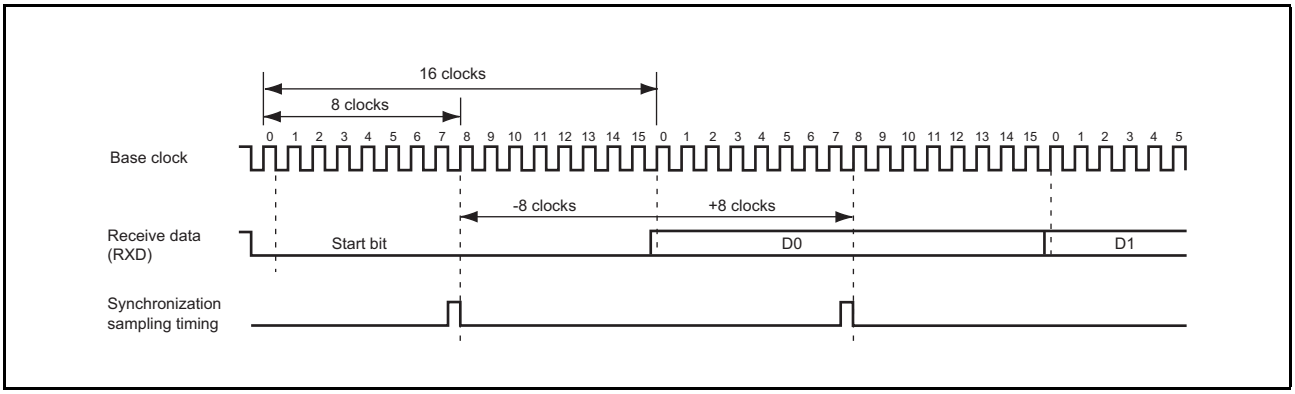


Figure 25.24 Receive Data Sampling Timing in Asynchronous Mode

Note 1. This is an example when the SEMR.ABCS0 bit is 0. When the ABCS0 bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock. The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

Equation 1:

$$M = \{(0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\} \times 100[\%]$$

Where: M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation 1, if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation 2.

Equation 2:

When D = 0.5 and F = 0:
M = (0.5 - 1/(2 ×16)) ×100% = 46.875%

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

25.8.7 Note on FER Flag and PER Flag in Serial Status Register (FSR)

The FER flag and PER flag in the serial status register (FSR) are status flags that apply to next entry to be read from the receive FIFO data register (FRDR). After the CPU reads the receive FIFO data register, the flags of framing errors and parity errors in the receive data will be cleared. To check the received data for the states of framing errors and parity errors, only read the receive FIFO data register after reading the serial status register.

25.8.8 Notes on External Clock Input in Clock Synchronous Mode

Before setting the TE and RE bits in the serial control register (SCR) to 1, wait for four or more cycles of the peripheral operating clock after the external clock (SCK) is changed from 0 (low) to 1 (high). To input the external clock (SCK) (to start communication), wait for one or more cycles of the external clock after the TE and RE bits in the SCR register are set to 1.

25.8.9 Module Standby Mode Setting

SCIFA operation can be disabled or enabled using the standby control register. As the initial setting, the SCIFA operation is halted. Register access is enabled by clearing module standby mode. For details, refer to section 9, Low-Power Consumption Function.

25.8.10 Notes on Operation for Reception when an Internal Clock is Selected in Clock Synchronous Mode

When an internal clock is selected as the clock for reception in clock-synchronous mode, if the number of data stored through the receive FIFO data register (FRDR) becomes equal to or greater than the specified reception trigger number, the RDF flag is set, the RXIF interrupt request is generated and, at the same time, output of the synchronizing clock and reception of serial data are stopped. Once the number of data are again less than the specified reception trigger number, output of the synchronizing clock and the reception of serial data are restarted. In addition, if an internal clock is selected for reception in clock synchronous mode, the ORER flag is not set to 1 since no overrun occurs. Accordingly, overruns (indicated by the ORER flag) cannot be used as a BRIF interrupt source.

25.8.11 Notes on Initialization of the SCIFA

In the SCIFA initialization process, clearing of the TE and RE bits of the serial control register (SCR) should be taken place at the same time or the clearing of the RE bit should precede that of the TE bit. This is because clearing the TE bit to 0 while the RE bit remains 1 enables data reception and may lead to start of unintended data reception.

26. I²C Bus Interface (RIICa)

This LSI has one I2C bus interface (RIIC module).
The RIIC module conforms with and provides a subset of the NXP I2C bus (Inter-IC-Bus) interface functions.

26.1 Overview

Table 26.1 lists the specifications of the RIIC, Figure 26.1 shows a block diagram of the RIIC, and Figure 26.2 shows an example of I/O pin connections to external circuits (I2C bus configuration example). Table 26.2 lists the I/O pins of the RIIC.

Table 26.1 RIIC Specifications (1 / 2)

Item	Description
Communications format	• I2C bus format • Master mode or slave mode selectable • Automatic securing of the various setup times, hold times, and bus-free times for the transfer rate
Transfer rate	Up to 400 kbps: fast mode
SCL clock	For master operation, the duty cycle of the SCL clock is selectable in the range from 4% to 96%.
Issuing and detecting conditions	Start, restart, and stop conditions are automatically generated. Start conditions (including restart conditions) and stop conditions are detectable.
Slave address	• Up to three different slave addresses can be set. • 7-bit and 10-bit address formats are supported (along with the use of both at once). • General call addresses and device ID addresses are detectable.
Acknowledgement	• For transmission, the acknowledge bit is automatically loaded. Transfer of the next data for transmission can be automatically suspended on detection of a not-acknowledge bit. • For reception, the acknowledge bit is automatically transmitted. If a wait between the eighth and ninth clock cycles has been selected, software control of the value in the acknowledge field in response to the received value is possible.
Wait function	• In reception, the following periods of waiting can be obtained by holding the SCL clock at the low level: Waiting between the eighth and ninth clock cycles Waiting between the ninth clock cycle and the first clock cycle of the next transfer
SDA output delay function	Timing of the output of transmitted data, including the acknowledge bit, can be delayed.
Arbitration	• Synchronized operation of multiple SCL clocks to avoid contention with other masters (providing support for multiple masters). • Loss in arbitration as a master Detection of mismatches of state between the SDA signal and another signal on the SDA line when a start condition is issued. Detection of a start condition being issued while in the bus busy state. Detection of mismatches of state between the data being transmitted and the signal on the SDA line in transmission as a master. • Loss in arbitration on NACK transmission Detection of mismatches of state between the data being transmitted and the signal on the SDA line in transmission of a not-acknowledge signal. • Loss in arbitration as a slave Detection of mismatches of state between the data being transmitted and the signal on the SDA line in transmission as a slave.
Timeout function	The internal timeout function is capable of detecting long-interval stop of the SCL clock.
Noise cancellation	The interface incorporates digital noise filters for both the SCL and SDA signals, and the width for noise cancellation by the filters is adjustable by software.
Interrupt sources	Four sources: • Error in transfer or occurrence of events Detection of arbitration, NACK, timeout, a start condition including a restart condition, or a stop condition • Receive-data-full (including matching with a slave address) • Transmit-data-empty (including matching with a slave address) • Transmission complete
Low-power consumption function	Module-stop state can be set.

Table 26.1 RIIC Specifications (2 / 2)

Item	Description
RIIC operating modes	• Four modes: Master transmission mode, master reception mode, slave transmission mode, and slave reception mode

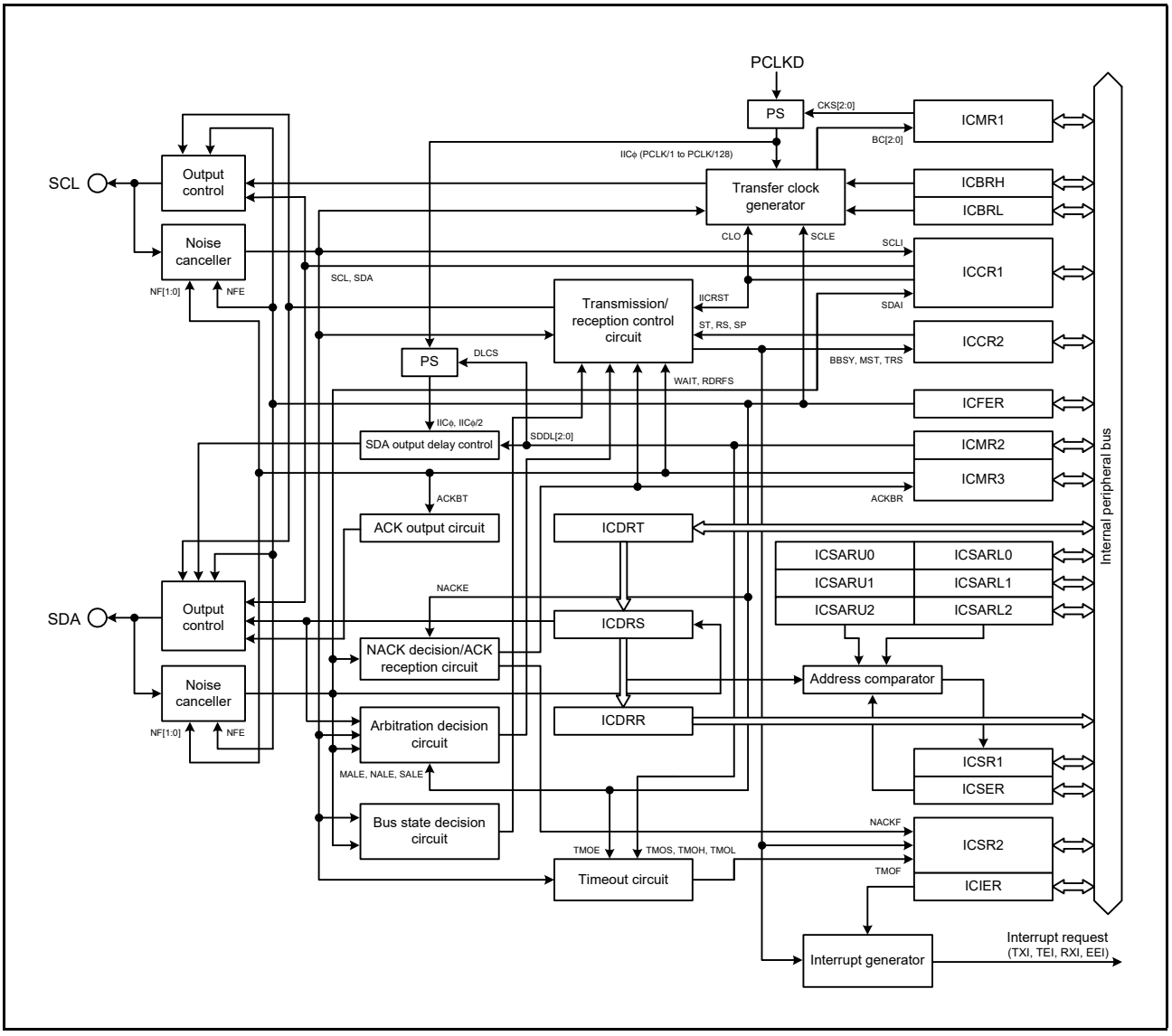


Figure 26.1 RIIC Block Diagram

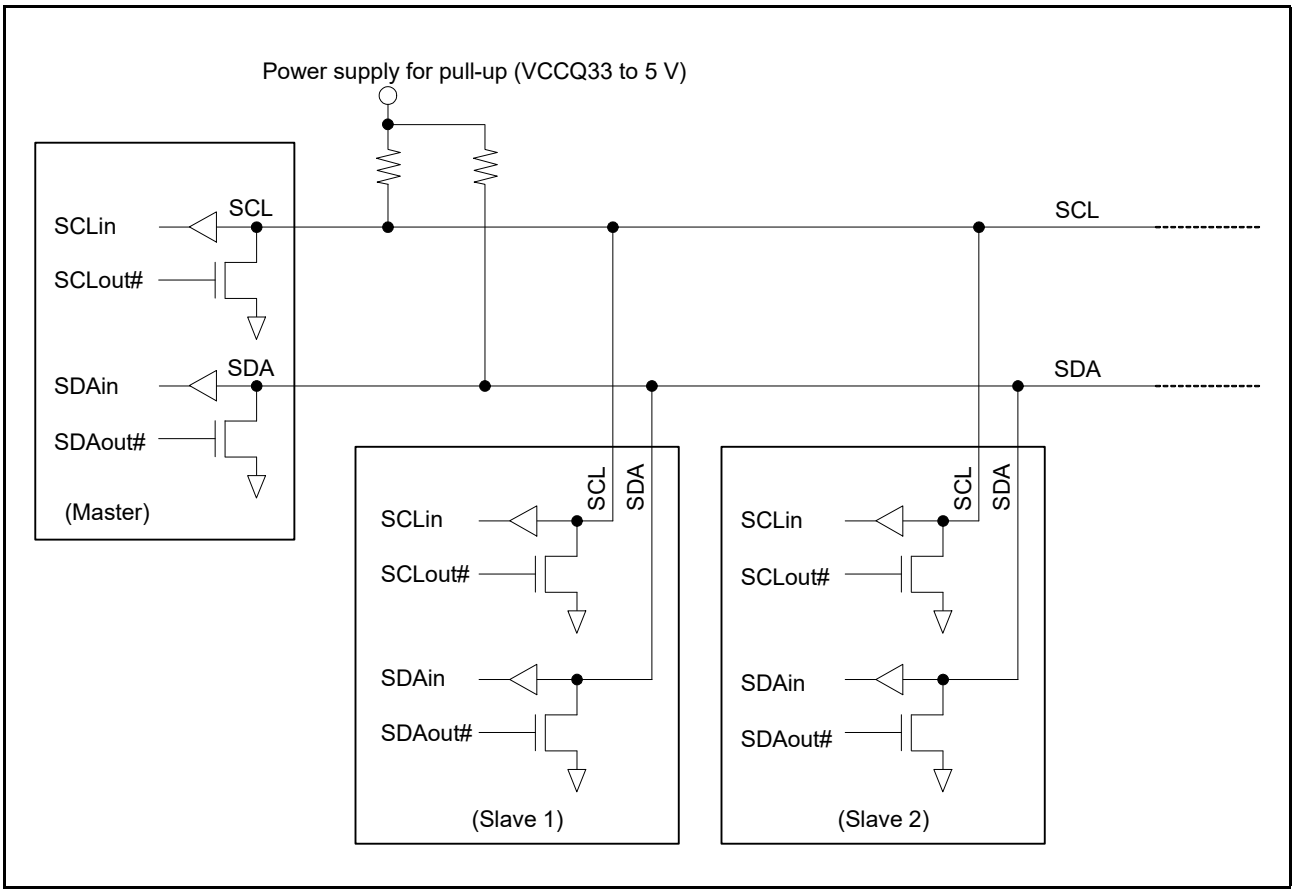


Figure 26.2 Connections to the External Circuit by the I/O Pins (I²C Bus Configuration Example)

The input level of the signals for RIIC is CMOS.

Table 26.2 Pin Configuration

Channel	Pin Name	I/O	Function
RIIC1	SCL1	I/O	RIIC1 serial clock I/O pin
	SDA1	I/O	RIIC1 serial data I/O pin

26.2 Register Descriptions

26.2.1 I²C Bus Control Register 1 (ICCR1)

The ICCR1 register controls the SDA and SCL signals output by the RIIC.

Address(es): RIIC1.ICCR1 A008 0940h

	b7	b6	b5	b4	b3	b2	b1	b0
	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
Value after reset:	0	0	0	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	SDAI	SDA Line Monitor	0: SDA line is low. 1: SDA line is high.	R
b1	SCLI	SCL Line Monitor	0: SCL line is low. 1: SCL line is high.	R
b2	SDAO	SDA Output Control/Monitor	• Read: 0: SDA pin output is low. 1: SDA pin output is high. • Write: 0: SDA pin output is set to low. 1: SDA pin output is changed to high impedance. (High level output is achieved through an external pull-up resistor.)	R/W
b3	SCLO	SCL Output Control/Monitor	• Read: 0: SCL pin output is low. 1: SCL pin output is high. • Write: 0: SCL pin output is set to low. 1: SCL pin output is changed to high impedance. (High level output is achieved through an external pull-up resistor.)	R/W
b4	SOWP	SCLO/SDAO Write Protect	0: Bits SCLO and SDAO can be written. 1: Bits SCLO and SDAO are protected. (This bit is always read as 1.)	R/W
b5	CLO	Extra SCL Clock Cycle Output	0: Does not output an extra SCL clock cycle (default). 1: Outputs an extra SCL clock cycle. (The CLO bit is cleared automatically after one clock cycle is output.)	R/W
b6	IICRST	I²C Bus Interface Internal Reset	0: Releases the RIIC reset or internal reset. 1: Initiates the RIIC reset or internal reset. (Clears the bit counter and the SCL/SDA output latch)	R/W
b7	ICE	I²C Bus Interface Enable	0: Disable (SCL and SDA pins in inactive state) 1: Enable (SCL and SDA pins in active state) (Combined with the IICRST bit to select either RIIC or internal reset.)	R/W

SDAO Bit (SDA Output Control/Monitor) and SCLO Bit (SCL Output Control/Monitor)

These bits are used to directly control the SDA and SCL signals output from the RIIC.

When writing to these bits, also write 0 to the SOWP bit.

The result of setting these bits is input to the RIIC via the input buffer. When slave mode is selected, a start condition may be detected and the bus may be released depending on the bit settings.

Do not rewrite these bits during a start condition, stop condition, restart condition, or during transmission or reception. Operation after rewriting under the above conditions is not guaranteed.

When reading these bits, the state of signals output from the RIIC can be read.

CLO Bit (Extra SCL Clock Cycle Output)

This bit is used to output an extra SCL clock cycle for debugging or error processing.
Normally, set the bit to 0. Setting the bit to 1 in a normal communication state causes a communication error.
For details on this function, refer to section 26.11.2, Extra SCL Clock Cycle Output Function.

IICRST Bit (I²C Bus Interface Internal Reset)

This bit is used to reset the internal states of the RIIC.
Setting this bit to 1 initiates an RIIC reset or internal reset.
Whether an RIIC reset or internal reset is initiated is determined according to the combination with the ICE bit. Table 26.3 lists the resets of the RIIC.
The RIIC reset resets all registers and internal states of the RIIC, and the internal reset resets the bit counter (ICMR1.BC[2:0] bits), the I²C bus shift register (ICDRS), and the I²C bus status registers (ICSR1 and ICSR2) as well as the internal states of the RIIC. For the reset conditions for each register, refer to section 26.13, Resets and Register and Function States When Issuing Each Condition.
An internal reset initiated with the IICRST bit set to 1 during operation (with the ICE bit set to 1) resets the internal states of the RIIC without initializing the port settings and the control and setting registers of the RIIC when the bus or RIIC hangs up due to a communication error.
If the RIIC hangs up in a low level output state, resetting the internal states cancels the low level output state and releases the bus with the SCL pin and SDA pin at a high impedance.

Note: If an internal reset is initiated using the IICRST bit for a bus hang-up occurred during communication with the master device in slave mode, the states may become different between the slave device and the master device (due to the difference in the bit counter information). For this reason, do not initiate an internal reset in slave mode, but initiate restoration processing from the master device. If an internal reset is necessary because the RIIC hangs up with the SCL line in a low level output state in slave mode, initiate an internal reset and then issue a restart condition from the master device or resume communication from the start condition issuance after issuing a stop condition. If communication is restarted by initiating a reset solely in the slave device without issuing a start condition or restart condition from the master device, synchronization will be lost because the master and slave devices operate asynchronously.

Table 26.3 RIIC Resets

IICRST	ICE	State	Specifications
1	0	RIIC reset	Resets all registers and internal states of the RIIC.
	1	Internal reset	Resets the ICMR1.BC[2:0] bits, the ICSR1, ICSR2, and ICDRS registers, and the internal states of the RIIC.

ICE Bit (I²C Bus Interface Enable)

This bit selects the active or inactive state of the SCL and SDA pins. It can also be combined with the IICRST bit to initiate two types of resets. See Table 26.3, RIIC Resets, for the types of resets.
Set the ICE bit to 1 when using the RIIC. The SCL and SDA pins are placed in the active state when the ICE bit is set to 1.
Set the ICE bit to 0 when the RIIC is not to be used. The SCL and SDA pins are placed in the inactive state when the ICE bit is set to 0.

26.2.2 I²C Bus Control Register 2 (ICCR2)

The ICCR2 register controls start condition issuance, restart condition issuance, and stop condition issuance.

Address(es): RIIC1.ICCR2 A008 0941h

	b7	b6	b5	b4	b3	b2	b1	b0
	BBSY	MST	TRS	—	SP	RS	ST	—
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b1	ST	Start Condition Issuance Request	0: Does not request to issue a start condition. 1: Requests to issue a start condition.	R/W
b2	RS	Restart Condition Issuance Request	0: Does not request to issue a restart condition. 1: Requests to issue a restart condition.	R/W
b3	SP	Stop Condition Issuance Request	0: Does not request to issue a stop condition. 1: Requests to issue a stop condition.	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	TRS	Transmission/Reception Mode	0: Reception mode 1: Transmission mode	R/W *1
b6	MST	Master/Slave Mode	0: Slave mode 1: Master mode	R/W *1
b7	BBSY	Bus Busy Detection Flag	0: The I²C bus is released (bus free state). 1: The I²C bus is occupied (bus busy state).	R

Note 1. When the ICMR1.MTWP bit is set to 1, the MST and TRS bits can be written to.

ST Bit (Start Condition Issuance Request)

This bit is used to request transition to master mode and issuance of a start condition.

When this bit is set to 1 to request to issue a start condition, a start condition is issued when the BBSY flag is set to 0 (bus free state).

For details on the start condition issuance, refer to section 26.10, Start Condition/Restart Condition/Stop Condition Issuing Function.

[Setting condition]

- When 1 is written to the ST bit

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the ST bit
- When a start condition has been issued (A start condition is detected)
- When the AL (arbitration-lost) flag in ICSR2 is set to 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: Set the ST bit to 1 (start condition issuance request) when the BBSY flag is set to 0 (bus free state). Note that setting the ST bit to 1 (start condition issuance request) when the BBSY flag is set to 1 (bus busy state) is handled as a start condition issuance error and arbitration may be lost.

RS Bit (Restart Condition Issuance Request)

This bit is used to request that a restart condition be issued in master mode.
When this bit is set to 1 to request to issue a restart condition, a restart condition is issued when the BBSY flag is set to 1 (bus busy state) and the MST bit is set to 1 (master mode).
For details on the restart condition issuance, refer to section 26.10, Start Condition/Restart Condition/Stop Condition Issuing Function.
[Setting condition]

- When 1 is written to the RS bit with the BBSY flag in ICCR2 set to 1 (1 cannot be written with the BBSY flag in ICCR2 set to 0)

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the RS bit
- When a restart condition has been issued (A start condition is detected)
- When the AL (arbitration-lost) flag in ICSR2 is set to 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: Do not set the RS bit to 1 while issuing a stop condition.
Note: We recommend that you issue the restart condition in master transmission mode. If 1 (requests to issue a restart condition) is written to the RS bit in slave mode, the restart condition is not issued but the RS bit remains set to 1. If the operating mode changes to master mode with the bit not being cleared, note that the restart condition may be issued.

SP Bit (Stop Condition Issuance Request)

This bit is used to request that a stop condition be issued in master mode.
When this bit is set to 1 to request to issue a stop condition, a stop condition is issued when the BBSY flag is set to 1 (bus busy state) and the MST bit is set to 1 (master mode).
For details on the stop condition issuance, refer to section 26.10, Start Condition/Restart Condition/Stop Condition Issuing Function.
[Setting condition]

- When 1 is written to the SP bit with both the BBSY flag and the MST bit in ICCR2 set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the SP bit
- When a stop condition has been issued (A stop condition is detected)
- When the AL (arbitration-lost) flag in ICSR2 is set to 1
- When a start condition and a restart condition are detected
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: Writing to the SP bit is not possible while the setting of the BBSY flag is 0 (bus free state).
Note: Do not set the SP bit to 1 while a restart condition is being issued.

TRS Bit (Transmission/Reception Mode)

This bit indicates transmit or reception mode.
The RIIC is in reception mode when the TRS bit is set to 0 and is in transmission mode when the bit is set to 1.
Combination of this bit and the MST bit indicates the operating mode of the RIIC.
The value of TRS bit is automatically changed to 1 for transmission mode or 0 for reception mode by issuing or detection of a start condition and setting of the R/W# bit. Although writing to the TRS bit is possible when the MTWP bit in ICMR1 is set to 1, writing to this bit is not necessary during normal usage.
[Setting conditions] One of the following conditions is satisfied:

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the BBSY flag set to 0 (bus free state) and the ST bit set to 1)
- When a restart condition is issued normally according to the restart condition issuance request (when a restart condition is detected with the RS bit set to 1)
- When the R/W# bit added to the slave address is set to 0 in master mode
- When the address received in slave mode matches the address enabled in ICSER, with the R/W# bit set to 1
- When 1 is written to the TRS bit with the MTWP bit in ICMR1 set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When a stop condition is detected
- The AL (arbitration-lost) flag in ICSR2 being set to 1
- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended
- In slave mode, a match between the received address and the address enabled in ICSER when the value of the received R/W# bit is 0 (including cases where the received address is the general call address)
- In slave mode, a restart condition is detected (a start condition is detected with ICCR2.BBSY = 1 and ICCR2.MST = 0)
- When 0 is written to the TRS bit with the MTWP bit in ICMR1 set to 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

The R/W# bit, which is contained in transmit data, indicates the transmit and receive direction. Data is transferred from the slave device to the master device when the R/W# bit is 1, or from the master device to the slave device when the R/W# bit is 0.

MST Bit (Master/Slave Mode)

This bit indicates master or slave mode.
The RIIC is in slave mode when the MST bit is set to 0 and is in master mode when the bit is set to 1. Combination of this bit and the TRS bit indicates the operating mode of the RIIC.
The value of the MST bit is automatically changed to 1 for master mode or 0 for slave mode by issuing of a start condition and issuing or detection of a stop condition, etc. Although writing to the MST bit is possible when the MTWP bit in ICMR1 is set to 1, writing to this bit is not necessary during normal usage.
[Setting conditions] One of the following conditions is satisfied:

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the BBSY flag set to 0 (bus free state) and the ST bit set to 1)
- When 1 is written to the MST bit with the MTWP bit in ICMR1 set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When a stop condition is detected
- When the AL (arbitration-lost) flag in ICSR2 is set to 1
- When 0 is written to the MST bit with the MTWP bit in ICMR1 set to 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

BBSY Flag (Bus Busy Detection Flag)

The BBSY flag indicates whether the I2C bus is occupied (bus busy state) or released (bus free state).
This bit is set to 1 when the SDA line changes from high to low under the condition of SCL line = high, assuming that a start condition has been issued.
When the SDA line changes from low to high under the condition of SCL line = high, this bit is set to 0 after the bus free time (specified in ICBRL) start condition is not detected, assuming that a stop condition has been issued.
[Setting condition]

- When a start condition is detected

[Clearing conditions] One of the following conditions is satisfied:

- When the bus free time (specified in ICBRL) start condition is not detected after detecting a stop condition
- When 1 is written to the IICRST bit in ICCR1 with the ICE bit in ICCR1 set to 0 (RIIC reset)

26.2.3 I²C Bus Mode Register 1 (ICMR1)

The ICMR1 register specifies the number of bits of the down counter and the reference clock source.

Address(es): RIIC1.ICMR1 A008 0942h

	b7	b6	b5	b4	b3	b2	b1	b0
	MTWP	CKS[2:0]			BCWP	BC[2:0]		
Value after reset:	0	0	0	0	1	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	BC[2:0]	Bit Counter	b2 b0 0 0 0: 9 bits 0 0 1: 2 bits 0 1 0: 3 bits 0 1 1: 4 bits 1 0 0: 5 bits 1 0 1: 6 bits 1 1 0: 7 bits 1 1 1: 8 bits	R/W *1
b3	BCWP	BC Write Protect	0: Enables a value to be written in the BC[2:0] bits. (This bit is read as 1.)	R/W *1
b6 to b4	CKS[2:0]	Internal Reference Clock Selection	Selects the internal reference clock source (IICϕ) for the RIIC. b6 b4 0 0 0: PCLKD/1 clock 0 0 1: PCLKD/2 clock 0 1 0: PCLKD/4 clock 0 1 1: PCLKD/8 clock 1 0 0: PCLKD/16 clock 1 0 1: PCLKD/32 clock 1 1 0: PCLKD/64 clock 1 1 1: PCLKD/128 clock	R/W
b7	MTWP	MST/TRS Write Protect	0: Disables writing to the MST and TRS bits in ICCR2. 1: Enables writing to the MST and TRS bits in ICCR2.	R/W

Note 1. Rewrite the BC[2:0] bits and set the BCWP bit to 0 at the same time.

BC[2:0] Bits (Bit Counter)

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a rising edge on the SCL line. Although these bits are writable and readable, it is not necessary to access these bits under normal conditions.

To write to these bits, specify the number of bits to be transferred plus one (data is transferred with an additional acknowledge bit) between transferred frames when the SCL line is at a low level.

The values of the BC[2:0] bits return to 000b at the end of a data transfer including the acknowledge bit or when a start condition including a restart condition is detected.

26.2.4 I²C Bus Mode Register 2 (ICMR2)

The ICMR2 register specifies various settings regarding the timeout detection function and SDA output delay function.

Address(es): RIIC1.ICMR2 A008 0943h

	b7	b6	b5	b4	b3	b2	b1	b0
	DLCS	SDDL[2:0]			—	TMOH	TMOL	TMOS
Value after reset:	0	0	0	0	0	1	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOS	Timeout Detection Time Selection	0: Long mode is selected. 1: Short mode is selected.	R/W
b1	TMOL	Timeout L Count Control	0: Count is disabled while the SCL line is at a low level. 1: Count is enabled while the SCL line is at a low level.	R/W
b2	TMOH	Timeout H Count Control	0: Count is disabled while the SCL line is at a high level. 1: Count is enabled while the SCL line is at a high level.	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b6 to b4	SDDL[2:0]	SDA Output Delay Counter	• When ICMR2.DLCS = 0 (IICϕ) b6 b4 0 0 0: No output delay 0 0 1: 1 IICϕ cycle 0 1 0: 2 IICϕ cycles 0 1 1: 3 IICϕ cycles 1 0 0: 4 IICϕ cycles 1 0 1: 5 IICϕ cycles 1 1 0: 6 IICϕ cycles 1 1 1: 7 IICϕ cycles • When ICMR2.DLCS = 1 (IICϕ/2) b6 b4 0 0 0: No output delay 0 0 1: 1 or 2 IICϕ cycles 0 1 0: 3 or 4 IICϕ cycles 0 1 1: 5 or 6 IICϕ cycles 1 0 0: 7 or 8 IICϕ cycles 1 0 1: 9 or 10 IICϕ cycles 1 1 0: 11 or 12 IICϕ cycles 1 1 1: 13 or 14 IICϕ cycles	R/W
b7	DLCS	SDA Output Delay Clock Source Selection	0: The internal reference clock (IICϕ) is selected as the clock source of the SDA output delay counter. 1: The internal reference clock divided by 2 (IICϕ/2) is selected as the clock source of the SDA output delay counter.*1	R/W

Note 1. The setting DLCS = 1 (IICϕ/2) only becomes valid when SCL is at the low level. When SCL is at the high level, the setting DLCS = 1 becomes invalid and the clock source becomes the internal reference clock (IICϕ).

TMOS Bit (Timeout Detection Time Selection)

This bit is used to select long mode or short mode for the timeout detection time when the timeout function is enabled (TMOE bit = 1 in ICFER). When this bit is set to 0, long mode is selected. When this bit is set to 1, short mode is selected. In long mode, the timeout detection internal counter functions as a 16 bit-counter. In short mode, the counter functions as a 14 bit-counter. While the SCL line is in the state that enables this counter as specified by bits TMOH and TMOL, the counter counts up in synchronization with the internal reference clock (IICϕ) as a count source.
For details on the timeout function, refer to section 26.11.1, Timeout Function.

TMOL Bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held low when the timeout function is enabled (TMOE bit = 1 in ICFER).

TMOH Bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held high when the timeout function is enabled (TMOE bit = 1 in ICFER).

SDDL[2:0] Bits (SDA Output Delay Counter)

The SDA output can be delayed by the SDDL[2:0] setting. This counter works with the clock source selected by the DLCS bit. The setting of this function can be used for all types of SDA output, including the transmission of the acknowledge bit.

Set the SDA output delay time to meet the I^{2}C bus standard (within the data enable time/acknowledge enable time*1) and to be within 250 ns (SCL-clock low-level period - the data setup time). Note that, if a value outside the standard is set, communication with communication devices may malfunction or it may seemingly become a start condition or stop condition depending on the bus state.

For details on this function, refer to section 26.5, Facility for Delaying SDA Output.

Note 1. Data enable time/acknowledge enable time
900 ns (up to 400 kbps: fast mode [Fm])

26.2.5 I²C Bus Mode Register 3 (ICMR3)

The ICMR3 register specifies the settings for acknowledgement and wait and digital noise filter.

Address(es): RIIC1.ICMR3 A008 0944h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	NF[1:0]	Noise Filter Stage Selection	b1 b0 0 0: Noise of up to one IICϕ cycle is filtered out (single-stage filter). 0 1: Noise of up to two IICϕ cycles is filtered out (2-stage filter). 1 0: Noise of up to three IICϕ cycles is filtered out (3-stage filter). 1 1: Noise of up to four IICϕ cycles is filtered out (4-stage filter).	R/W
b2	ACKBR	Receive Acknowledge	0: A 0 is received as the acknowledge bit (ACK reception). 1: A 1 is received as the acknowledge bit (NACK reception).	R
b3	ACKBT	Transmit Acknowledge	0: A 0 is sent as the acknowledge bit (ACK transmission). 1: A 1 is sent as the acknowledge bit (NACK transmission).	R/W *1
b4	ACKWP	ACKBT Write Protect	0: Modification of the ACKBT bit is disabled. 1: Modification of the ACKBT bit is enabled.	W*1
b5	RDRFS	RDRF Flag Set Timing Selection	0: The RDRF flag is set at the rising edge of the ninth SCL clock cycle. (The SCL line is not held low at the falling edge of the eighth clock cycle.) 1: The RDRF flag is set at the rising edge of the eighth SCL clock cycle. (The SCL line is held low at the falling edge of the eighth clock cycle.) Low-hold is released at the ninth cycle since a value is written to the ACKBT bit.	R/W *2
b6	WAIT	WAIT	0: No WAIT (The period between ninth clock cycle and first clock cycle is not held low.) 1: WAIT (The period between ninth clock cycle and first clock cycle is held low.) Low-hold is released by reading ICDRR.	R/W *2
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

Note 1. Write to the ACKBT bit only while the ACKWP bit is already 1. If it is attempted to write 1 to both the ACKWP and ACKBT bits at the same time, the ACKBT bit will not be set to 1.

Note 2. The WAIT and RDRFS bits are valid only in reception mode. These bits are invalid in transmission mode.

NF[1:0] Bits (Number of Noise Filter Stages Select)

These bits are used to select the number of stages in the digital noise filter.

For details on the digital noise filter function, refer to section 26.6, Digital Noise-Filter Circuits.

ACKBR Bit (Receive Acknowledge)

This bit is used to store the acknowledge bit information received from the receive device in transmission mode.
[Setting condition]

- When 1 (Not Acknowledge) is received as the acknowledge bit with the TRS bit in ICCR2 set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When 0 (Acknowledge) is received as the acknowledge bit with the TRS bit in ICCR2 set to 1
- When 1 is written to the IICRST bit in ICCR1 while the ICE bit in ICCR1 is 0 (RIIC reset)

ACKBT Bit (Transmit Acknowledge)

This bit is used to set the bit to be sent at the acknowledge timing in reception mode.
[Setting condition]

- When 1 (Not Acknowledge) is written to this bit with the ACKWP bit set to 1

[Clearing conditions] One of the following conditions is satisfied:

- When 0 (Acknowledge) is written to this bit with the ACKWP bit set to 1
- When stop condition issuance is detected (when a stop condition is detected with the SP bit in ICCR2 set to 1)
- When 1 is written to the IICRST bit in ICCR1 while the ICE bit in ICCR1 is 0 (RIIC reset)

ACKWP Bit (ACKBT Write Protect)

This bit is used to control the modification of the ACKBT bit.

RDRFS Bit (RDRF Flag Set Timing Selection)

This bit is used to select the RDRF flag set timing in reception mode and also to select whether to hold the SCL line low at the falling edge of the eighth SCL clock cycle.
When the RDRFS bit is 0, the SCL line is not held low at the falling edge of the eighth SCL clock cycle, and the RDRF flag is set to 1 at the rising edge of the ninth SCL clock cycle.
When the RDRFS bit is 1, the RDRF flag is set to 1 at the rising edge of the eighth SCL clock cycle and the SCL line is held low at the falling edge of the eighth SCL clock cycle. Low-hold state is released at the ninth clock cycle or later, by writing the ACKBT bit.
After data is received with this setting, the SCL line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKBT = 0) or NACK (ACKBT = 1) according to receive data.

WAIT Bit (WAIT)

This bit is used to control whether to hold the period between the ninth SCL clock cycle and the first SCL clock cycle low until the receive data buffer (ICDRR) is completely read each time single-byte data is received in reception mode.
When the WAIT bit is 0, the receive operation is continued without holding the period between the ninth and the first SCL clock cycle low. When both the RDRFS and WAIT bits are 0, continuous receive operation is enabled with the double buffer.
When the WAIT bit is 1, the SCL line is held low from the falling edge of the ninth clock cycle until the ICDRR value is read each time single-byte data is received. This enables receive operation in byte units.

Note: When writing 0 to the WAIT bit, be sure to read the ICDRR beforehand.

26.2.6 I²C Bus Function Enable Register (ICFER)

The ICFER register specifies the settings for various arbitration functions.

Address(es): RIIC1.ICFER A008 0945h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
Value after reset:	0	1	1	1	0	0	1	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOE	Timeout Function Enable	0: The timeout function is disabled. 1: The timeout function is enabled.	R/W
b1	MALE	Master Arbitration-Lost Detection Enable	0: Master arbitration-lost detection is disabled. (Disables the arbitration-lost detection function and does not clear the MST and TRS bits in ICCR2 automatically when arbitration is lost.) 1: Master arbitration-lost detection is enabled. (Enables the arbitration-lost detection function and clears the MST and TRS bits in ICCR2 automatically when arbitration is lost.)	R/W
b2	NALE	NACK Transmission Arbitration-Lost Detection Enable	0: NACK transmission arbitration-lost detection is disabled. 1: NACK transmission arbitration-lost detection is enabled.	R/W
b3	SALE	Slave Arbitration-Lost Detection Enable	0: Slave arbitration-lost detection is disabled. 1: Slave arbitration-lost detection is enabled.	R/W
b4	NACKE	NACK Reception Transfer Suspension Enable	0: Transfer operation is not suspended during NACK reception (transfer suspension disabled). 1: Transfer operation is suspended during NACK reception (transfer suspension enabled).	R/W
b5	NFE	Digital Noise Filter Circuit Enable	0: No digital noise filter circuit is used. 1: A digital noise filter circuit is used.	R/W
b6	SCLE	SCL Synchronous Circuit Enable	0: No SCL synchronous circuit is used. 1: An SCL synchronous circuit is used.	R/W
b7	—	Reserved	This bit is read as 0. The write value should be 0.	R/W

TMOE Bit (Timeout Function Enable)

This bit is used to enable or disable the timeout function.
For details on the timeout function, refer to section 26.11.1, Timeout Function.

MALE Bit (Master Arbitration-Lost Detection Enable)

This bit is used to specify whether to use the arbitration-lost detection function in master mode. Normally, set this bit to 1.
For details on master arbitration lost detection function, see section 26.9.1, Master Arbitration-Lost Detection (MALE Bit).

NALE Bit (NACK Transmission Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in reception mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).
For details on NACK transmission arbitration-lost detection function, section 26.9.2, Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit).

SALE Bit (Slave Arbitration-Lost Detection Enable)

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmission mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).
For details on slave arbitration-lost detection function, see section 26.9.3, Slave Arbitration-Lost Detection (SALE Bit).

NACKE Bit (NACK Reception Transfer Suspension Enable)

This bit is used to specify whether to continue or discontinue the transfer operation when NACK is received from the slave device in transmission mode. Normally, set this bit to 1.
When NACK is received with the NACKE bit set to 1, the next transfer operation is suspended.
When the NACKE bit is 0, the next transfer operation is continued regardless of the received acknowledge content.
For details on the NACK reception transfer suspension function, refer to section 26.8.2, NACK Reception Transfer Suspension Function.

SCLE Bit (SCL Synchronous Circuit Enable)

This bit is used to specify whether to synchronize the SCL clock with the SCL input clock. Normally, set this bit to 1.
When the SCLE bit is set to 0 (no SCL synchronous circuit used), the RIIC does not synchronize the SCL clock with the SCL input clock. In this setting, the RIIC outputs the SCL clock with the transfer rate set in ICBRH and ICBRL regardless of the SCL line state. For this reason, if the bus load of the I^2C bus line is much larger than the specification value or if the SCL clock output overlaps in multiple masters, the short-cycle SCL clock that does not meet the specification may be output. When no SCL synchronous circuit is used, it also affects the issuance of a start condition, restart condition, and stop condition, and the continuous output of extra SCL clock cycles.
This bit must not be set to 0 except for checking the output of the set transfer rate.
For details on SCL synchronous circuit function, see section 26.4, SCL Synchronization Circuit.

26.2.7 I²C Bus Status Enable Register (ICSER)

The ICSER register specifies the settings for enabling slave addresses and ID address detection.

Address(es): RIIC1.ICSER A008 0946h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DIDE	—	GCAE	SAR2E	SAR1E	SAR0E
Value after reset:	0	0	0	0	1	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	SAR0E	Slave Address Register 0 Enable	0: Slave address in ICSARL0 and ICSARU0 is disabled. 1: Slave address in ICSARL0 and ICSARU0 is enabled.	R/W
b1	SAR1E	Slave Address Register 1 Enable	0: Slave address in ICSARL1 and ICSARU1 is disabled. 1: Slave address in ICSARL1 and ICSARU1 is enabled.	R/W
b2	SAR2E	Slave Address Register 2 Enable	0: Slave address in ICSARL2 and ICSARU2 is disabled. 1: Slave address in ICSARL2 and ICSARU2 is enabled.	R/W
b3	GCAE	General Call Address Enable	0: General call address detection is disabled. 1: General call address detection is enabled.	R/W
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	DIDE	Device-ID Address Detection Enable	0: Device-ID address detection is disabled. 1: Device-ID address detection is enabled.	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SARyE Bit (Slave Address Register y Enable) (y = 0 to 2)

This bit is used to enable or disable the slave address set in ICSARLy and ICSARUy.

When this bit is set to 1, the slave address set in ICSARLy and ICSARUy is enabled and is compared with the received slave address.

When this bit is set to 0, the slave address set in ICSARLy and ICSARUy is disabled and is ignored even if it matches the received slave address.

GCAE Bit (General Call Address Enable)

This bit is used to specify whether to ignore the general call address (0000 000b + 0 [W]: All 0) when it is received.

When this bit is set to 1, if the received slave address matches the general call address, the RIIC recognizes the received slave address as the general call address independently of the slave addresses set in ICSARLy and ICSARUy (y = 0 to 2) and performs data receive operation.

When this bit is set to 0, the received slave address is ignored even if it matches the general call address.

DIDE Bit (Device-ID Address Detection Enable)

This bit is used to specify whether to recognize and execute the Device-ID address when a device ID (1111 100b) is received in the first frame after a start condition or restart condition is detected.

When this bit is set to 1, if the received first frame matches the device ID, the RIIC recognizes that the Device-ID address has been received. When the following R/W# bit is 0 [W], the RIIC recognizes the second and the following frames as slave addresses and continues the receive operation.

When this bit is set to 0, the RIIC ignores the received first frame even if it matches the device ID address and recognizes the first frame as a normal slave address.

For details on the device-ID address detection, refer to section 26.7.3, Device-ID Address Detection.

26.2.8 I²C Bus Interrupt Enable Register (ICIER)

The ICIER register enables or disables interrupt requests regarding RIIC.

Address(es): RIIC1.ICIER A008 0947h

	b7	b6	b5	b4	b3	b2	b1	b0
	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOIE	Timeout Interrupt Request Enable	0: Timeout interrupt request (TMOI) is disabled. 1: Timeout interrupt request (TMOI) is enabled.	R/W
b1	ALIE	Arbitration-Lost Interrupt Request Enable	0: Arbitration-lost interrupt request (ALI) is disabled. 1: Arbitration-lost interrupt request (ALI) is enabled.	R/W
b2	STIE	Start Condition Detection Interrupt Request Enable	0: Start condition detection interrupt request (STI) is disabled. 1: Start condition detection interrupt request (STI) is enabled.	R/W
b3	SPIE	Stop Condition Detection Interrupt Request Enable	0: Stop condition detection interrupt request (SPI) is disabled. 1: Stop condition detection interrupt request (SPI) is enabled.	R/W
b4	NAKIE	NACK Reception Interrupt Request Enable	0: NACK reception interrupt request (NAKI) is disabled. 1: NACK reception interrupt request (NAKI) is enabled.	R/W
b5	RIE	Receive Data Full Interrupt Request Enable	0: Receive data full interrupt request (RXI) is disabled. 1: Receive data full interrupt request (RXI) is enabled.	R/W
b6	TEIE	Transmit End Interrupt Request Enable	0: Transmit end interrupt request (TEI) is disabled. 1: Transmit end interrupt request (TEI) is enabled.	R/W
b7	TIE	Transmit Data Empty Interrupt Request Enable	0: Transmit data empty interrupt request (TXI) is disabled. 1: Transmit data empty interrupt request (TXI) is enabled.	R/W

TMOIE Bit (Timeout Interrupt Request Enable)

This bit is used to enable or disable timeout interrupt requests (TMOI) when the TMOF flag in ICSR2 is set to 1. A TMOI interrupt request is canceled by setting the TMOF flag or the TMOIE bit to 0.

ALIE Bit (Arbitration-Lost Interrupt Request Enable)

This bit is used to enable or disable arbitration-lost interrupt requests (ALI) when the AL flag in ICSR2 is set to 1. An ALI interrupt request is canceled by setting the AL flag or the ALIE bit to 0.

STIE Bit (Start Condition Detection Interrupt Request Enable)

This bit is used to enable or disable start condition detection interrupt requests (STI) when the START flag in ICSR2 is set to 1. An STI interrupt request is canceled by setting the START flag or the STIE bit to 0.

SPIE Bit (Stop Condition Detection Interrupt Request Enable)

This bit is used to enable or disable stop condition detection interrupt requests (SPI) when the STOP flag in ICSR2 is set to 1. An SPI interrupt request is canceled by setting the STOP flag or the SPIE bit to 0.

NAKIE Bit (NACK Reception Interrupt Request Enable)

This bit is used to enable or disable NACK reception interrupt requests (NAKI) when the NACKF flag in ICSR2 is set to 1. An NAKI interrupt request is canceled by setting the NACKF flag or the NAKIE bit to 0.

RIE Bit (Receive Data Full Interrupt Request Enable)

This bit is used to enable or disable receive data full interrupt requests (RXI) when the RDRF flag in ICSR2 is set to 1.

TEIE Bit (Transmit End Interrupt Request Enable)

This bit is used to enable or disable transmit end interrupt requests (TEI) when the TEND flag in ICSR2 is set to 1. An TEI interrupt request is canceled by setting the TEND flag or the TEIE bit to 0.

TIE Bit (Transmit Data Empty Interrupt Request Enable)

This bit is used to enable or disable transmit data empty interrupt requests (TXI) when the TDRE flag in ICSR2 is set to 1.

26.2.9 I²C Bus Status Register 1 (ICSR1)

The ICSR1 register is a status register that indicates the status of detection of various addresses.

Address(es): RIIC1.ICSR1 A008 0948h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DID	—	GCA	AAS2	AAS1	AAS0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	AAS0	Slave Address 0 Detection Flag	0: Slave address 0 is not detected. 1: Slave address 0 is detected.	R/(W) *1
b1	AAS1	Slave Address 1 Detection Flag	0: Slave address 1 is not detected. 1: Slave address 1 is detected.	R/(W) *1
b2	AAS2	Slave Address 2 Detection Flag	0: Slave address 2 is not detected. 1: Slave address 2 is detected.	R/(W) *1
b3	GCA	General Call Address Detection Flag	0: General call address is not detected. 1: General call address is detected.	R/(W) *1
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b5	DID	Device-ID Address Detection Flag	0: Device-ID address is not detected. 1: Device-ID address is detected. • This bit is set to 1 when the first frame received immediately after a start condition is detected matches a value of (device ID address (1111 100b) + 0[W]).	R/(W) *1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Only 0 can be written to clear the flag.

AASy Flag (Slave Address y Detection Flag) (y = 0 to 2)

[Setting conditions]

For 7-bit address format: ICSARUy.FS = 0

- When the received slave address matches the SVA[6:0] value in ICSARLy with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

For 10-bit address format: ICSARUy.FS = 1

- When the received slave address matches a value of (11110b + SVA[1:0] in ICSARUy) and the following address matches the ICSARLy value with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the AASy bit after reading AASy = 1
- When a stop condition is detected
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

For 7-bit address format: ICSARUy.FS = 0

- When the received slave address does not match the SVA[6:0] value in ICSARLy with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.

For 10-bit address format: ICSARUy.FS = 1

- When the received slave address does not match a value of (11110b + SVA[1:0] in ICSARUy) with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the received slave address matches a value of (11110b + SVA[1:0] in ICSARUy) and the following address does not match the ICSARLy value with the SARyE bit in ICSER set to 1 (slave address y detection enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.

GCA Flag (General Call Address Detection Flag)

[Setting condition]

- When the received slave address matches the general call address (0000 000b + 0 [W]) with the GCAE bit in ICSER set to 1 (general call address detection is enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the GCA bit after reading GCA = 1
- When a stop condition is detected
- When the received slave address does not match the general call address (0000 000b + 0 [W]) with the GCAE bit in ICSER set to 1 (general call address detection is enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

DID Flag (Device-ID Address Detection Flag)

[Setting condition]

- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID address (1111 100b) + 0 [W]) with the DIDE bit in ICSER set to 1 (Device-ID address detection is enabled)
  This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the DID bit after reading DID = 1
- When a stop condition is detected
- When the first frame received immediately after a start condition or restart condition is detected does not match a value of (device ID address (1111 100b)) with the DIDE bit in ICSER set to 1 (Device-ID address detection is enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID address (1111 100b) + 0 [W]) and the second frame does not match any of slave addresses 0 to 2 with the DIDE bit in ICSER set to 1 (Device-ID address detection is enabled)
  This flag is set to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

26.2.10 I²C Bus Status Register 2 (ICSR2)

The ICSR2 register is a status register that indicates the status of detection of various conditions.

Address(es): RIIC1.ICSR2 A008 0949h

	b7	b6	b5	b4	b3	b2	b1	b0
	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMOF	Timeout Detection Flag	0: Timeout is not detected. 1: Timeout is detected.	R/(W) *1
b1	AL	Arbitration-Lost Flag	0: Arbitration is not lost. 1: Arbitration is lost.	R/(W) *1
b2	START	Start Condition Detection Flag	0: Start condition is not detected. 1: Start condition is detected.	R/(W) *1
b3	STOP	Stop Condition Detection Flag	0: Stop condition is not detected. 1: Stop condition is detected.	R/(W) *1
b4	NACKF	NACK Detection Flag	0: NACK is not detected. 1: NACK is detected.	R/(W) *1
b5	RDRF	Receive Data Full Flag	0: ICDRR contains no receive data. 1: ICDRR contains receive data.	R/(W) *1
b6	TEND	Transmit End Flag	0: Data is being transmitted. 1: Data has been transmitted.	R/(W) *1
b7	TDRE	Transmit Data Empty Flag	0: ICDRT contains transmit data. 1: ICDRT contains no transmit data.	R

Note 1. Only 0 can be written to clear the flag.

TMOF Flag (Timeout Detection Flag)

This flag is set to 1 when the RIIC recognizes timeout after the SCL line state remains unchanged for a certain period.

[Setting condition]

- When the SCL line state remains unchanged for the period specified by the ICMR2.TMOH, TMOL, and TMOS bits while the ICFER.TMOE bit is 1 (the timeout function is enabled) in master mode or in slave mode and the received slave address matches.

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the TMOF bit after reading TMOF = 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

AL Flag (Arbitration-Lost Flag)

This flag shows that bus mastership has been lost (loss in arbitration) due to a bus conflict or some other reason when a start condition is issued or an address and data are transmitted. The RIIC monitors the level on the SDA line during transmission and, if the level on the line does not match the value of the bit being output, sets the value of the AL flag to 1 to indicate that the bus is occupied by another device.

The RIIC can also set the flag to indicate the detection of loss of arbitration during NACK transmission in master mode or during data transmission in slave mode.

[Setting conditions] One of the following conditions is satisfied:

When master arbitration-lost detection is enabled: ICFER.MALE = 1

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data (including slave address) transmission in master transmission mode (when the SDA line is driven low while the internal SDA output is at a high level (the SDA pin is in the high-impedance state))
- When a start condition is detected while the ST bit in ICCR2 is 1 (start condition issuance request) or the internal SDA output state does not match the SDA line level
- When the ST bit in ICCR2 is set to 1 (start condition issuance request) with the BBSY flag in ICCR2 set to 1.

When NACK transmission arbitration-lost detection is enabled: ICFER.NALE = 1

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock in the ACK period during NACK transmission in reception mode

When slave arbitration-lost detection is enabled: ICFER.SALE = 1

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data transmission in slave transmission mode

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the AL bit after reading AL = 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Table 26.4 Relationship between Arbitration-Lost Generation Sources and Arbitration-Lost Enable Functions

ICFER			ICSR2		
MALE	NALE	SALE	AL	Error	Arbitration-Lost Generation Source
1	×	×	1	Start condition issuance error	When internal SDA output state does not match SDA line level when a start condition is detected while the ST bit in ICCR2 is 1
					When ST in ICCR2 is set to 1 with BBSY in ICCR2 set to 1
			1	Transmit data mismatch	When transmit data (including slave address) does not match the bus state in master transmission mode
×	1	×	1	NACK transmission mismatch	When ACK is detected during transmission of NACK in master reception mode or slave reception mode
×	×	1	1	Transmit data mismatch	When transmit data does not match the bus state in slave transmission mode

×: Don't care

START Flag (Start Condition Detection Flag)

[Setting condition]

- When a start condition (or a restart condition) is detected

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the START bit after reading START = 1
- When a stop condition is detected
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

STOP Flag (Stop Condition Detection Flag)

[Setting condition]

- When a stop condition is detected

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the STOP bit after reading STOP = 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

NACKF Flag (NACK Detection Flag)

[Setting condition]

- When acknowledge is not received (NACK is received) from the receive device in transmission mode with the NACKE bit in ICFER set to 1 (transfer suspension enabled)

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the NACKF bit after reading NACKF = 1
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: When the NACKF flag is set to 1, the RIIC suspends data transmission/reception. Writing to ICDRT in transmission mode or reading from ICDRR in reception mode with the NACKF flag set to 1 does not enable data transmit/receive operation. To restart data transmission/reception, set the NACKF flag to 0.

RDRF Flag (Receive Data Full Flag)

[Setting conditions] One of the following conditions is satisfied:

- When receive data has been transferred from ICDRS to ICDRR
  This flag is set to 1 at the rising edge of the eighth or ninth SCL clock cycle (selected by the RDRFS bit in ICMR3)
- When the received slave address matches after a start condition (or a restart condition) is detected with the TRS bit in ICCR2 set to 0

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the RDRF bit after reading RDRF = 1
- When data is read from ICDRR
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

TEND Flag (Transmit End Flag)

[Setting condition]

- At the rising edge of the ninth SCL clock cycle while the TDRE flag is 1

[Clearing conditions] One of the following conditions is satisfied:

- When 0 is written to the TEND bit after reading TEND = 1
- When data is written to ICDRT
- When a stop condition is detected
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

TDRE Flag (Transmit Data Empty Flag)

[Setting conditions] One of the following conditions is satisfied:

- When data has been transferred from ICDRT to ICDRS and ICDRT becomes empty
- When the TRS bit in ICCR2 is set to 1 (including when 1 is written to the bit)
- When the received slave address matches while the TRS bit is 1

[Clearing conditions] One of the following conditions is satisfied:

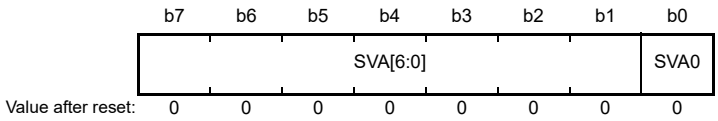
- When data is written to ICDRT
- When the TRS bit in ICCR2 is set to 0 (including when 0 is written to the bit)
- When 1 is written to the IICRST bit in ICCR1 to apply an RIIC reset or an internal reset

Note: When the NACKF flag is set to 1 while the NACKE bit in ICFER is 1, the RIIC suspends data transmission/reception. Here, if the TDRE flag is 0 (next transmit data has been written), data is transferred to the ICDRS register and the ICDRT register becomes empty at the rising edge of the ninth clock cycle, but the TDRE flag is not set to 1.

26.2.11 I²C Slave Address Register Ly (ICSARLy) (y = 0 to 2)

The ICSARLy register specifies the settings for the slave address.

Address(es): RIIC1.ICSARL0 A008 094Ah, RIIC1.ICSARL1 A008 094Ch, RIIC1.ICSARL2 A008 094Eh



Bit	Symbol	Bit Name	Description	R/W
b0	SVA0	10-Bit Address LSB	A slave address is set.	R/W
b7 to b1	SVA[6:0]	7-Bit Address/10-Bit Address Lower Bits	A slave address is set.	R/W

SVA0 Bit (10-Bit Address LSB)

When the 10-bit address format is selected (ICSARUy.FS = 1), this bit functions as the LSB of a 10-bit address and forms the lower 8 bits of a 10-bit address in combination with the SVA[6:0] bits.
When the SARyE bit in ICSER is set to 1 (ICSARLy and ICSARUy enabled) and the ICSARUy.FS bit is 1, this bit is valid. While the ICSARUy.FS bit or SARyE bit is 0, the setting of this bit is ignored.

SVA[6:0] Bits (7-Bit Address/10-Bit Address Lower Bits)

When the 7-bit address format is selected (ICSARUy.FS = 0), these bits function as a 7-bit address. When the 10-bit address format is selected (ICSARUy.FS = 1), these bits function as the lower 8 bits of a 10-bit address in combination with the SVA0 bit.
While the SARyE bit in ICSER is 0, the setting of these bits is ignored.

26.2.12 I²C Slave Address Register Uy (ICSARUy) (y = 0 to 2)

The ICSARUy register specifies the format of the slave address.

Address(es): RIIC1.ICSARU0 A008 094Bh, RIIC1.ICSARU1 A008 094Dh, RIIC1.ICSARU2 A008 094Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SVA[1:0]		FS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	FS	7-Bit/10-Bit Address Format Selection	0: The 7-bit address format is selected. 1: The 10-bit address format is selected.	R/W
b2, b1	SVA[1:0]	10-Bit Address Upper Bits	A slave address is set.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

FS Bit (7-Bit/10-Bit Address Format Selection)

This bit is used to select 7-bit address or 10-bit address for slave address y (in ICSARLy and ICSARUy).
When the SARyE bit in ICSER is set to 1 (ICSARLy and ICSARUy enabled) and the ICSARUy.FS bit is 0, the 7-bit address format is selected for slave address y, the SVA[6:0] setting in ICSARLy is valid, and the settings of the SVA[1:0] bits and the SVA0 bit in ICSARLy are ignored.
When the SARyE bit in ICSER is set to 1 (ICSARLy and ICSARUy enabled) and the ICSARUy.FS bit is 1, the 10-bit address format is selected for slave address y and the settings of the SVA[1:0] bits and ICSARLy are valid.
While the SARyE bit in ICSER is 0 (ICSARLy and ICSARUy disabled), the setting of the ICSARUy.FS bit is invalid.

SVA[1:0] Bits (10-Bit Address Upper Bits)

When the 10-bit address format is selected (FS = 1), these bits function as the upper 2 bits of a 10-bit address.
When the SARyE bit in ICSER is set to 1 (ICSARLy and ICSARUy enabled) and the ICSARUy.FS bit is 1, these bits are valid. While the ICSARUy.FS bit or SARyE bit is 0, the setting of these bits is ignored.

26.2.13 I²C Bus Bit Rate Low-Level Register (ICBRL)

The ICBRL register specifies the low-level period of SCL clock and the delay cycle to which the SDA signal is added.

Address(es): RIIC1.ICBRL A008 0950h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	BRL[4:0]				
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	BRL[4:0]	Bit Rate Low-Level Period	Low-level period of SCL clock	R/W
b7 to b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

ICBRL works to generate the data setup time for automatic SCL low-hold operation (refer to section 26.8, Automatic Low-Hold Function for SCL); when the RIIC is used only in slave mode, this register needs to be set to a value longer than the data setup time*1.
ICBRL counts the low-level period with the internal reference clock source (IICϕ) specified by the CKS[2:0] bits in ICMR1.
If the digital noise filter is enabled (the NFE bit in ICFER is 1), set the ICBRL register to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the ICMR3.NF[1:0] bits.

Note 1. Data setup time (tSU: DAT)
100 ns (up to 400 Kbps: fast mode [Fm])

26.2.14 I²C Bus Bit Rate High-Level Register (ICBRH)

The ICBRH register is a 5-bit register to set the high-level period of SCL clock, and ICBRH is valid in master mode. If the RIIC is used only in slave mode, this register need not set the high-level period.
ICBRH counts the high-level period with the internal reference clock source (IICϕ) specified by the CKS[2:0] bits in ICMR1.

Address(es): RIIC1.ICBRH A008 0951h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	BRH[4:0]				
Value after reset:	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	BRH[4:0]	Bit Rate High-Level Period	High-level period of SCL clock	R/W
b7 to b5	—	Reserved	These bits are read as 1. The write value should be 1.	R/W

If the digital noise filter is enabled (the NFE bit in ICFER is 1), set the ICBRH register to a value at least one greater than the number of stages in the noise filter. Regarding the number of stages in the noise filter, see the description of the ICMR3.NF[1:0] bits.

The I²C transfer rate and the SCL clock duty are calculated using the following expression.
Transfer rate = 1 / {[(ICBRH + 1) + (ICBRL + 1)] / IICϕ*1 + SCL line rising time [tr] + SCL line falling time [tf]}
Duty cycle = {SCL line rising time [tr]*2 + (ICBRH + 1) / IICϕ} / {SCL line falling time [tf]*2 + (ICBRL + 1) / IICϕ}

Note 1. IICϕ = Set value in ICMR1.CKS[2:0]
Note 2. The SCL line rising time [tr] and SCL line falling time [tf] depend on the total bus line capacitance [Cb] and the pull-up resistor [Rp]. For details, see the I²C bus standard from NXP Semiconductors.

Table 26.5 lists examples of ICBRH/ICBRL settings.

Table 26.5 Examples of ICBRH/ICBRL Settings for Transfer Rate

	Operating Frequency PCLKD (MHz)		
	75		
Transfer Rate (Kbps)	CKS[2:0]	ICBRH	ICBRL
10	111b	26 (FAh)	30 (FEh)
50	101b	20 (F4h)	23 (F7h)
100	100b	19 (F3h)	23 (F7h)
400	010b	11 (EBh)	24 (F8h)

Note: This is an example of the setting when the rising time (tr) of the SCL line is 300 ns and the falling time (tf) of the SCL line is 300 ns. For the specified values of the rising time (tr) and the falling time (tf) of the SCL line, see the I²C bus specifications from NXP Semiconductors.

26.2.15 I²C Bus Transmit Data Register (ICDRT)

The ICDRT register stores transmit data.

Address(es): RIIC1.ICDRT A008 0952h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	1	1	1	1	1	1	1	1

When ICDRT detects a space in the I2C bus shift register (ICDRS), it transfers the transmit data that has been written to ICDRT to ICDRS and starts transmitting data in transmission mode.
The double-buffer structure of ICDRT and ICDRS allows continuous transmit operation if the next transmit data has been written to ICDRT while the ICDRS data is being transmitted.
ICDRT can always be read and written. Write transmit data to ICDRT once when a transmit data empty interrupt (TXI) request is generated.

26.2.16 I²C Bus Receive Data Register (ICDRR)

The ICDRR register stores receive data.

Address(es): RIIC1.ICDRR A008 0953h

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0

When 1 byte of data has been received, the received data is transferred from the I2C bus shift register (ICDRS) to ICDRR to enable the next data to be received.
The double-buffer structure of ICDRS and ICDRR allows continuous receive operation if the received data has been read from ICDRR while ICDRS is receiving data.
ICDRR cannot be written. Read data from ICDRR once when a receive data full interrupt (RXI) request is generated.
If ICDRR receives the next receive data before the current data is read from ICDRR (while the RDRF flag in ICSR2 is 1), the RIIC automatically holds the SCL clock low one cycle before the RDRF flag is set to 1 next.

26.2.17 I²C Bus Shift Register (ICDRS)

The ICDRS register is a shift register to transmit and receive data.

	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	—	—	—	—	—	—	—	—

During transmission, transmit data is transferred from ICDRT to ICDRS and is sent from the SDA pin. During reception, data is transferred from ICDRS to ICDRR after 1 byte of data has been received.
ICDRS cannot be accessed directly.

26.3 Operation

26.3.1 Communication Data Format

The I2C bus format consists of 8-bit data and 1-bit acknowledge. The frame following a start condition or restart condition is an address frame used to specify a slave device with which the master device communicates. The specified slave is valid until a new slave is specified or a stop condition is issued.
Figure 26.3 shows the I2C bus format, and Figure 26.4 shows the I2C bus timing.

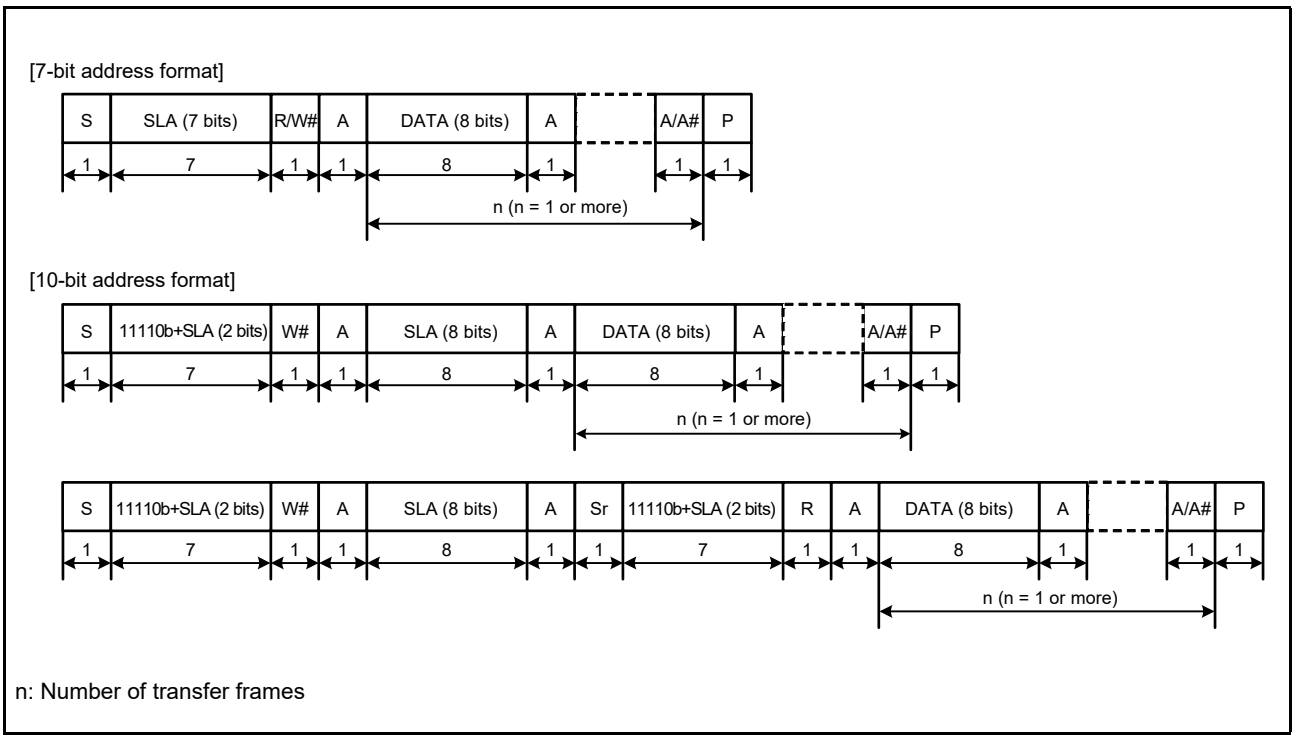


Figure 26.3 I2C Bus Format

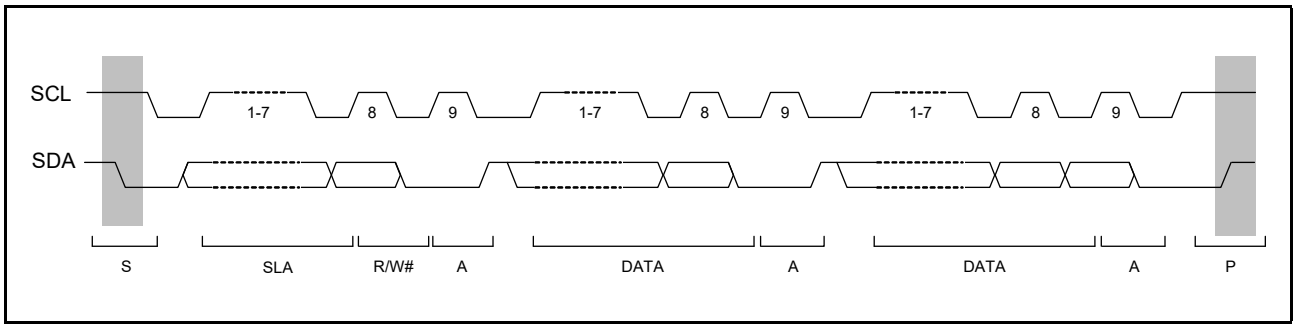


Figure 26.4 I²C Bus Timing (SLA = 7 Bits)

S: Start condition. The SDA line is changed from the high level to the low level while the SCL line of the master device is at a high level.

SLA: Slave address, by which the master device selects a slave device.

R/W#: Indicates the direction of data transfer: from the slave device to the master device when R/W# is 1, or from the master device to the slave device when R/W# is 0.

A: Acknowledge. The receive device drives the SDA line low. (In master transmission mode, the slave device returns acknowledge. In master reception mode, the master device returns acknowledge.)

A#: Not Acknowledge. The receive device drives the SDA line high.

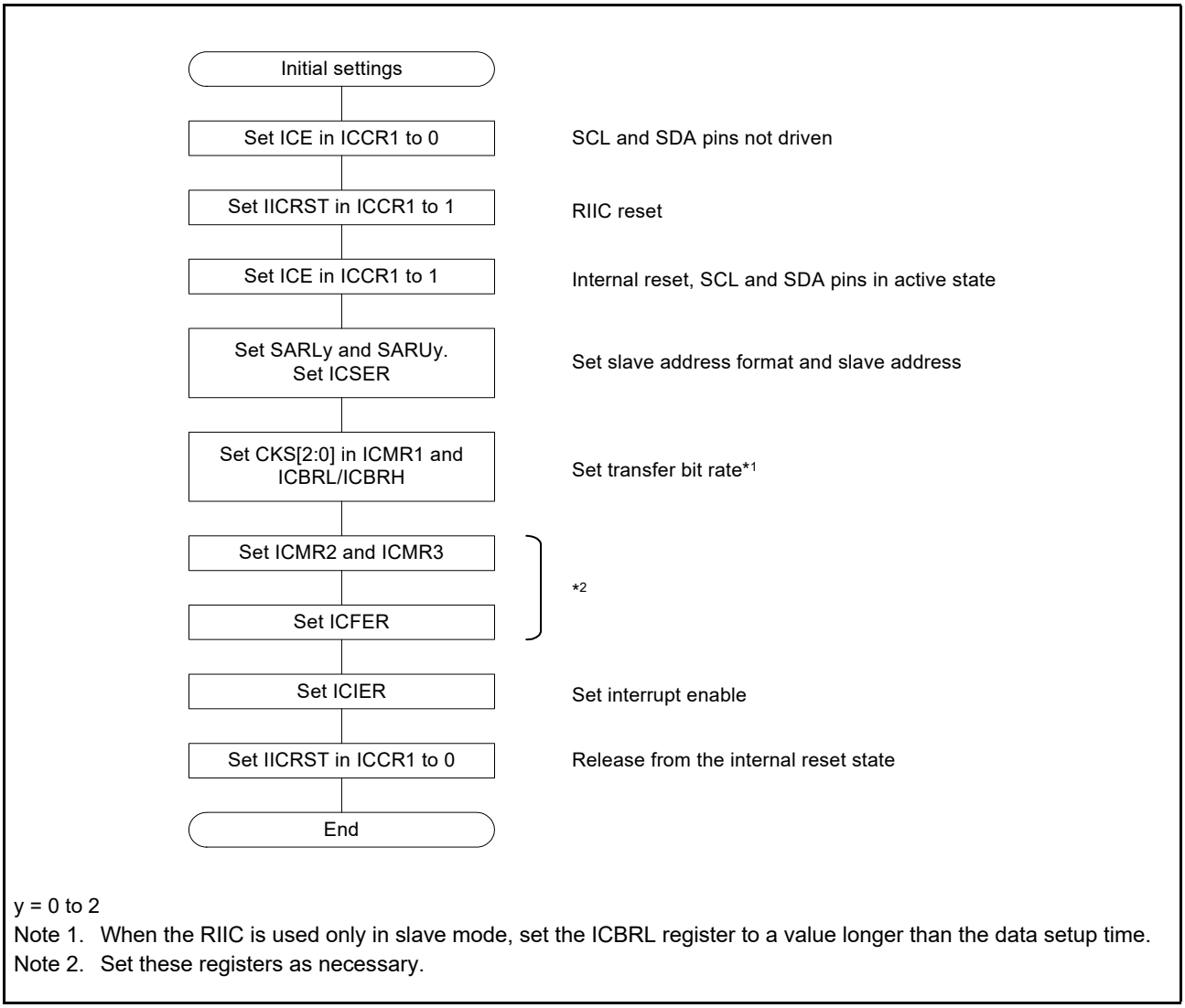
Sr: Restart condition. The SDA line is changed from the high level to the low level after the setup time has elapsed with the SCL line at the high level.

DATA: Transmitted or received data. The bit length of the transmitted or received data is set in ICMR1.BC[2:0].

P: Stop condition. The SDA line is changed from the low level to the high level while the SCL line is at a high level.

26.3.2 Initial Settings

Before starting data transmission and reception, initialize the RIIC according to the procedure in Figure 26.5.
Set the ICCR1.ICE bit to 1 (internal reset) after setting the ICCR1.IICRST bit to 1 (RIIC reset) with the ICCR1.ICE bit set to 0 (SCL and SDA pins in inactive state). This initializes the various flags and internal state of the ICSR1 register. After that, set registers ICSARLy, ICSARUy, ICSER, ICMR1, ICBRH, and ICBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see Figure 26.5). When the necessary register settings have been completed, set the ICCR1.IICRST bit to 0 (releases the RIIC reset). This step is not necessary if initialization of the RIIC has already been completed.



y = 0 to 2
Note 1. When the RIIC is used only in slave mode, set the ICBRL register to a value longer than the data setup time.
Note 2. Set these registers as necessary.

Figure 26.5 Example of RIIC Initialization Flowchart

26.3.3 Master Transmit Operation

In master transmit operation, the RIIC outputs the SCL clock and transmitted data signals as the master device, and the slave device returns acknowledgements. Figure 26.6 shows an example of usage of master transmission and Figure 26.7 to Figure 26.9 show the timing of operations in master transmission.
The following describes the procedure and operations for master transmission.

(1) Initial settings. For details, refer to section 26.3.2, Initial Settings.
(2) Read the BBSY flag in ICCR2 to check that the bus is open, and then set the ST bit in ICCR2 to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. At the same time, the BBSY flag and the START flag in ICSR2 are automatically set to 1 and the ST bit is automatically set to 0. At this time, if the start condition is detected and the internal levels for the SDA output state and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the MST and TRS bits in ICCR2 are automatically set to 1, placing the RIIC in master transmission mode. The TDRE flag in ICSR2 is also automatically set to 1 in response to setting of the TRS bit to 1.
(3) Check that the TDRE flag in ICSR2 is 1, and then write the value for transmission (the slave address and the R/W# bit) to ICDRT. Once the data for transmission are written to ICDRT, the TDRE flag is automatically set to 0, the data are transferred from ICDRT to ICDRS, and the TDRE flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRS bit is automatically updated to select master transmit or master reception mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, the RIIC continues in master transmission mode.
Since the ICSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the ICCR2.SP bit to issue a stop condition.
For data transmission with an address in the 10-bit format, start by writing 1111 0b, the 2 higher-order bits of the slave address, and W to ICDRT as the first address transmission. Then, as the second address transmission, write the 8 lower-order bits of the slave address to ICDRT.
(4) After confirming that the TDRE flag in ICSR2 is 1, write the data for transmission to the ICDRT register. The RIIC automatically holds the SCL line low until the data for transmission are ready or a stop condition is issued.
(5) After all bytes of data for transmission have been written to the ICDRT register, wait until the value of the TEND flag in ICSR2 returns to 1, and then set the SP bit in ICCR2 to 1 (stop condition issuance request). Upon receiving a stop condition issuance request, the RIIC issues the stop condition.
(6) Upon detecting the stop condition, the RIIC automatically sets the MST and TRS bits in ICCR2 to 00b and enters slave reception mode. Furthermore, it automatically sets the TDRE and TEND flags in ICSR2 to 0, and sets the STOP flag in ICSR2 to 1.
(7) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

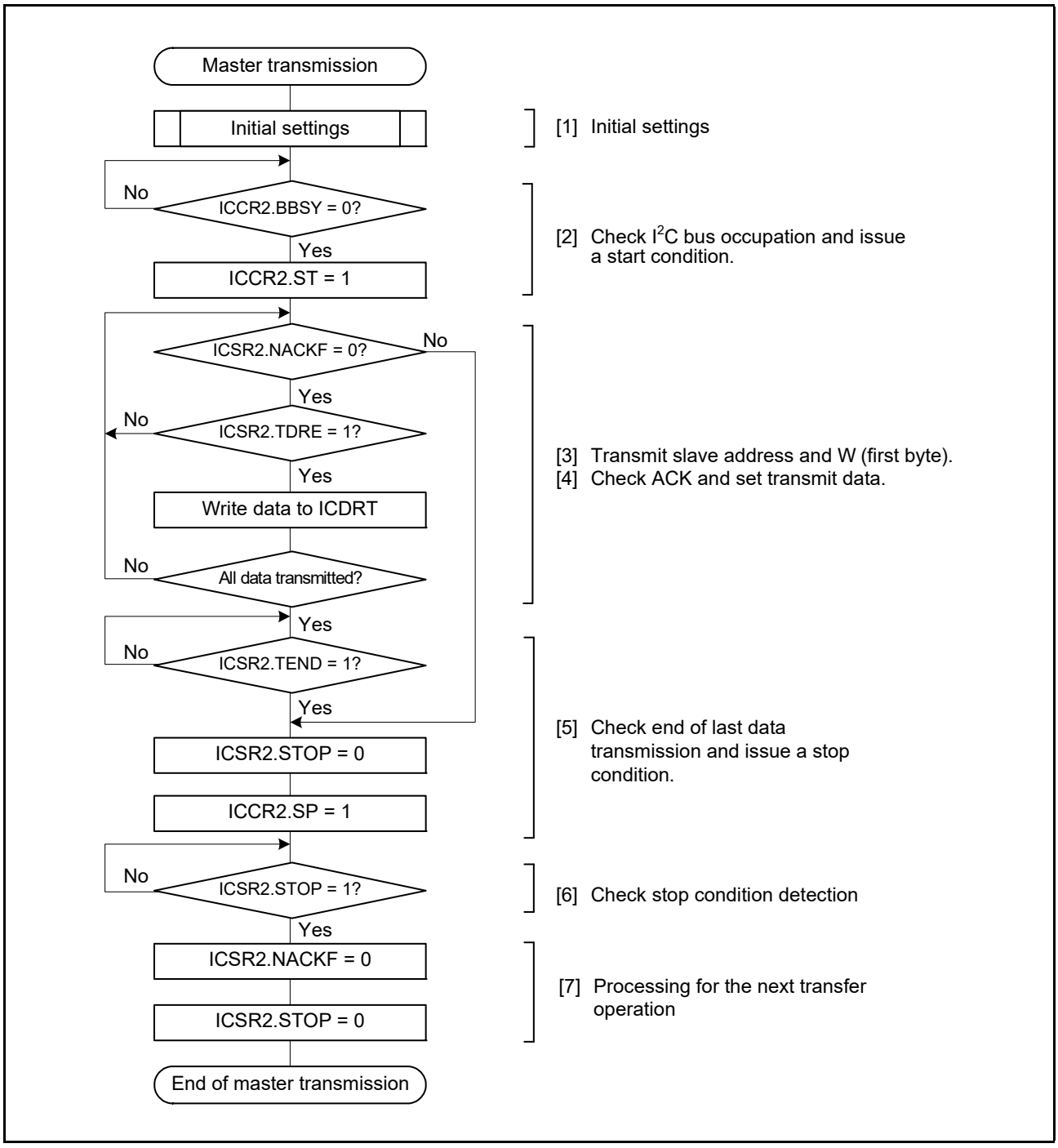


Figure 26.6 Example of Master Transmission Flowchart

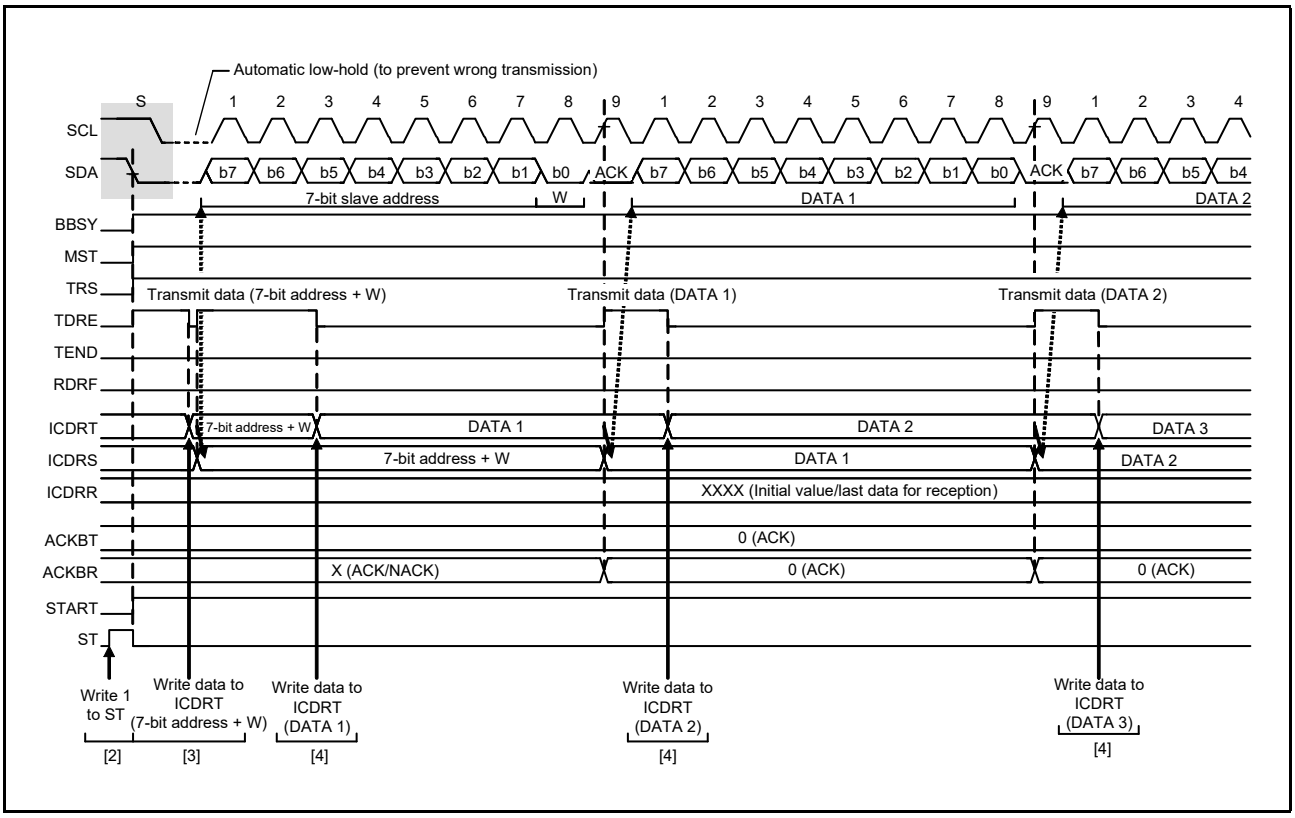


Figure 26.7 Master Transmit Operation Timing (1) (7-Bit Address Format)

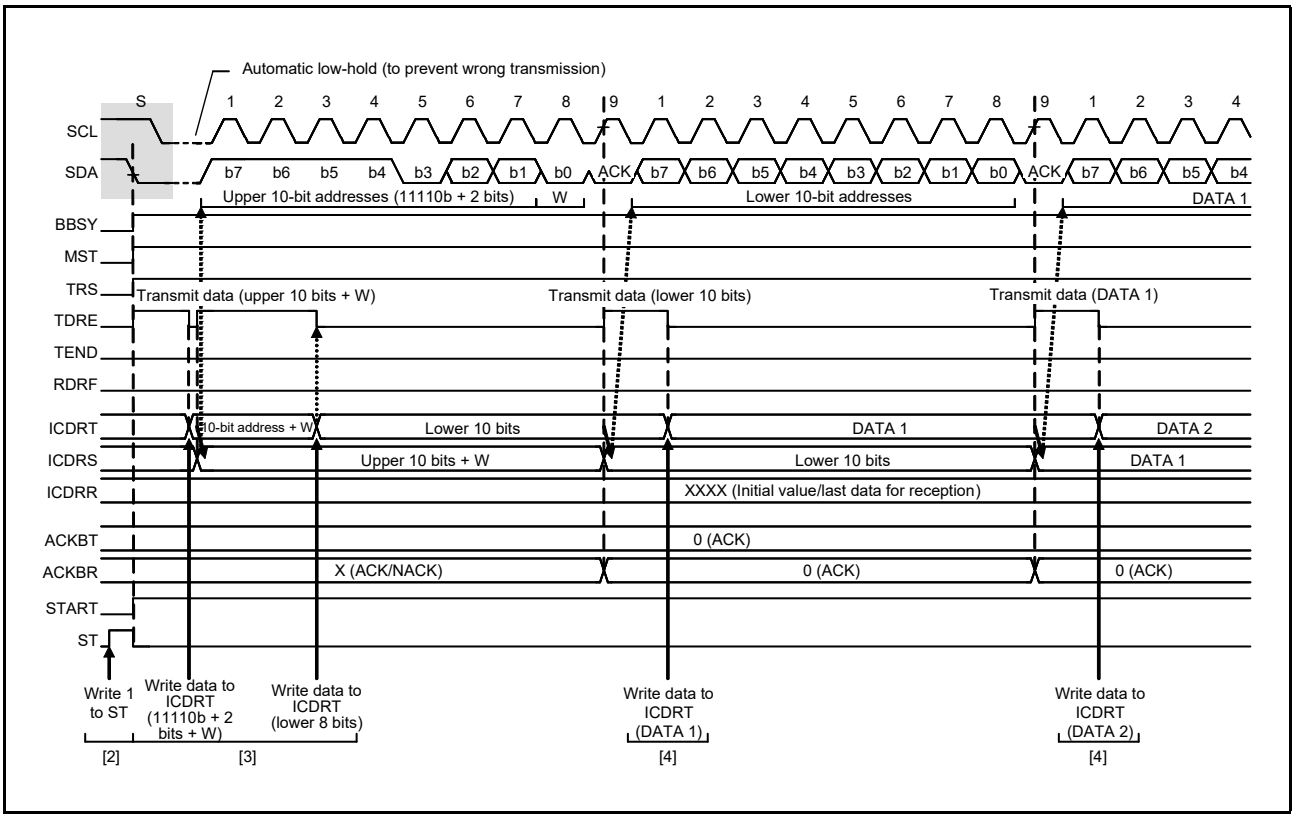


Figure 26.8 Master Transmit Operation Timing (2) (10-Bit Address Format)

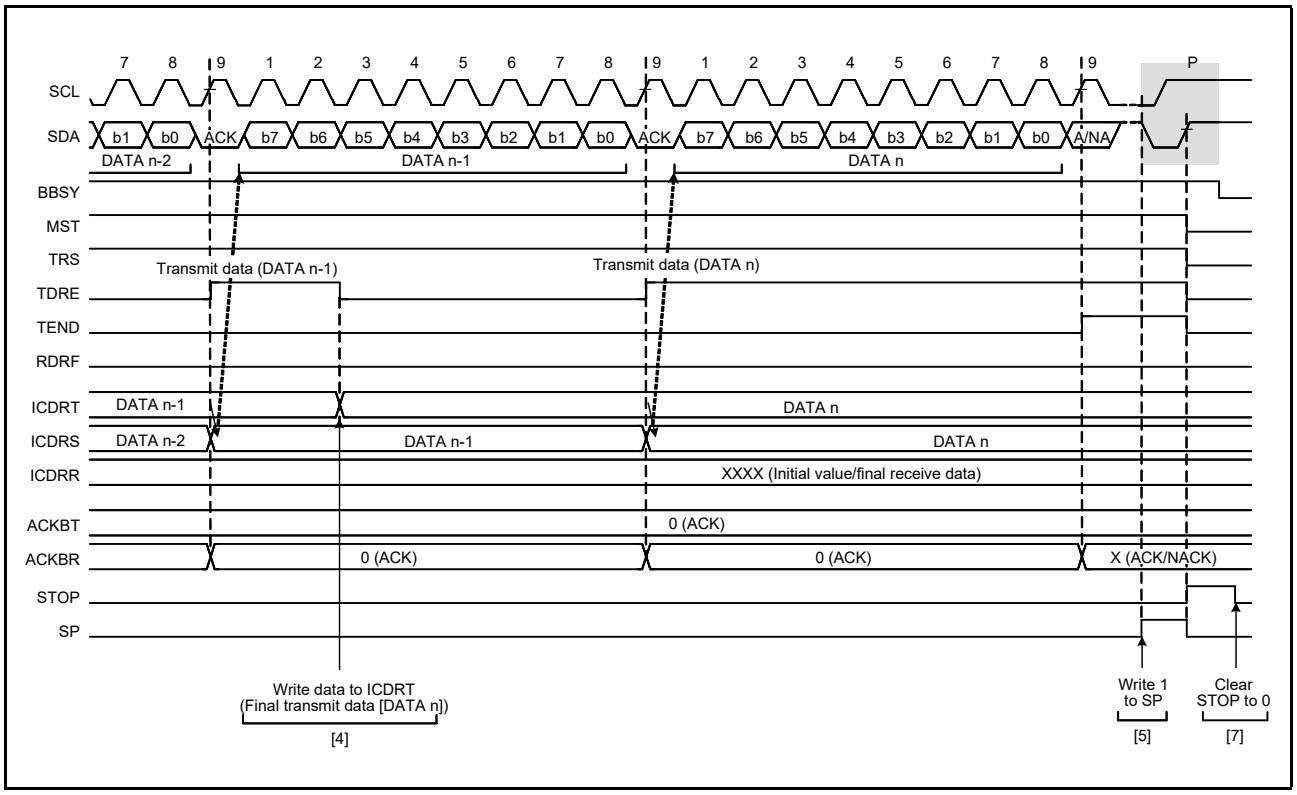


Figure 26.9 Master Transmit Operation Timing (3)

26.3.4 Master Receive Operation

In master receive operation, the RIIC as a master device outputs the SCL clock, receives data from the slave device, and returns acknowledgements. Since the RIIC must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmission mode, but the subsequent steps are in master reception mode.
Figure 26.10 and Figure 26.11 show examples of usage of master reception in the 7-bit address format and Figure 26.12 to Figure 26.14 show the timing of operations in master reception.
The following describes the procedure and operations for master reception.

(1) Initial settings. For details, refer to section 26.3.2, Initial Settings.
(2) Read the BBSY flag in ICCR2 to check that the bus is open, and then set the ST bit in ICCR2 to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. When the RIIC detects the start condition, the BBSY flag and the START flag in ICSR2 are automatically set to 1 and the ST bit is automatically set to 0. At this time, if the start condition is detected and the levels for the SDA output and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the MST and TRS bits in ICCR2 are automatically set to 1, placing the RIIC in master transmission mode. The TDRE flag in ICSR2 is also automatically set to 1 in response to setting of the TRS bit to 1.
(3) Check that the TDRE flag in ICSR2 is 1, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to ICDRT. Once the data for transmission are written to ICDRT, the TDRE flag is automatically set to 0, the data are transferred from ICDRT to ICDRS, and the TDRE flag is again set to 1. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the ICCR2.TRS bit is automatically updated to select transmit or reception mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the TRS bit is set to 0 on the rising edge of the ninth cycle of SCL clock, placing the RIIC in master reception mode. At this time, the TDRE flag is set to 0 and the ICSR2.RDRF flag is automatically set to 1.
Since the ICSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the ICCR2.SP bit to issue a stop condition.
For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a restart condition. After that, transmitting 1111 0b, the two higher-order bits of the slave address, and the R bit places the RIIC in master reception mode.
(4) Dummy read ICDRR after confirming that the RDRF flag in ICSR2 is 1; this makes the RIIC start output of the SCL clock and start data reception.
(5) After 1 byte of data has been received, the RDRF flag in ICSR2 is set to 1 on the rising edge of the eighth or ninth cycle of SCL clock (the clock signal) as selected by the RDRFS bit in ICMR3. Reading out ICDRR at this time will produce the received data, and the RDRF flag is automatically set to 0 at the same time. Furthermore, the value of the acknowledgement field received during the ninth cycle of SCL clock is returned as the value set in the ICMR3.ACKBT bit. If the next byte to be received is the next to last byte, set the ICMR3.WAIT bit to 1 (for wait insertion) before reading the ICDRR (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the ICMR3.ACKBT bit to 1 (NACK) in step (6), due to other interrupts, etc., this fixes the SCL line to the low level on the falling edge of the ninth clock cycle in reception of the last byte, so the state is such that issuing a stop condition is possible.
(6) When the ICMR3.RDRFS bit is 0 and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the ICMR3.ACKBT bit to 1 (NACK).

(7) After reading out the byte before last from the ICDRR register, if the value of the ICSR2.RDRF flag is confirmed to be 1, write 1 to the SP bit in ICCR2 (stop condition issuance request) and then read the last byte from ICDRR. When ICDRR is read, the RIIC is released from the wait state and issues the stop condition after low-level output in the ninth clock cycle is completed or the SCL line is released from the low-hold state.

(8) Upon detecting the stop condition, the RIIC automatically sets the MST and TRS bits in ICCR2 to 00b and enters slave reception mode. Furthermore, detection of the stop condition leads to setting of the ICSR2.STOP flag to 1.

(9) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

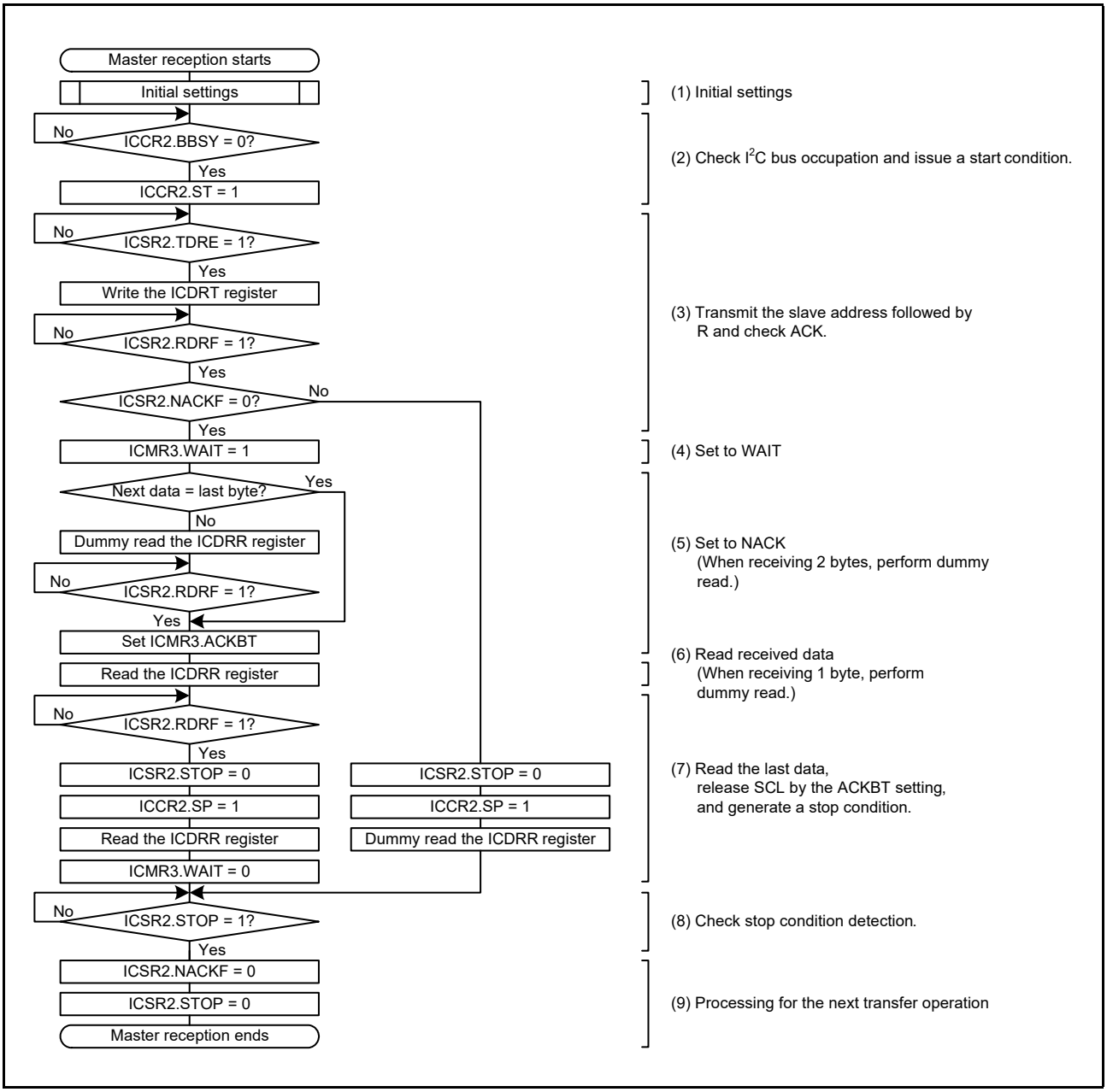


Figure 26.10 Example of Master Reception (7-Bit Address Format, 1 or 2 bytes)

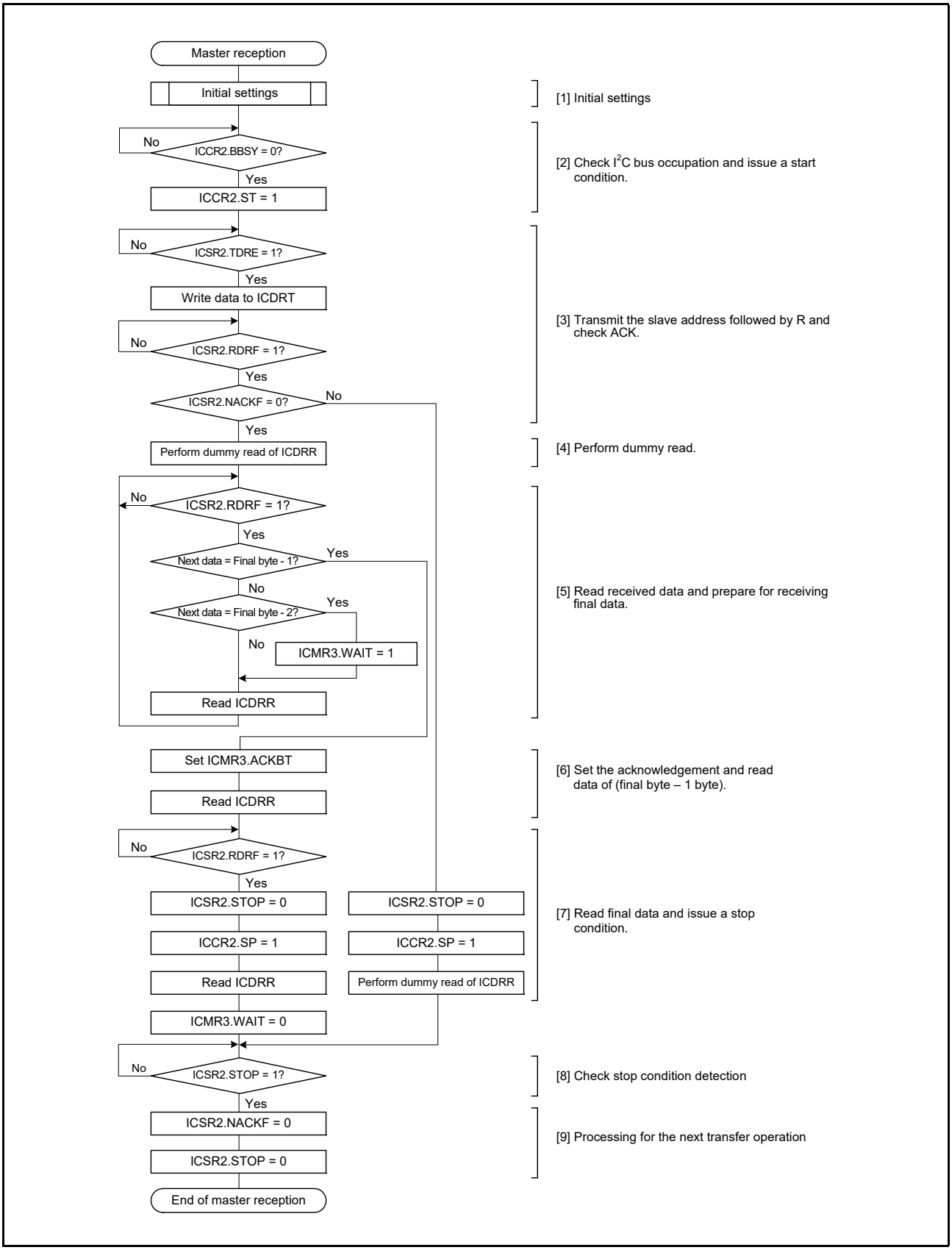


Figure 26.11 Example of Master Reception (7-Bit Address Format, 3 Bytes or More)

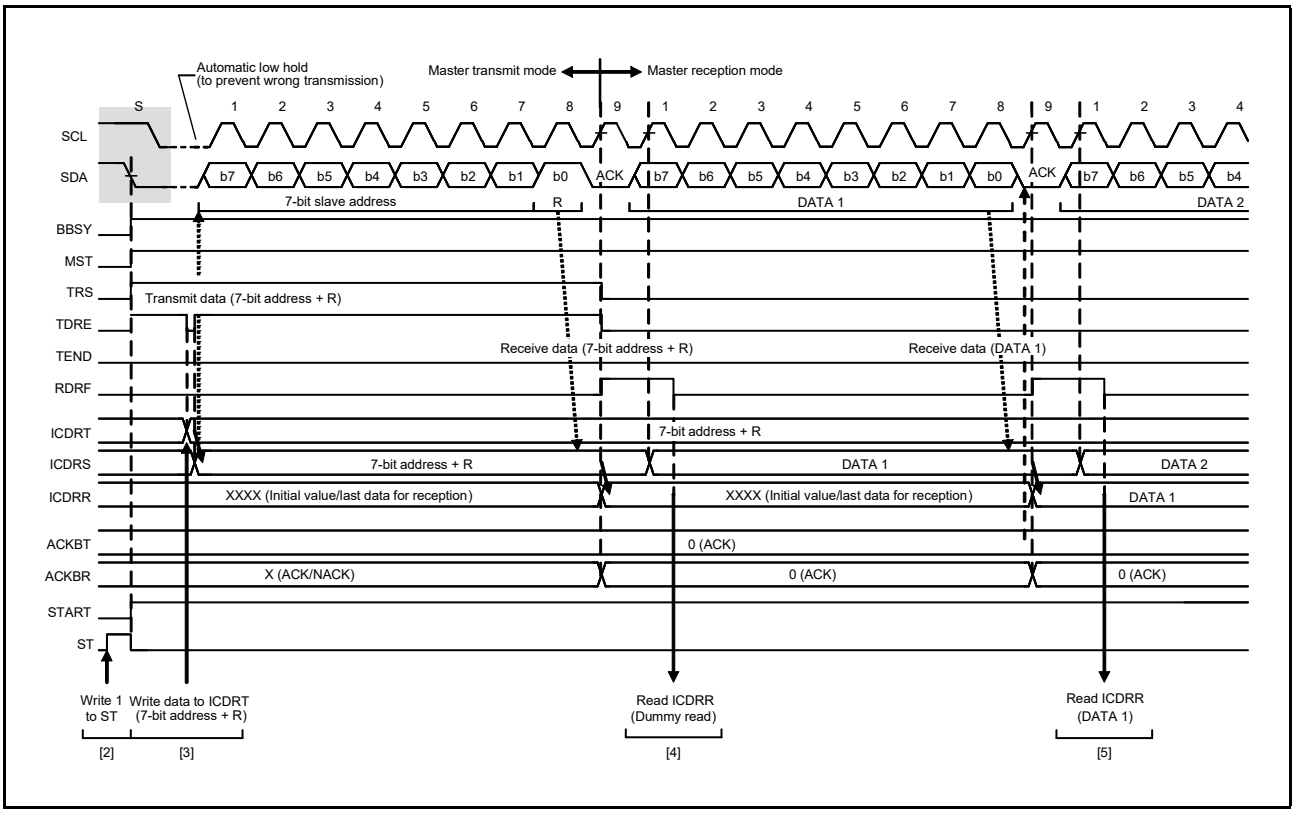


Figure 26.12 Master Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

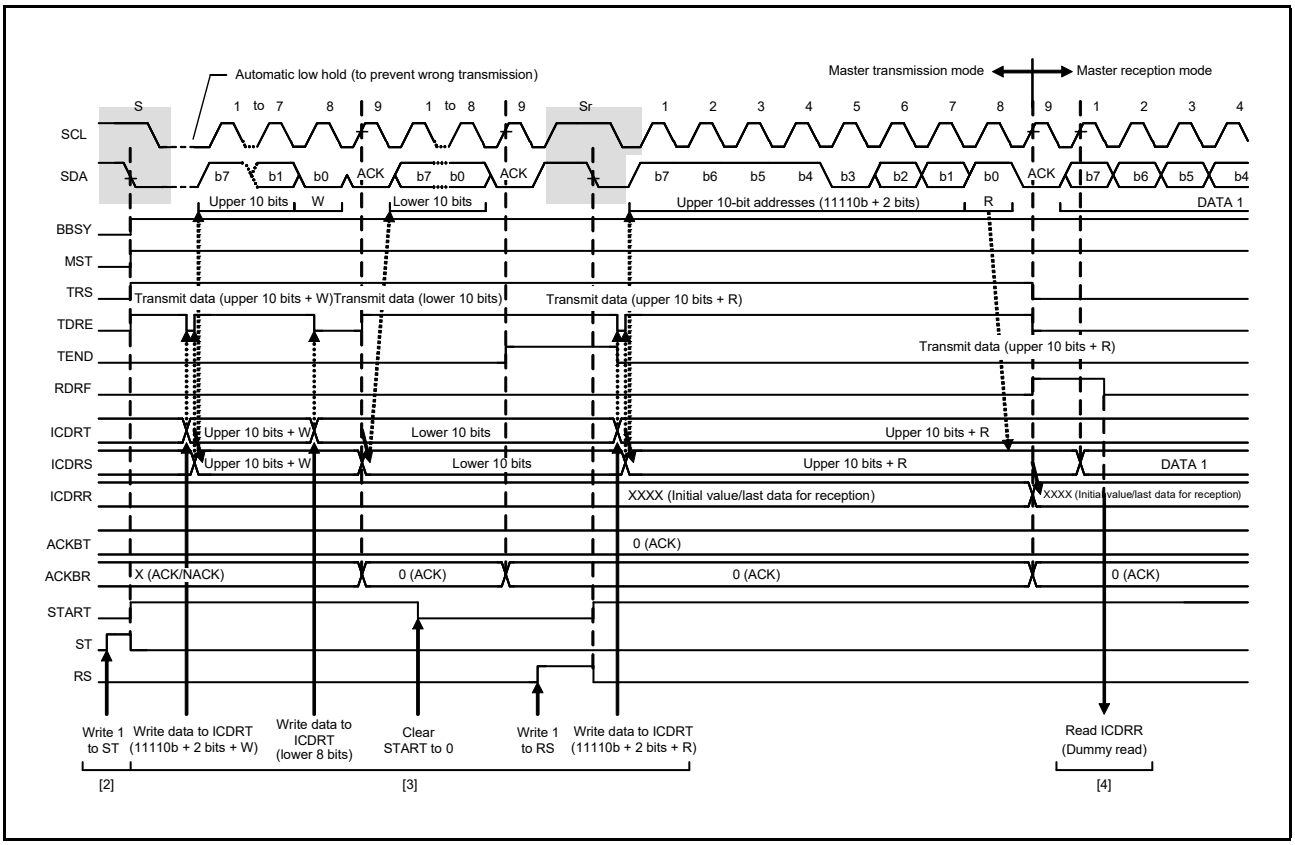


Figure 26.13 Master Receive Operation Timing (2) (10-Bit Address Format, when RDRFS = 0)

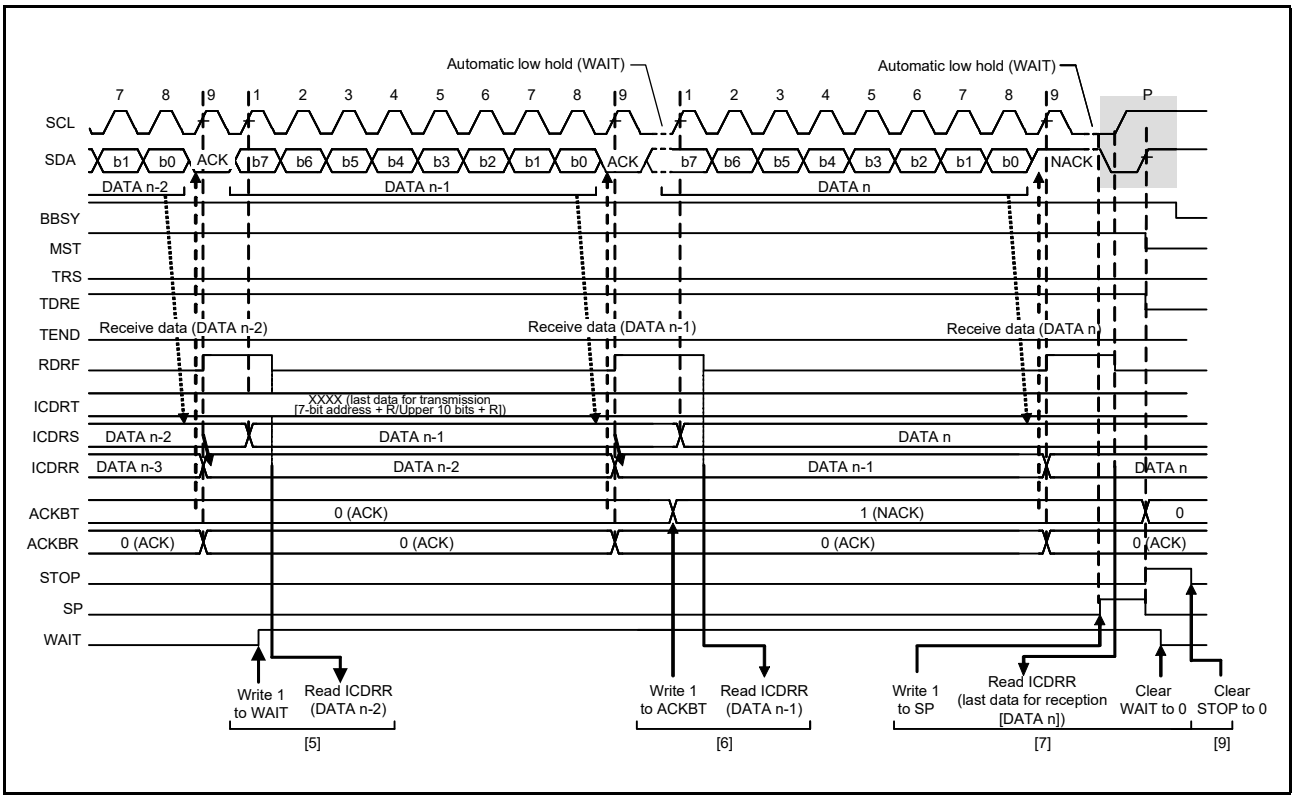


Figure 26.14 Master Receive Operation Timing (3) (when RDRFS = 0)

26.3.5 Slave Transmit Operation

In slave transmit operation, the master device outputs the SCL clock, the RIIC transmits data as a slave device, and the master device returns acknowledgements.
Figure 26.15 shows an example of usage of slave transmission and Figure 26.16 and Figure 26.17 show the timing of operations in slave transmission.
The following describes the procedure and operations for slave transmission.

(1) Initial settings. For details, refer to section 26.3.2, Initial Settings.
After initial settings, the RIIC will stay in the standby state in slave reception mode until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets the corresponding bit among ICSR1.GCA and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of the SCL clock and outputs the value set in the ICMR3.ACKBT bit as the acknowledge bit on the ninth cycle of the SCL clock.
With the 7-bit address format, if the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmission mode by setting both the TRS bit and the TDRE flag in ICSR2 to 1.
With the 10-bit address format, after receiving a matching slave address, check that the ICSR2.STOP flag and the ICSR2.RDRF flag are 0 and 1, respectively, and dummy-read the ICDRR register. Here, the received value that is dummy read is the lower-order 8 bits of the address. Following the dummy read, data reception is restarted on detection of a restart condition. If the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmission mode by setting both the ICCR2.TRS bit and the ICSR2.TDRE flag to 1.

(3) After the ICSR2.TDRE flag is confirmed to be 1, write the data for transmission to the ICDRT register. At this time, if the RIIC receives no acknowledge from the master device (receives an NACK signal) while the ICFER.NACKE bit is 1, the RIIC suspends transfer of the next data.

(4) Wait until the ICSR2.TEND flag is set to 1 while the ICSR2.TDRE flag is 1, after the ICSR2.NACKF flag is set to 1 or the last byte for transmission is written to the ICDRT register. When the ICSR2.NACKF flag or the TEND flag is 1, the RIIC drives the SCL line low on the ninth falling edge of SCL clock.

(5) When the ICSR2.NACKF flag or the ICSR2.TEND flag is 1, dummy read ICDRR to complete the processing. This releases the SCL line.

(6) Upon detecting the stop condition, the RIIC automatically sets bits ICSR1.GCA and AASy (y = 0 to 2), flags ICSR2.TDRE and TEND, and the ICCR2.TRS bit to 0, and enters slave reception mode.

(7) After checking that the ICSR2.STOP flag is 1, set the ICSR2.NACKF and STOP flags to 0 for the next transfer operation.

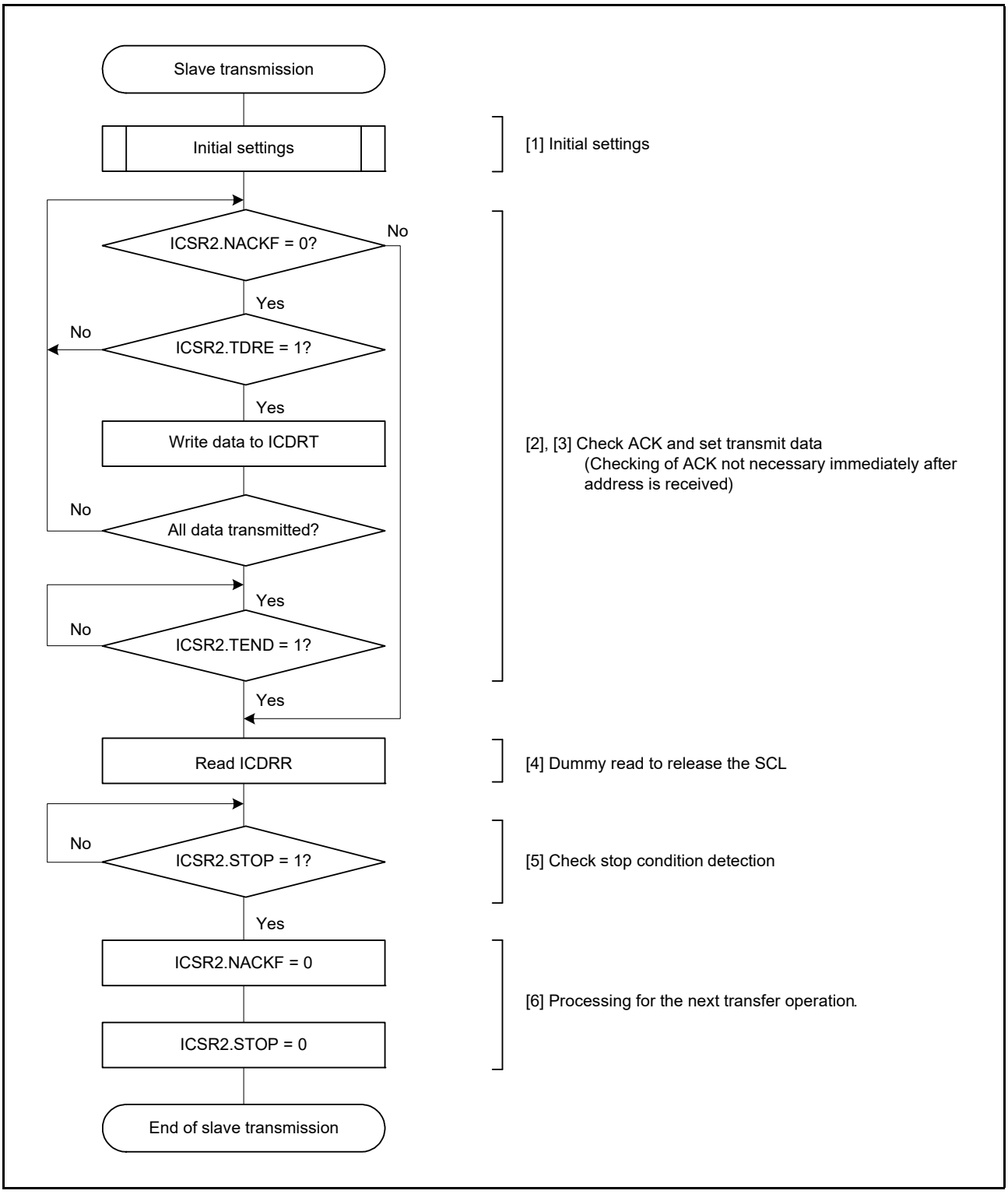


Figure 26.15 Example of Slave Transmission Flowchart

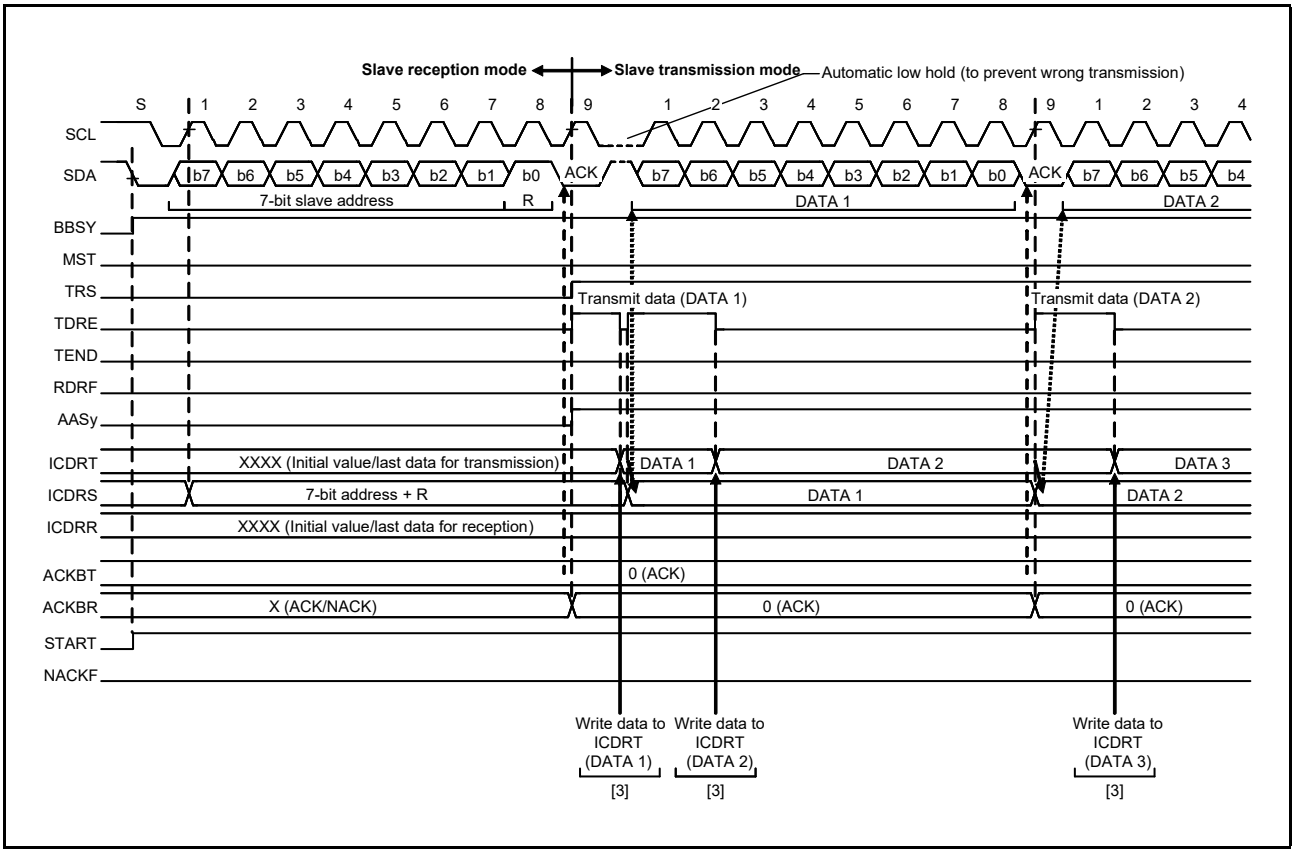


Figure 26.16 Slave Transmit Operation Timing (1) (7-Bit Address Format)

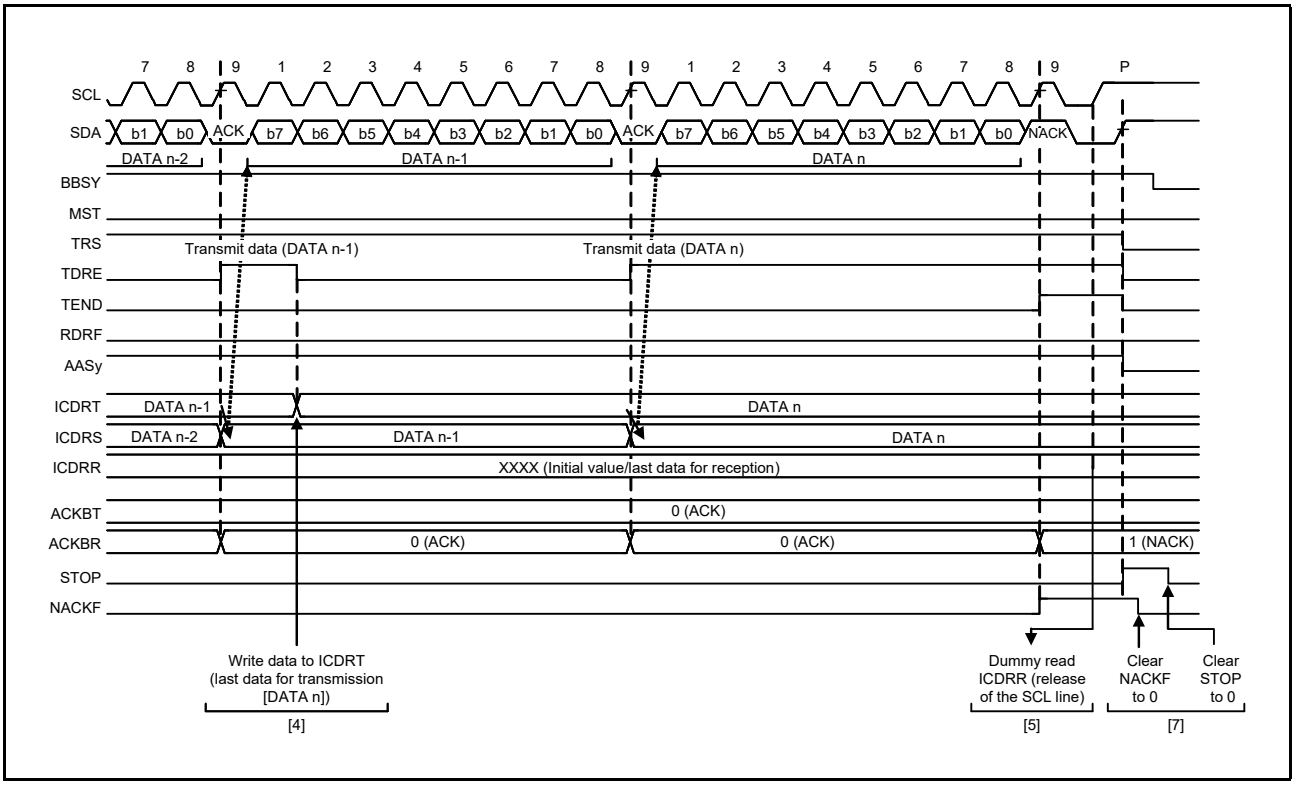


Figure 26.17 Slave Transmit Operation Timing (2)

26.3.6 Slave Receive Operation

In slave receive operation, the master device outputs the SCL clock and transmit data, and the RIIC returns acknowledgements as a slave device.

Figure 26.18 shows an example of usage of slave reception and Figure 26.19 and Figure 26.20 show the timing of operations in slave reception.

The following describes the procedure and operations for slave reception.

(1) Initial settings. For details, refer to section 26.3.2, Initial Settings.
After initial settings, the RIIC will stay in the standby state in slave reception mode until it receives a slave address that it matches.

(2) After receiving a matching slave address, the RIIC sets one of the corresponding bits ICSR1.GCA and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the ICMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 0, the RIIC continues to place itself in slave reception mode and sets the RDRF flag in ICSR2 to 1.

(3) After the ICSR2.STOP flag is confirmed to be 0 and the ICSR2.RDRF flag to be 1, dummy read ICDRR (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower 8 bits when the 10-bit address format is selected). After the dummy-read, the RIIC clears the RDRF flag to 0 and starts reception of data.

(4) On completion of the reception of 1 byte of data, the value of the ICSR2.RDRF flag becomes 1 on the rising edge of the 8th or 9th clock cycle of the SCL clock, according to the setting made in the ICMR3.RDRFS bit. At this time, reading the ICDRR register returns the received value and the RDRF flag is automatically cleared to 0 in response. If reading of ICDRR is delayed and a next byte is received while the RDRF flag is still set to 1, the RIIC holds the SCL line low from the falling edge of one SCL cycle before the timing with which RDRF should be set. In this case, reading ICDRR releases the SCL line from being held at the low level.
When the ICSR2.STOP flag is 1 and the ICSR2.RDRF flag is also 1, read ICDRR until all the data is completely received.

(5) Upon detecting the stop condition, the RIIC automatically clears bits ICSR1.GCA and AASy (y = 0 to 2) to 0.

(6) After checking that the ICSR2.STOP flag is 1, set the ICSR2.STOP flag to 0 for the next transfer operation.

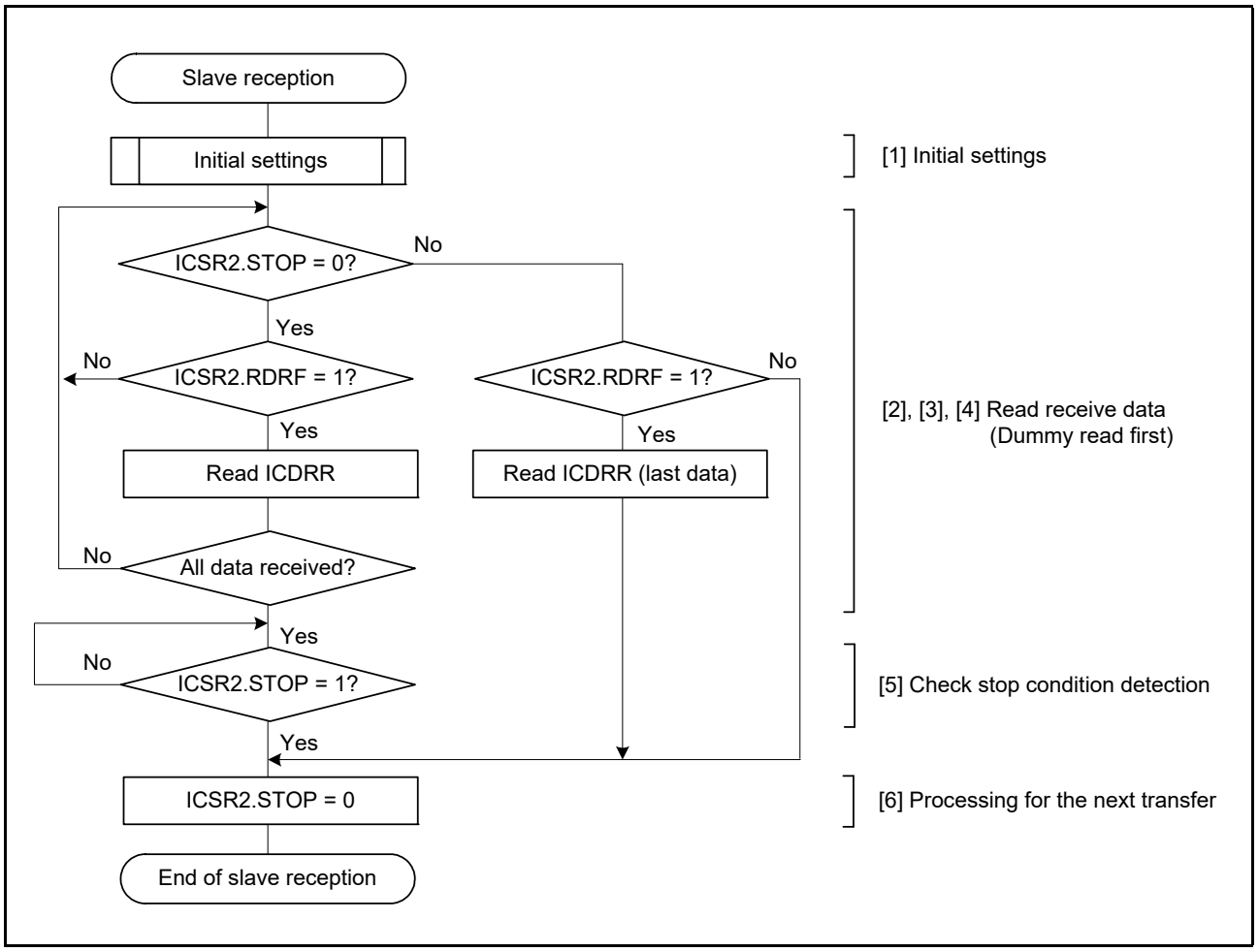


Figure 26.18 Example of Slave Reception Flowchart

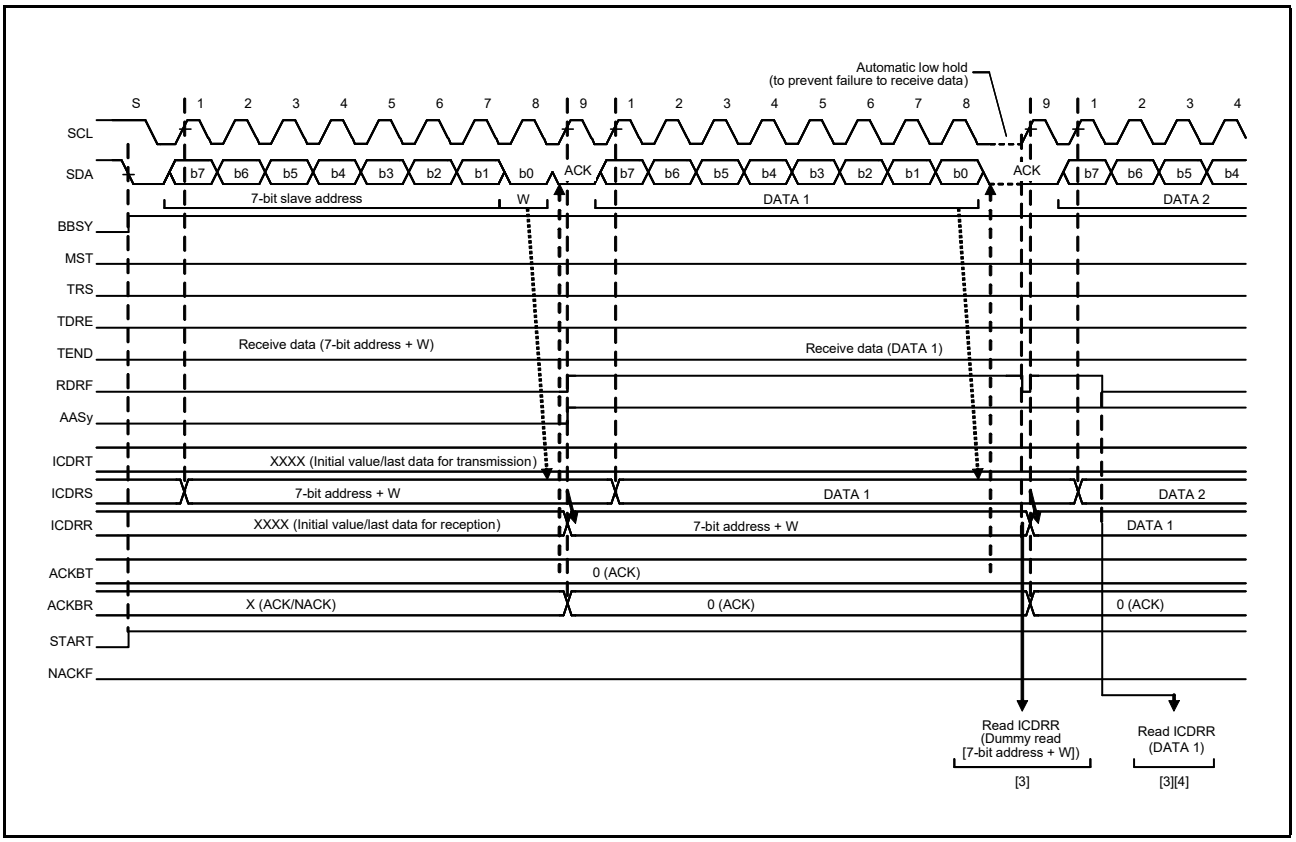


Figure 26.19 Slave Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

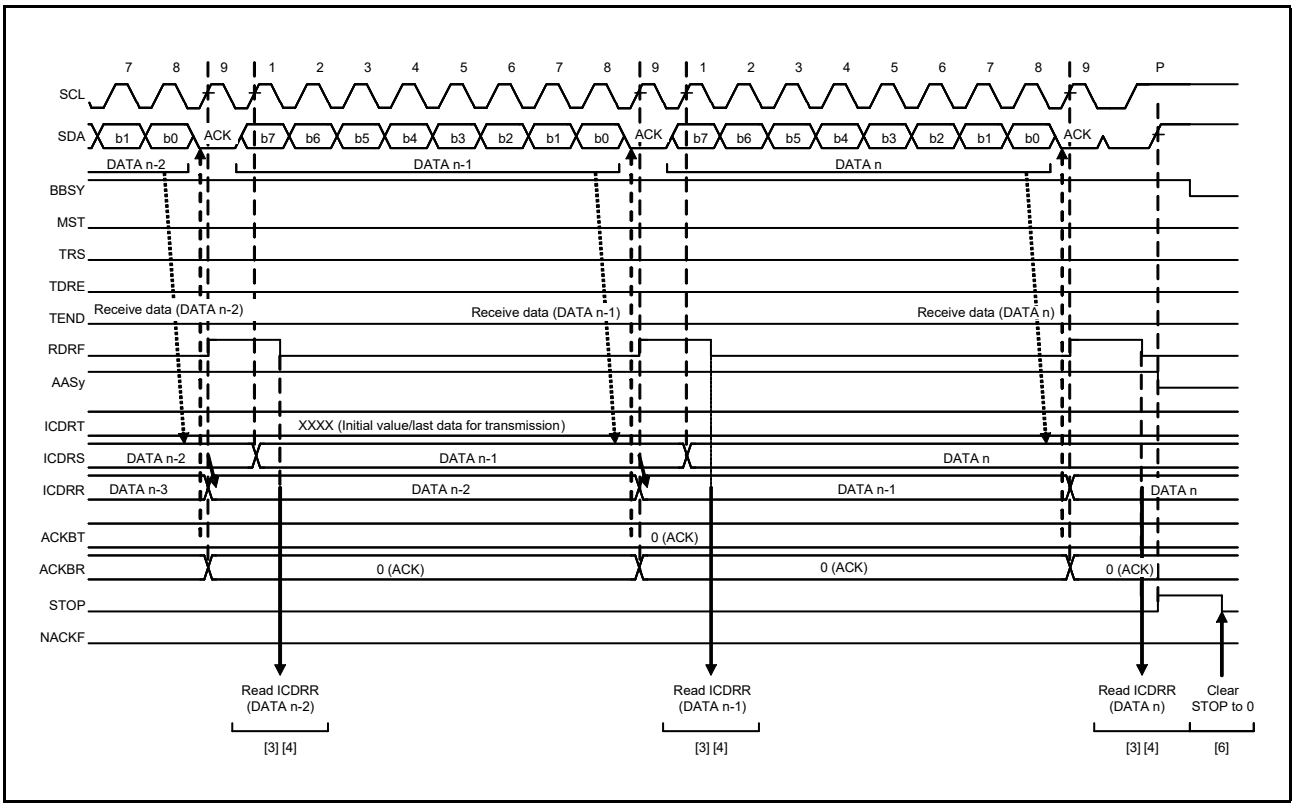


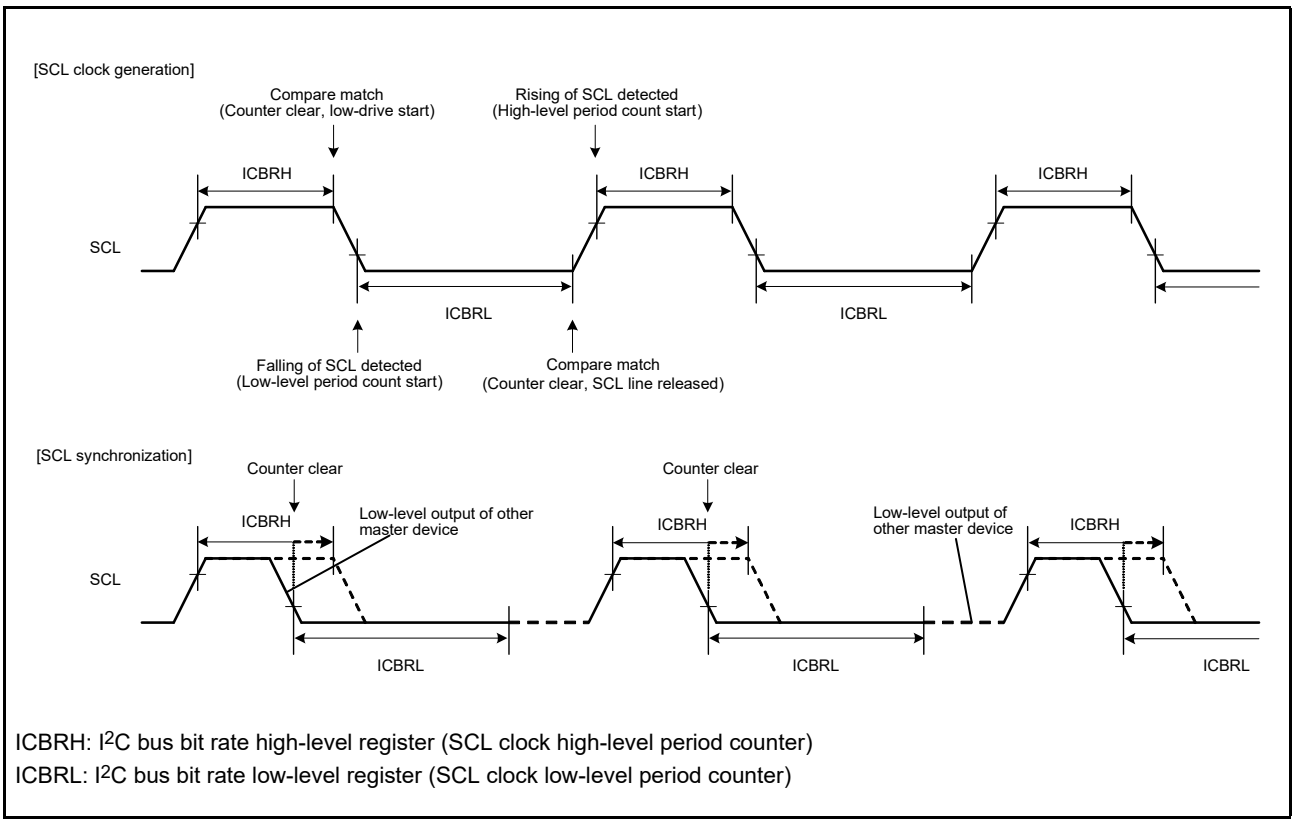
Figure 26.20 Slave Receive Operation Timing (2) (when RDRFS = 0)

26.4 SCL Synchronization Circuit

The RIIC starts counting out the value for width at high level specified in ICBRH when it detects a rising edge on the SCL line, drives the SCL line low once counting of the width at high level is complete, and then generates the SCL clock. When the RIIC detects the falling edge of the SCL line, it starts counting out the width at low level period specified in ICBRL, stops driving the SCL line (releases the line) once counting of the width at low level is complete, and then generates the SCL clock.

If multiple master devices are connected to the I²C bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since this synchronization of SCL signals must be bit by bit, the RIIC is equipped with a facility (the SCL synchronization circuit) to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCL line while in master mode.

When the RIIC has detected a rising edge on the SCL line and thus started counting out the width at high level specified in ICBRH, and the level on the SCL line falls because an SCL signal is being generated by another master device, the RIIC stops counting when it detects the falling edge, drives the level on the SCL line low, and starts counting out the width at low level specified in ICBRL. When the RIIC finishes counting out the width at low level, it stops driving the SCL line to the low level (i.e. releases the line). At this time, if the width at low level of the SCL clock signal from the other master device is longer than the width at low level set in the RIIC, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCL line has been released. When the RIIC finishes outputting the low-level period of the SCL clock, the SCL line is released and the SCL clock rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the SCLE bit in ICFER is set to 1.



ICBRH: I²C bus bit rate high-level register (SCL clock high-level period counter)
ICBRL: I²C bus bit rate low-level register (SCL clock low-level period counter)

Figure 26.21 Generation and Synchronization of the SCL Signal from the RIIC

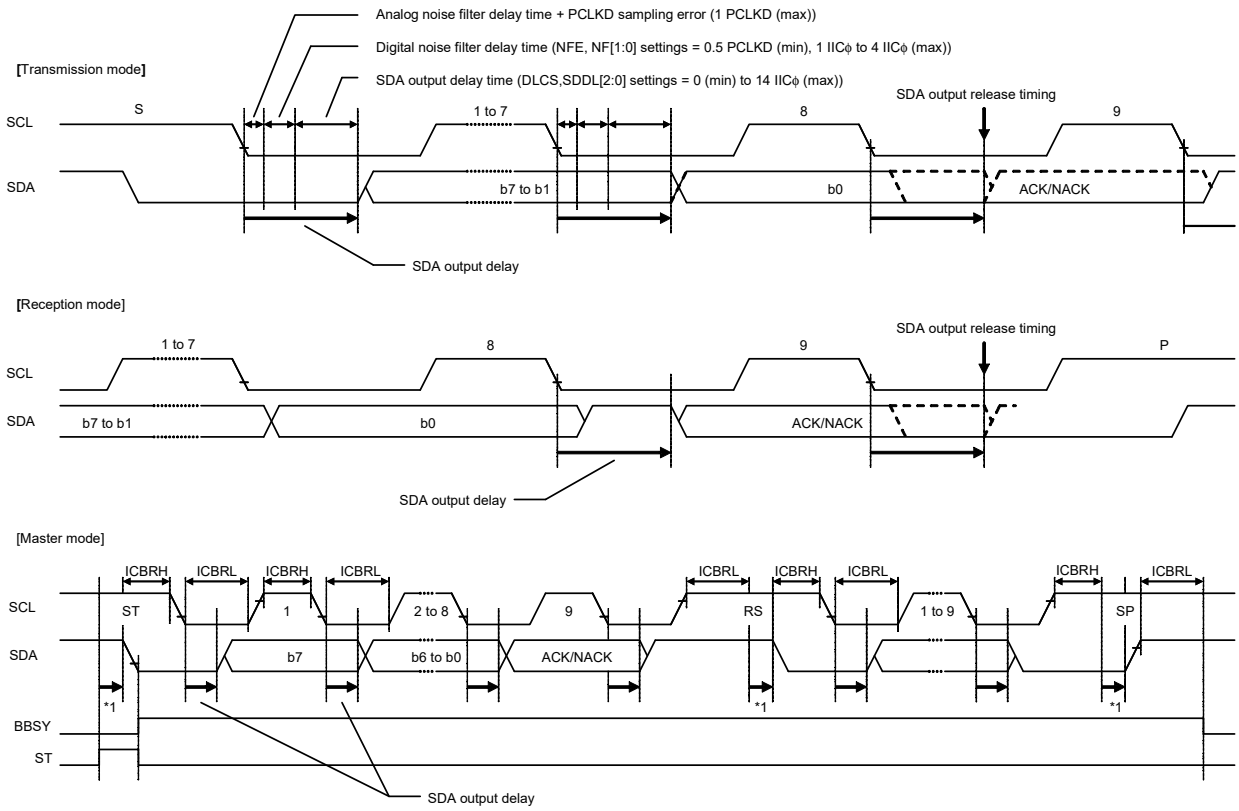
26.5 Facility for Delaying SDA Output

The RIIC module incorporates a facility for delaying output on the SDA line. The delay can be applied to all output (issuing of the start, restart, and stop conditions, data, and the ACK and NACK signals) on the SDA line.

With the SDA output delay facility, SDA output is delayed from detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval over which the SCL clock is at the low level. Doing this leads to usage with the aim of preventing erroneous operation of communications devices.

The output delay facility is enabled by setting the SDDL[2:0] bits in ICMR2 to any value other than 000b, and disabled by setting the same bits to 000b.

While the SDA output delay facility is enabled (i.e. while the SDDL[2:0] bits in ICMR2 are set to any value other than 000b), the DLCS bit in ICMR2 selects the clock source for counting by the SDA output delay counter as the internal base clock (IICϕ) for the RIIC module or as a clock signal derived by dividing the frequency of the internal base clock by 2 (IICϕ/2). The counter counts the number of cycles set in the SDDL[2:0] bits in ICMR2. After counting of the set number of cycles of delay is completed, the RIIC module places the required output (start, restart, or stop condition, data, or an ACK or NACK signal) on the SDA line.



Note 1. The output delay facility is set by the DLCS, and SDDL[2:0] bits when a start (ST), restart (RS), or stop (SP) condition is issued.

Figure 26.22 SDA Output Delay Facility

26.6 Digital Noise-Filter Circuits

The states of the SCL and SDA pins are conveyed to the internal circuitry through analog noise-filter and digital noise-filter circuits. Figure 26.23 is a block diagram of the digital noise-filter circuit.

The on-chip digital noise-filter circuit of the RIIC consists of four flip-flop circuit stages connected in series and a match-detection circuit.

The number of effective stages in the digital noise filter is selected by the NF[1:0] bits in ICMR3. The selected number of effective stages determines the noise-filtering capability as a period from one to four IICϕ cycles.

The input signal to the SCL pin (or SDA pin) is sampled on falling edges of the IICϕ signal. When the input signal level matches the output level of the number of effective flip-flop circuit stages as selected by the NF[1:0] bits in ICMR3, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is retained.

If the ratio between the frequency of the internal operating clock (PCLKD) and the transfer rate is small (e.g. data transfer at 400 Kbps with PCLKD = 4 MHz), the characteristics of the digital noise filter may lead to the elimination of needed signals as noise. In such cases, it is possible to disable the digital noise-filter circuit (by setting the ICFER.NFE bit to 0) and use only the analog noise-filter circuit.

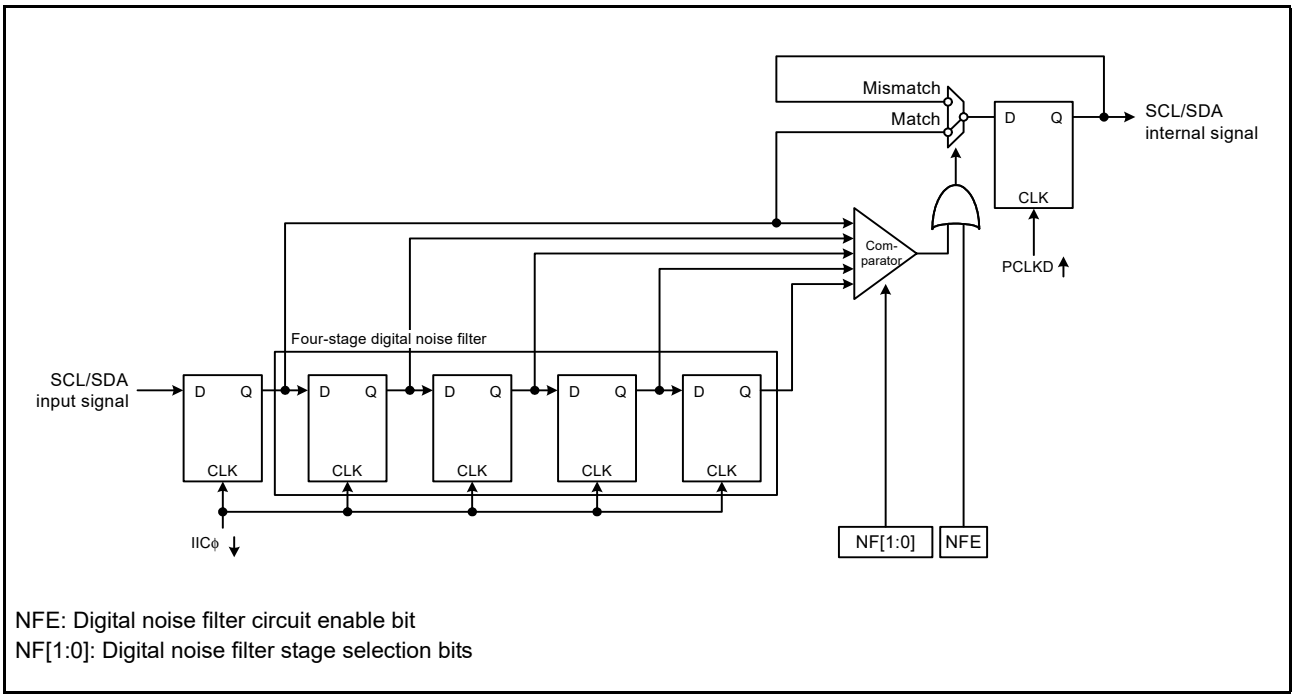


Figure 26.23 Block Diagram of Digital Noise Filter Circuit

26.7 Address Match Detection

The RIIC can set three unique slave addresses in addition to the general call address and host address, and also can set 7-bit or 10-bit slave addresses.

26.7.1 Slave-Address Match Detection

The RIIC can set three unique slave addresses, and has a slave address detection function for each unique slave address. When the SARyE bit (y = 0 to 2) in ICSER is set to 1, the slave addresses set in ICSARUy and ICSARLy (y = 0 to 2) can be detected.

When the RIIC detects a match of the set slave address, the corresponding AASy flag (y = 0 to 2) in ICSR1 is set to 1 at the rising edge of the ninth SCL clock cycle, and the RDRF flag in ICSR2 or the TDRE flag in ICSR2 is set to 1 by the following R/W# bit. This causes a receive data full interrupt (RXI) or transmit data empty interrupt (TXI) to be generated. The AASy flag is used to identify which slave address has been specified.

Figure 26.24 to Figure 26.26 show the AASy flag set timing in three cases.

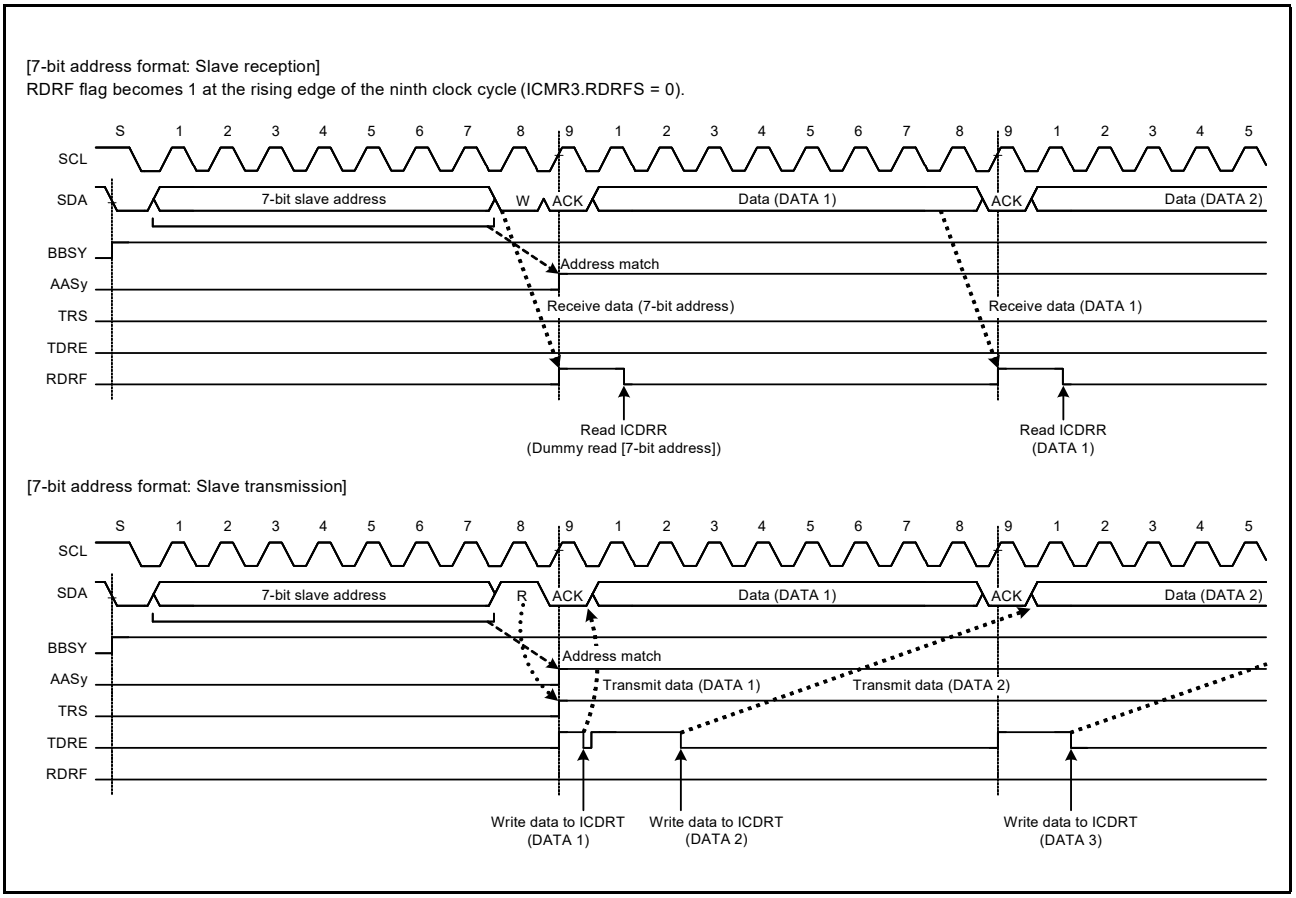


Figure 26.24 AASy Flag Set Timing with 7-Bit Address Format Selected

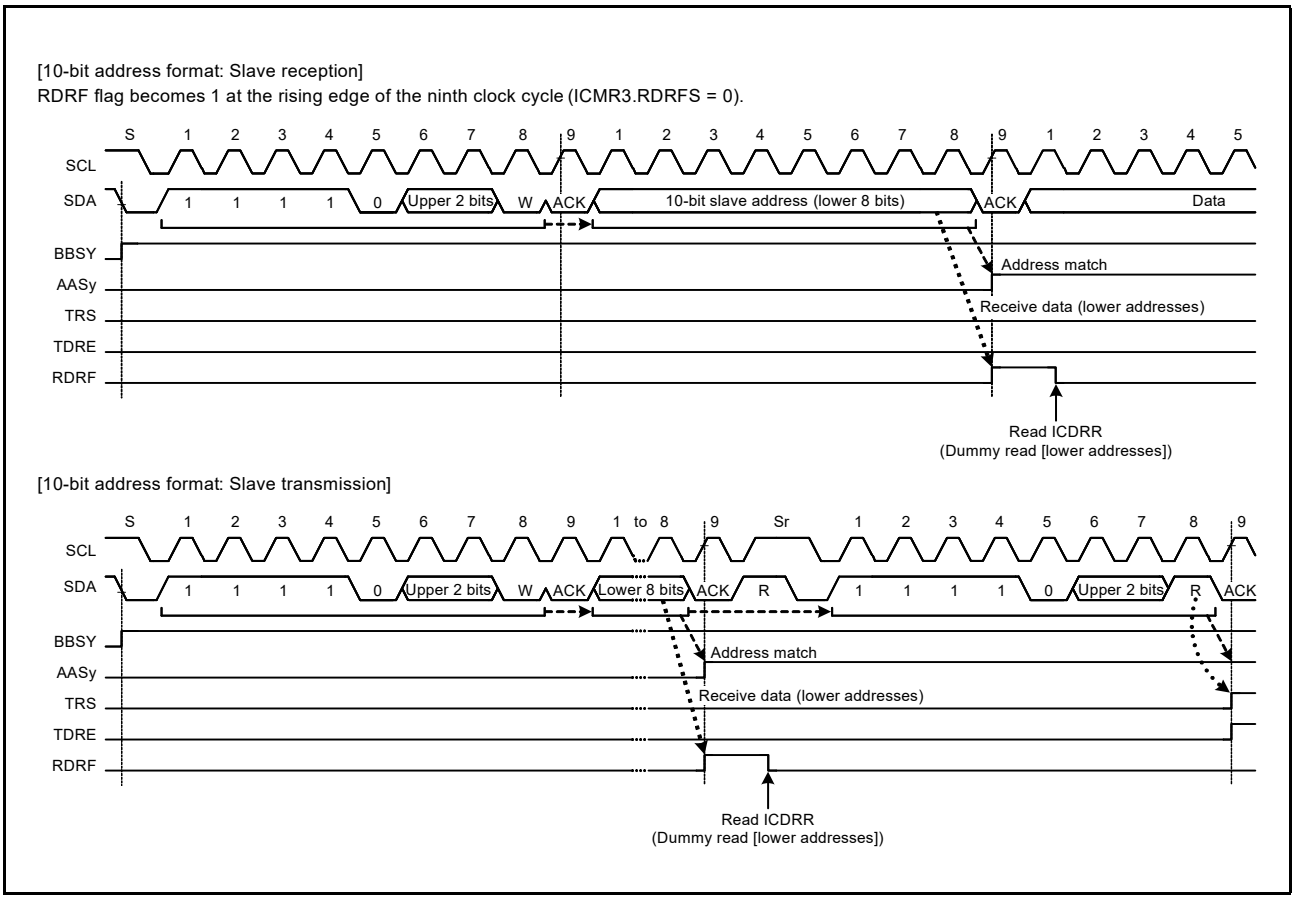


Figure 26.25 AASy Flag Set Timing with 10-Bit Address Format Selected

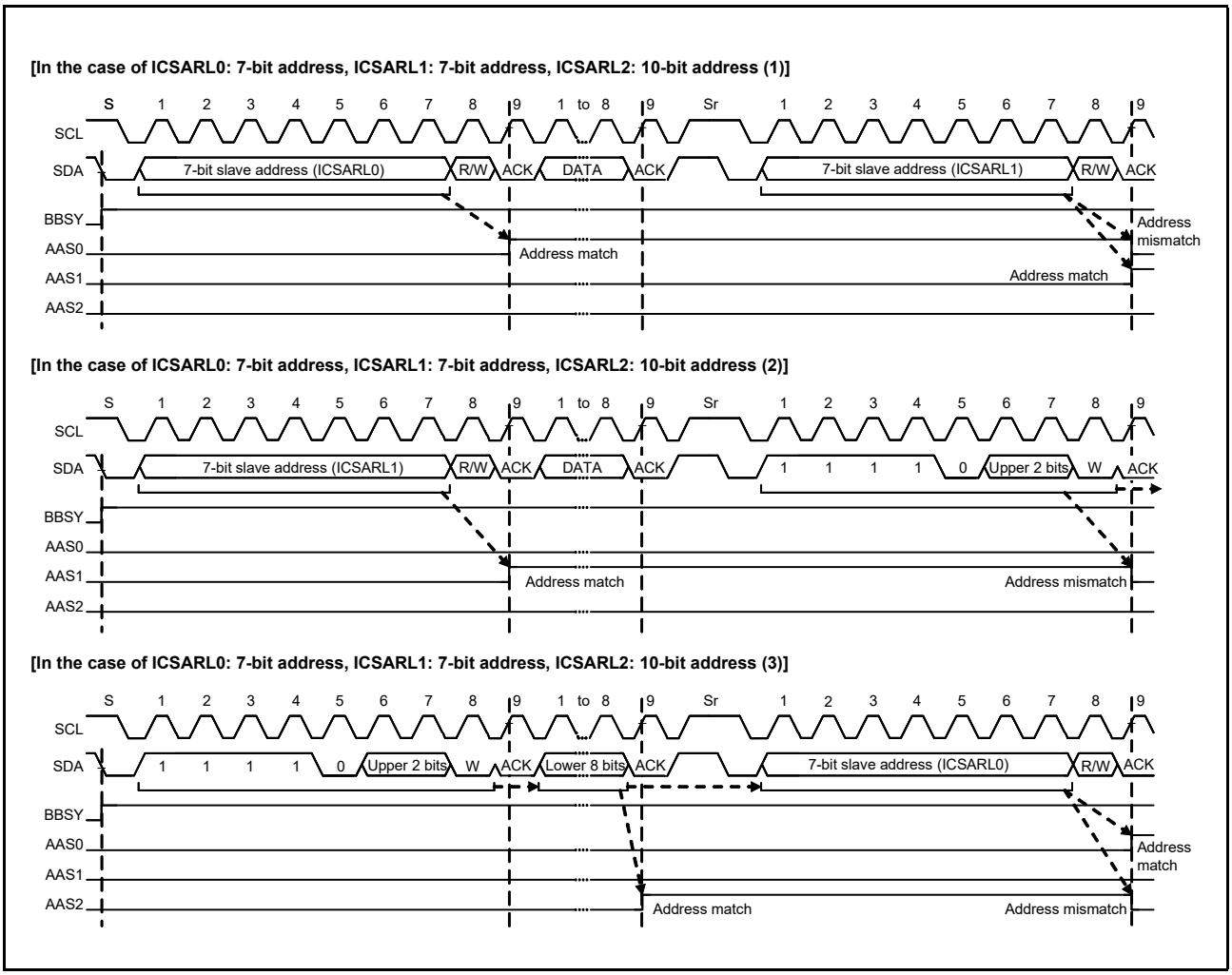


Figure 26.26 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed

26.7.2 Detection of the General Call Address

The RIIC has a facility for detecting the general call address (0000 000b + 0 [W]). This is enabled by setting the GCAE bit in ICSER to 1.

If the address received after a start or restart condition is issued is 0000 000b + 1[R] (start byte), the RIIC recognizes this as the address of a slave device with an “all-zero” address but not as the general call address.

When the RIIC detects the general call address, both the GCA flag in ICSR1 and the RDRF flag in ICSR2 are set to 1 on the rising edge of the ninth cycle of SCL clock. This leads to the generation of a receive data full interrupt (RXI). The value of the GCA flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

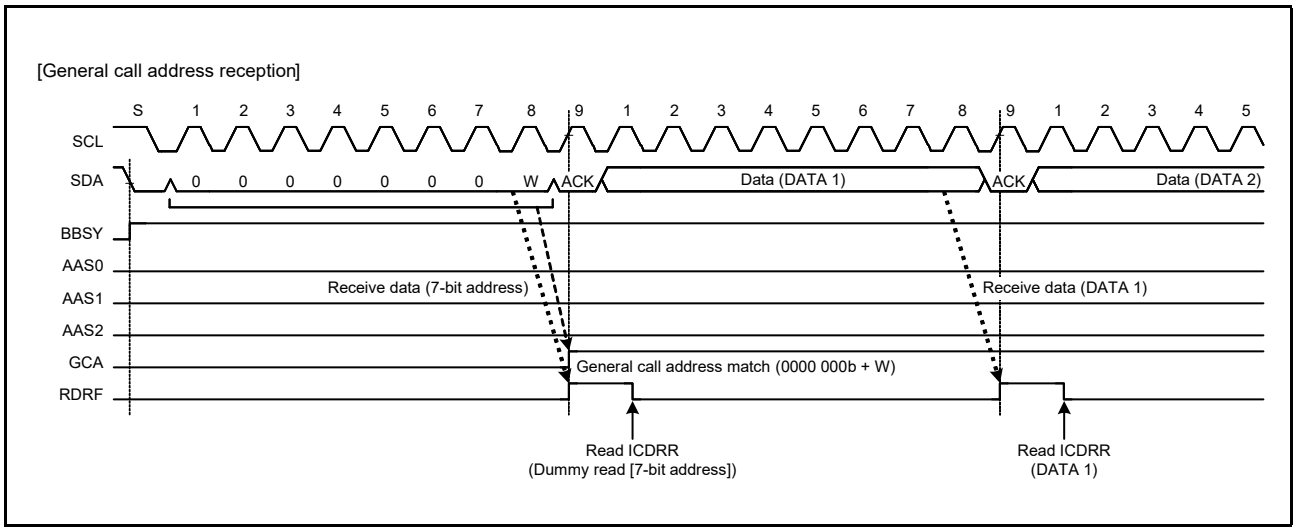


Figure 26.27 Timing of GCA Flag Setting during Reception of General Call Address

26.7.3 Device-ID Address Detection

The RIIC module has a facility for detecting device-ID addresses conforming with the I2C bus specification (Rev. 03). When the RIIC receives 1111 100b as the first byte after a start condition or restart condition was issued with the DIDE bit in ICSER set to 1, the RIIC recognizes the address as a device ID, sets the DID flag in ICSR1 to 1 on the rising edge of the eighth SCL clock cycle when the following R/W# bit is 0, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, the RIIC sets the corresponding AASy flag (y = 0 to 2) in ICSR1 to 1.

After that, when the first byte received after a start or restart condition is issued matches the device ID address (1111 100b) again and the following R/W# bit is 1, the RIIC does not compare the second and subsequent bytes and sets the ICSR2.TDRE flag to 1.

In the device-ID address detection function, the RIIC sets the DID flag to 0 if a match with the RIIC's own slave address is not obtained or a match with the device ID address is not obtained after a match with the RIIC's own slave address and the detection of a restart condition. In this case, if the first byte after detection of a start or restart condition matches the device ID address (1111 100b) and the R/W# bit is 0, the RIIC sets the DID flag to 1 and compares the second and subsequent bytes with the RIIC's slave address. If the R/W# bit is 1, the DID flag holds the previous value and the RIIC does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DID flag after confirming that TDRE = 1.

Furthermore, prepare the device-ID fields (three bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details of the information that must be included in device-ID fields, contact NXP Semiconductors.

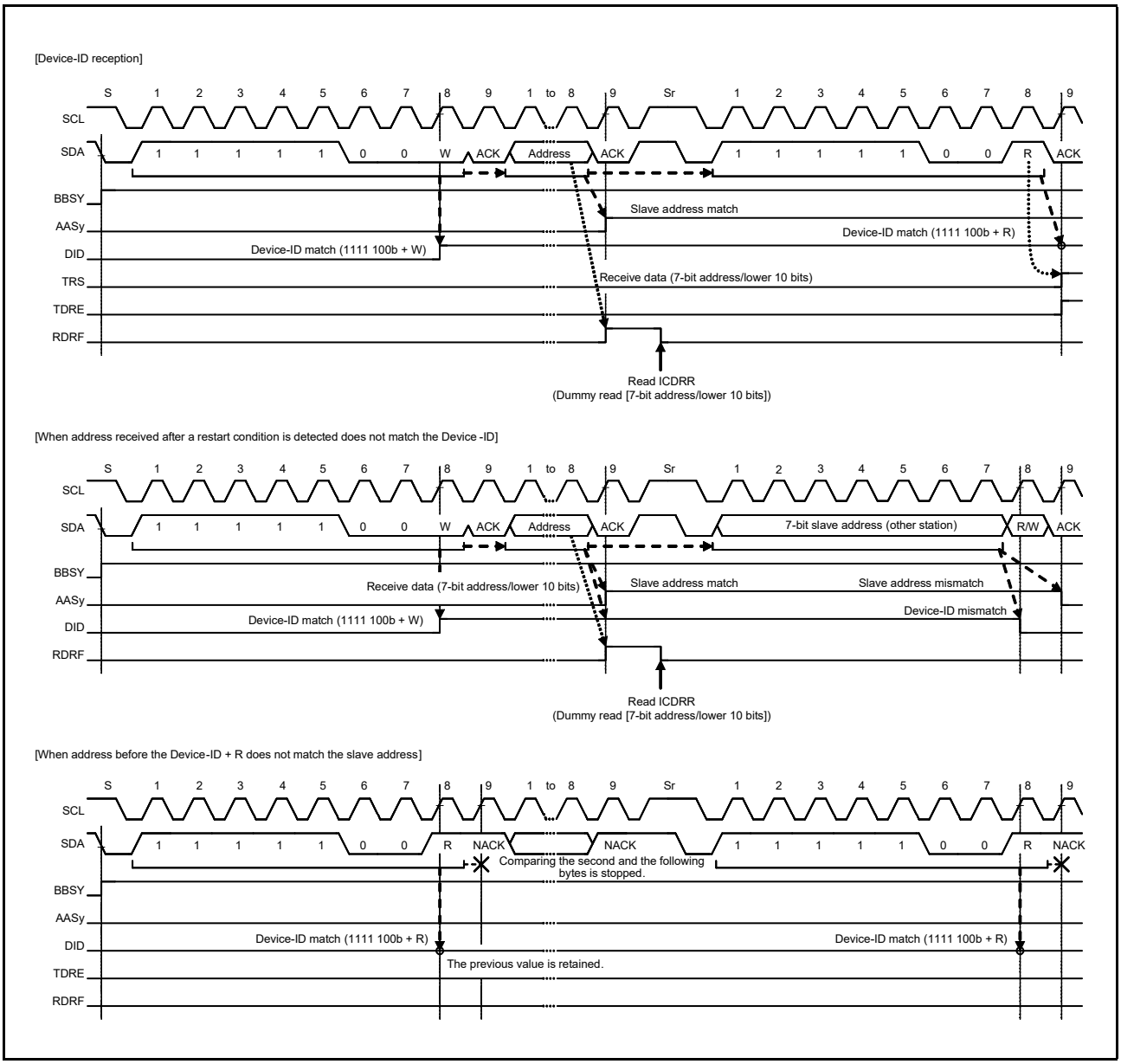


Figure 26.28 AASy/DID Flag Set/Clear Timing during Reception of Device-ID

26.8 Automatic Low-Hold Function for SCL

26.8.1 Function to Prevent Wrong Transmission of Transmit Data

If the shift register (ICDRS) is empty when data have not been written to the I2C bus transmit data register (ICDRT) with the RIIC in transmission mode (TRS bit = 1 in ICCR2), the SCL line is automatically held at the low level over the intervals shown below. This low-hold period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

Master transmission mode

- Low-level interval after a start condition or restart condition is issued
- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

Slave transmission mode

- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

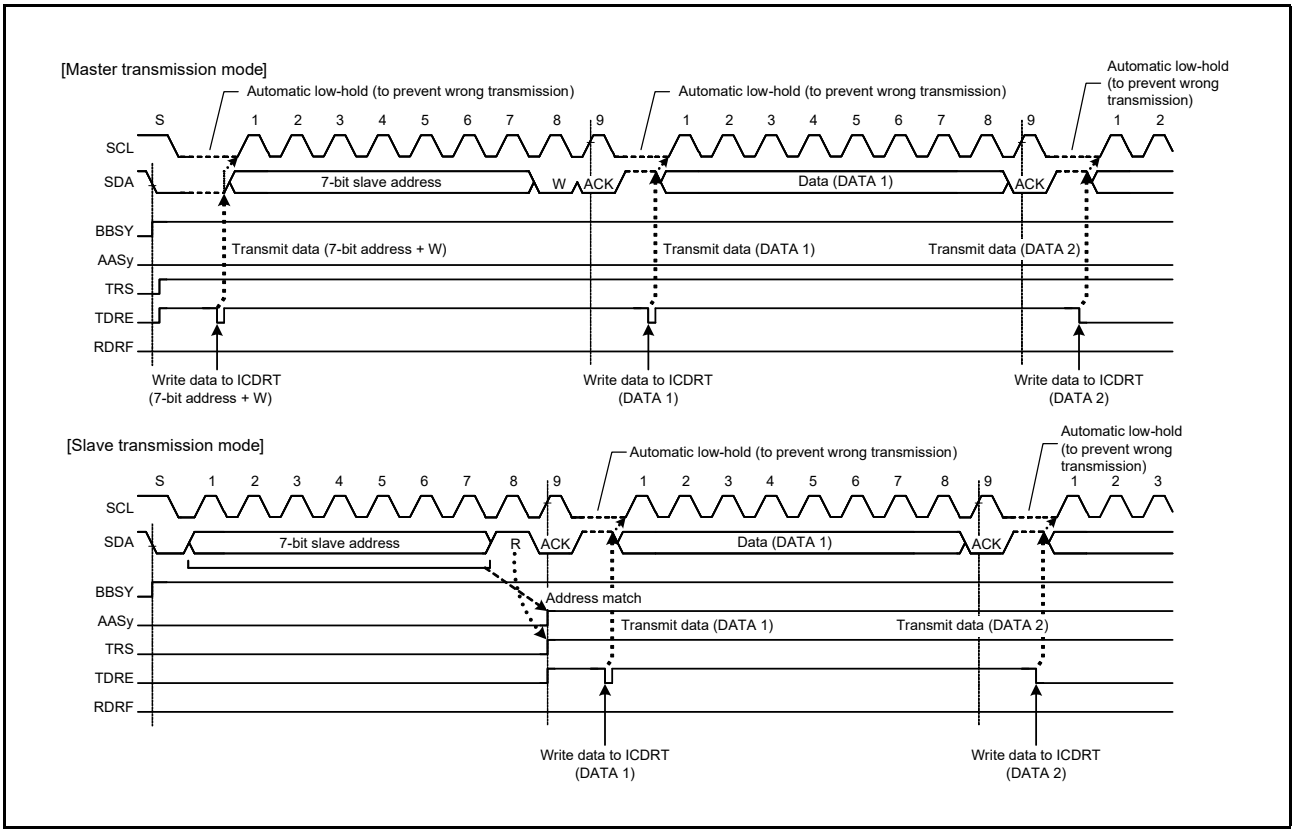


Figure 26.29 Automatic Low-Hold Operation in Transmission Mode

26.8.2 NACK Reception Transfer Suspension Function

The RIIC has a function to suspend transfer operation when NACK is received in transmission mode (TRS bit = 1 in ICCR2). This function is enabled when the NACKE bit in ICFER is set to 1 (transfer suspension enabled). If the next transmit data has already been written (TDRE flag = 0 in ICSR2) when NACK is received, next data transmission at the falling edge of the ninth SCL clock cycle is automatically suspended. This prevents the SDA line output level from being held low when the MSB of the next transmit data is 0.

If the transfer operation is suspended by this function (NACKF flag = 1 in ICSR2), transmit operation and receive operation are discontinued. To restore transmit/receive operation, be sure to set the NACKF flag to 0. In master transmission mode, set the NACKF flag to 0, issue a restart or stop condition, and then issue a start condition again.

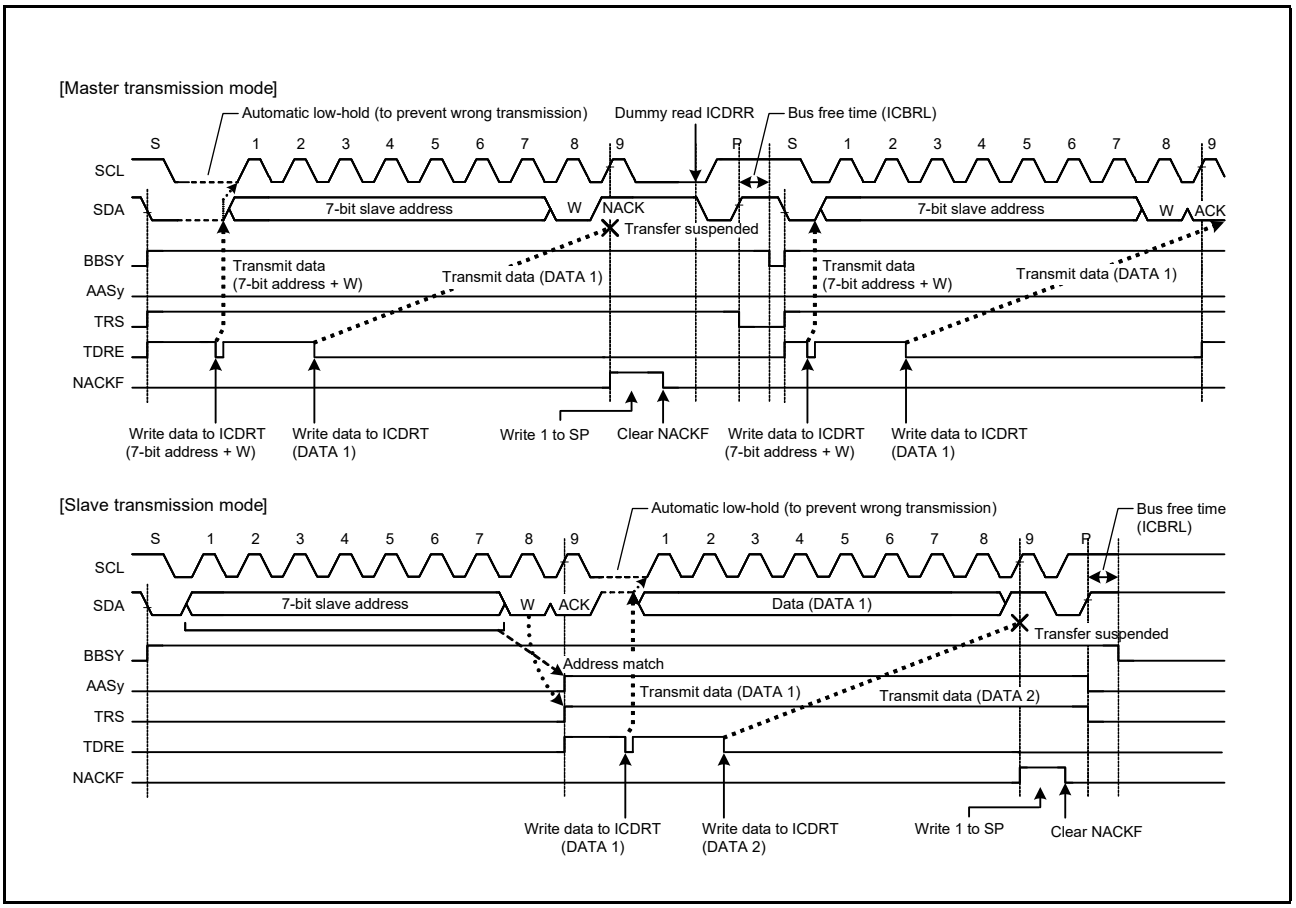


Figure 26.30 Suspension of Data Transfer When NACK is Received (NACKE = 1)

26.8.3 Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (ICDRR) read is delayed for a period of one transfer frame or more with receive data full (RDRF flag = 1 in ICSR2) in reception mode (TRS = 0 in ICCR2), the RIIC holds the SCL line low automatically immediately before the next data is received to prevent failure to receive data.
This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, the RIIC's own slave address is designated after a stop condition is issued. This function does not disturb other communication because the RIIC does not hold the SCL line low when a mismatch with its own slave address occurs after a stop condition is issued.
Sections in which the SCL line is held low can be selected with a combination of the WAIT and RDRFS bits in ICMR3.

(1) 1-Byte Receive Operation and Automatic Low-Hold Function Using the WAIT Bit

When the WAIT bit in ICMR3 is set to 1, the RIIC performs 1-byte receive operation using the WAIT bit function. When the ICMR3.RDRFS bit is 0, the ACKBT bit value in ICMR3 is automatically sent to the acknowledge bit in the period from the falling edge of the eighth SCL clock cycle to the falling edge of the ninth SCL clock cycle. When the falling edge of the ninth SCL clock cycle is detected, the SCL line is automatically held low by the WAIT bit function. This low-hold is released by reading data from ICDRR, which enables bytewise receive operation.
The WAIT bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master reception mode or slave reception mode.

(2) 1-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the RDRFS Bit

When the RDRFS bit in ICMR3 is set to 1, the RIIC performs 1-byte receive operation using the RDRFS bit function. When the RDRFS bit is set to 1, the RDRF flag (receive data full) in ICSR2 is set to 1 at the rising edge of the eighth SCL clock cycle, and the SCL line is automatically held low at the falling edge of the eighth SCL clock cycle. This low-hold is released by writing a value to the ACKBT bit in ICMR3, but cannot be released by reading data from ICDRR, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.
The RDRFS bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master reception mode or slave reception mode.

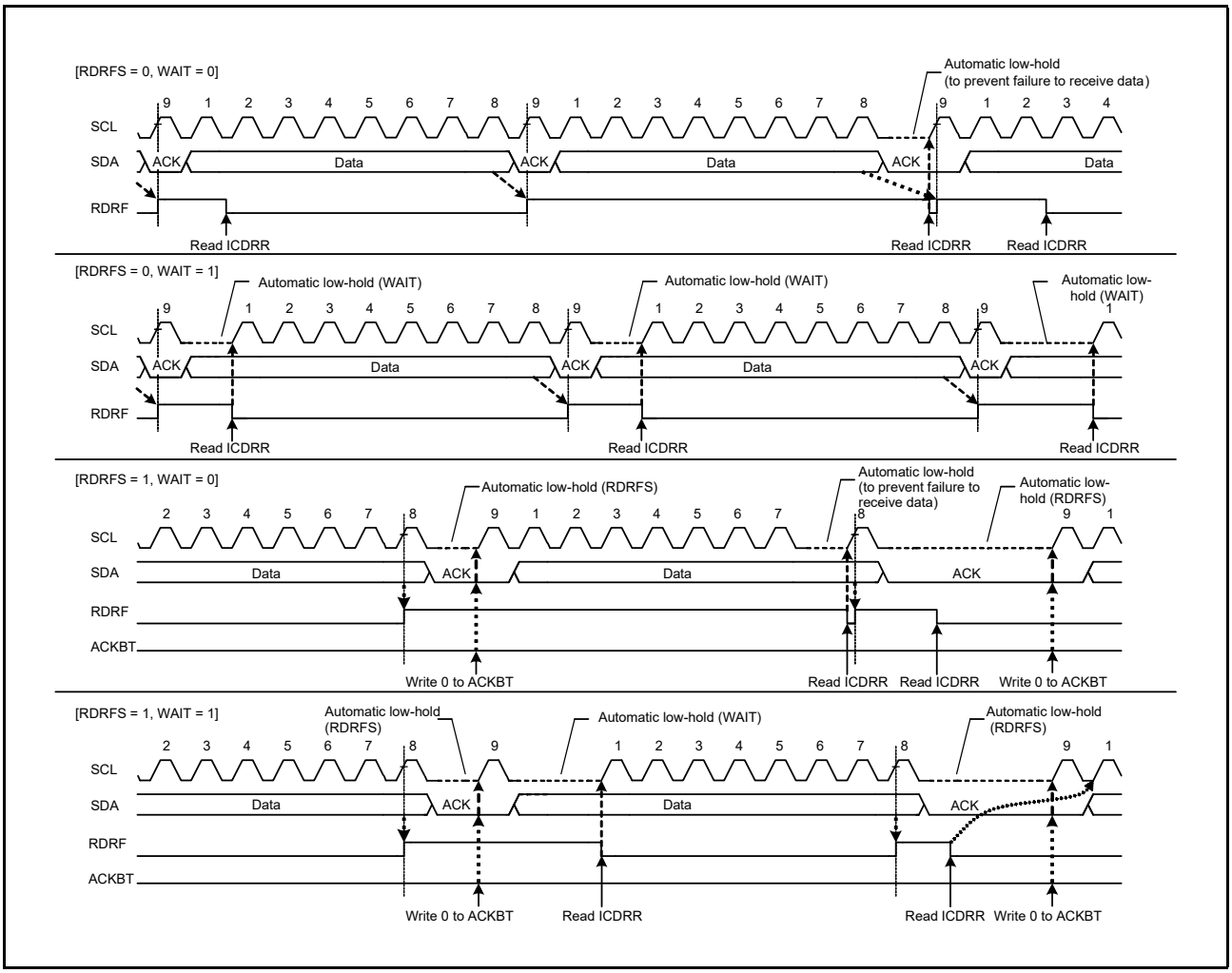


Figure 26.31 Automatic Low-Hold Operation in Reception Mode (Using RDRFS and WAIT Bits)

26.9 Arbitration-Lost Detection Functions

In addition to the normal arbitration-lost detection function defined by the I2C bus standard, the RIIC has functions to prevent double-issue of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration-lost in slave transmission mode.

26.9.1 Master Arbitration-Lost Detection (MALE Bit)

The RIIC drives the SDA line low to issue a start condition. However, if the SDA line has already been driven low by another master device issuing a start condition, the RIIC regards its own issuing of a start condition as an error and considers this a loss in arbitration, so priority is given to transfer by the other master device. Similarly, if a request to issue a start condition is made by setting the ST bit in ICCR2 to 1 while the bus is busy (BBSY flag = 1 in ICCR2), the RIIC regards this as a double-issuing-of-start-condition error and considers itself to have lost in arbitration, thus preventing a failure of transfer due to issuing of a start condition while transfer is in progress.
When a start condition is issued successfully, the data for transmission including the address bits (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state, and the low level is detected on the SDA line), the RIIC loses in arbitration.
After a loss in arbitration of mastership, the RIIC immediately enters slave reception mode. If a slave address (including the general call address) matches its own address at this time, the RIIC continues in slave operation.
A loss in arbitration of mastership is detected when the following conditions are met while the MALE bit in ICFER is 1 (master arbitration-lost detection enabled).
[Master arbitration-lost conditions]

- Non-matching of the internal level for output on SDA and the level on the SDA line after a start condition was issued by setting the ST bit in ICCR2 to 1 while the BBSY flag in ICCR2 was set to 0 (erroneous issuing of a start condition)
- Setting of the ST bit in ICCR2 to 1 (start condition double-issue error) while the BBSY flag in ICCR2 is set to 1
- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDA line in master transmission mode (MST and TRS bits = 11b in ICCR2)

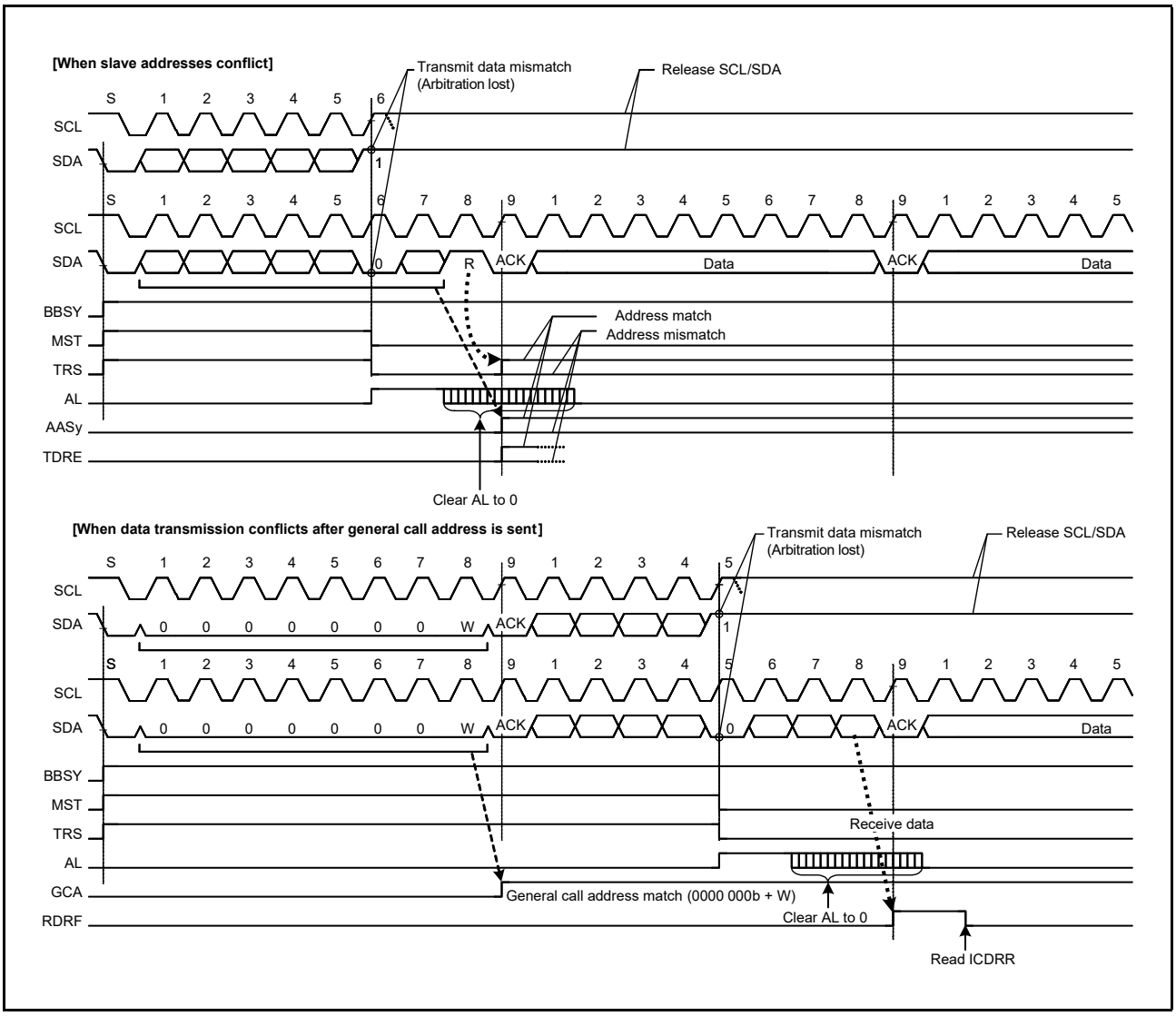


Figure 26.32 Examples of Master Arbitration-Lost Detection (MALE = 1)

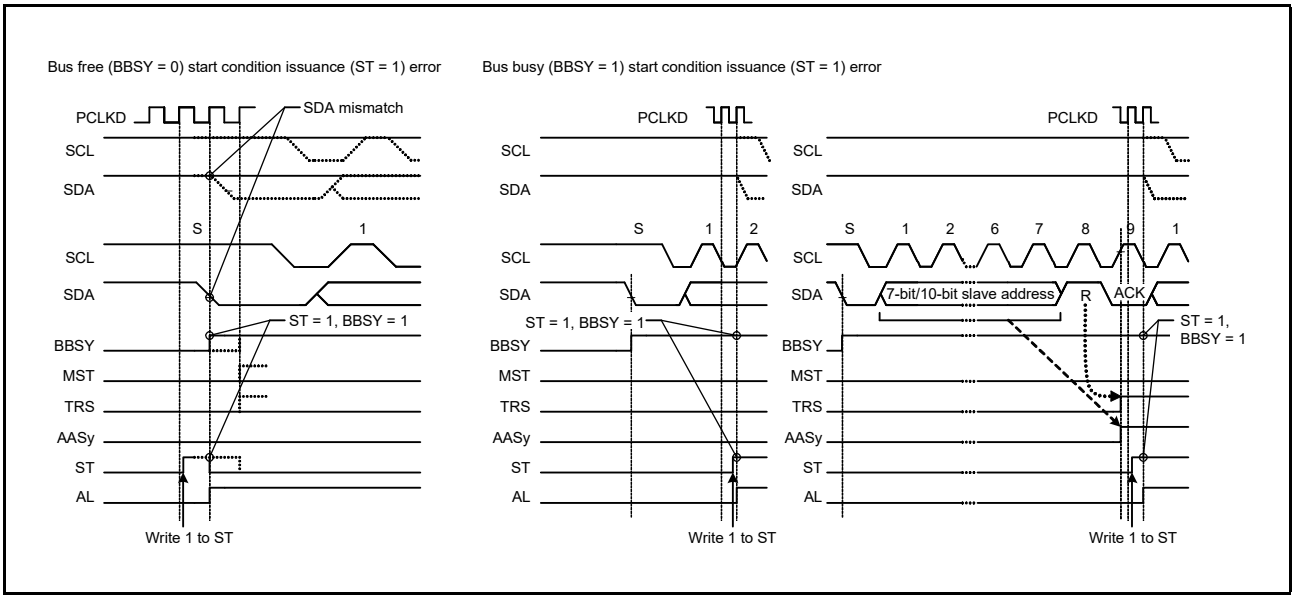


Figure 26.33 Arbitration-Lost When a Start Condition is Issued (MALE = 1)

26.9.2 Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit)

The RIIC has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDA line (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state, and the low level is detected on the SDA line) during transmission of NACK in reception mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device. Figure 26.34 shows an example of arbitration-lost detection during transmission of NACK.

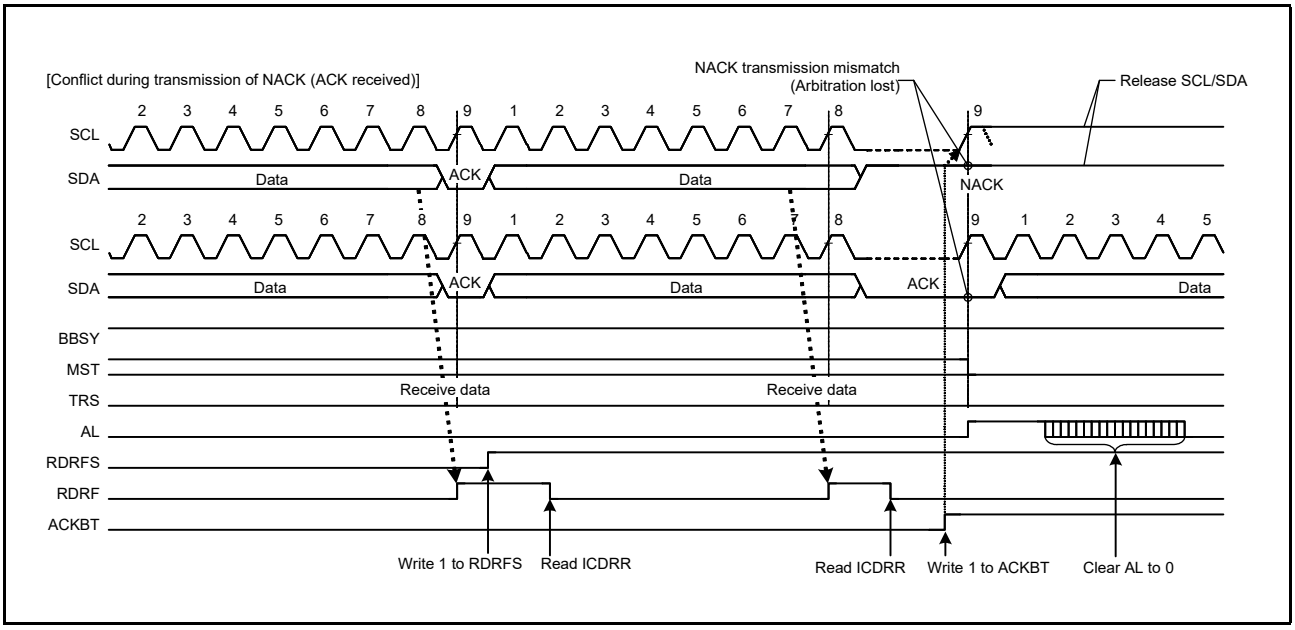


Figure 26.34 Example of Arbitration-Lost Detection during Transmission of NACK (NALE = 1)

The following explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives two bytes of data from the slave device, and master B receives four bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. Here, master A sends NACK when it has received two final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received necessary four bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and issues a stop condition. Therefore, the issuance of the stop condition conflicts with the SCL clock output of master B, which disturbs communication.

When the RIIC receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If arbitration is lost during transmission of NACK, the RIIC immediately cancels the slave match condition and enters slave reception mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

The RIIC detects arbitration-lost during transmission of NACK when the following condition is met with the NALE bit in ICFER set to 1 (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

- When the internal SDA output level does not match the SDA line (ACK is received) during transmission of NACK (ACKBT bit = 1 in ICMR3)

26.9.3 Slave Arbitration-Lost Detection (SALE Bit)

The RIIC has a function to cause arbitration to be lost if the data for transmission (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state, and the low level is detected on the SDA line) in slave transmission mode.
When it loses slave arbitration, the RIIC is immediately released from the slave-matched state and enters slave reception mode.
The RIIC detects slave arbitration-lost when the following condition is met with the SALE bit in ICFER set to 1 (slave arbitration-lost detection enabled).

[Condition for slave arbitration-lost]

- When transmit data excluding acknowledge (internal SDA output level) does not match the SDA line in slave transmission mode (MST and TRS bits = 01b in ICCR2)

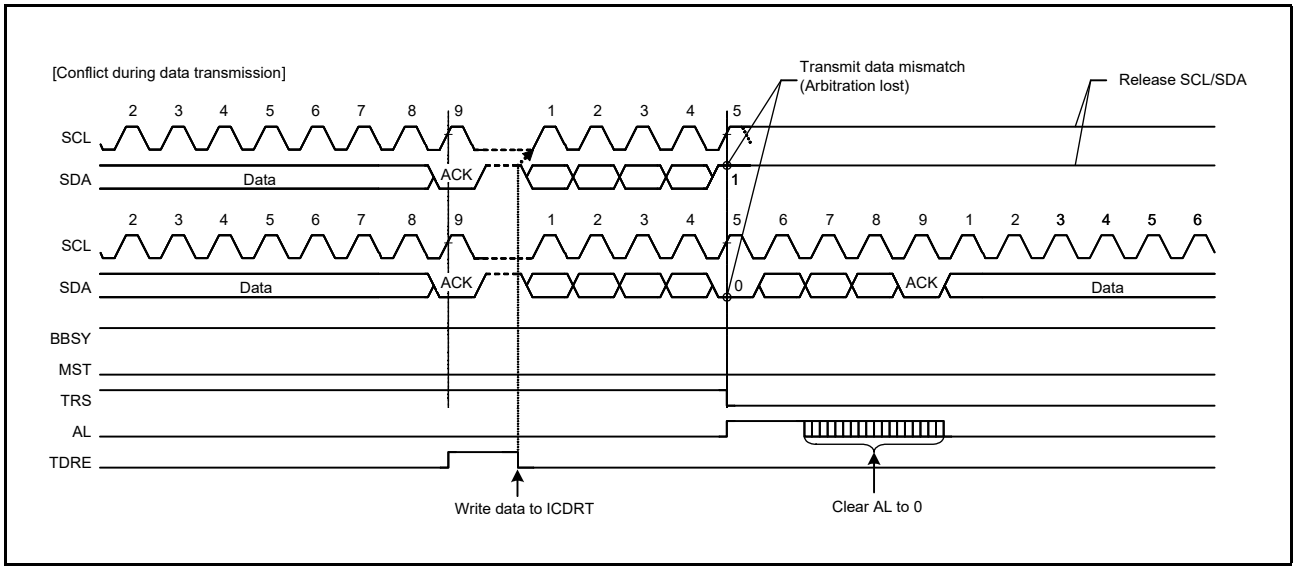


Figure 26.35 Example of Slave Arbitration-Lost Detection (SALE = 1)

26.10 Start Condition/Restart Condition/Stop Condition Issuing Function

26.10.1 Issuing a Start Condition

The RIIC issues a start condition when the ST bit in ICCR2 is set to 1.
When the ST bit is set to 1, a start condition issuance request is made and the RIIC issues a start condition when the BBSY flag in ICCR2 is 0 (bus free state). When a start condition is issued normally, the RIIC automatically shifts to the master transmission mode.
A start condition is issued in the following sequence.

[Start condition issuance]
(1) Drive the SDA line low (high level to low level).
(2) Ensure the start condition hold time set in ICBRH.
(3) Drive the SCL line low (high level to low level).
(4) Detect low level of the SCL line and ensure the low-level period of SCL line set in ICBRL.

26.10.2 Issuing a Restart Condition

The RIIC issues a restart condition when the RS bit in ICCR2 is set to 1.
When the RS bit is set to 1, a restart condition issuance request is made and the RIIC issues a restart condition when the BBSY flag in ICCR2 is 1 (bus busy state) and the MST bit in ICCR2 is 1 (master mode).
A restart condition is issued in the following sequence.

[Restart condition issuance]
(1) Release the SDA line.
(2) Ensure the low-level period of SCL line set in ICBRL.
(3) Release the SCL line (low level to high level).
(4) Detect a high level of the SCL line and ensure the restart condition setup time set in ICBRL.
(5) Drive the SDA line low (high level to low level).
(6) Ensure the restart condition hold time set in ICBRH.
(7) Drive the SCL line low (high level to low level).
(8) Detect a low level of the SCL line and ensure the low-level period of SCL line set in ICBRL.

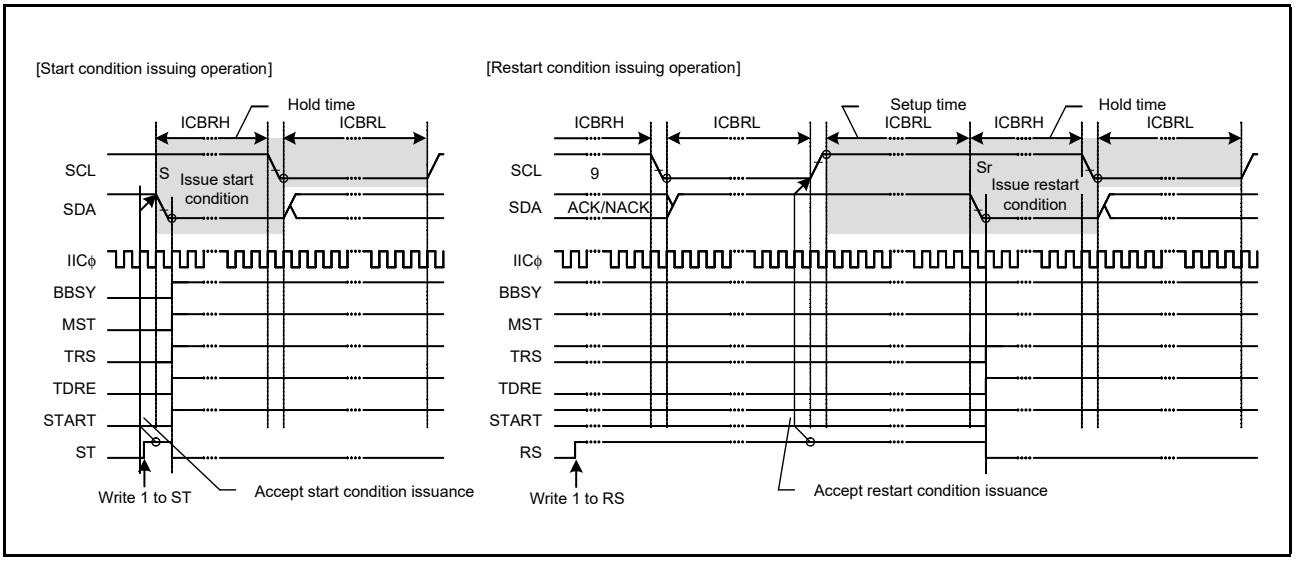


Figure 26.36 Start Condition/Restart Condition Issue Timing (ST and RS Bits)

26.10.3 Issuing a Stop Condition

The RIIC issues a stop condition when the SP bit in ICCR2 is set to 1.
When the SP bit is set to 1, a stop condition issuance request is made and the RIIC issues a stop condition when the BBSY flag in ICCR2 is 1 (bus busy state) and the MST bit in ICCR2 is 1 (master mode).
A stop condition is issued in the following sequence.

[Stop condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the low-level period of SCL line set in ICBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the stop condition setup time set in ICBRH.
- Release the SDA line (low level to high level).
- Ensure the bus free time set in ICBRL.
- Clear the BBSY flag to 0 (to release the bus mastership).

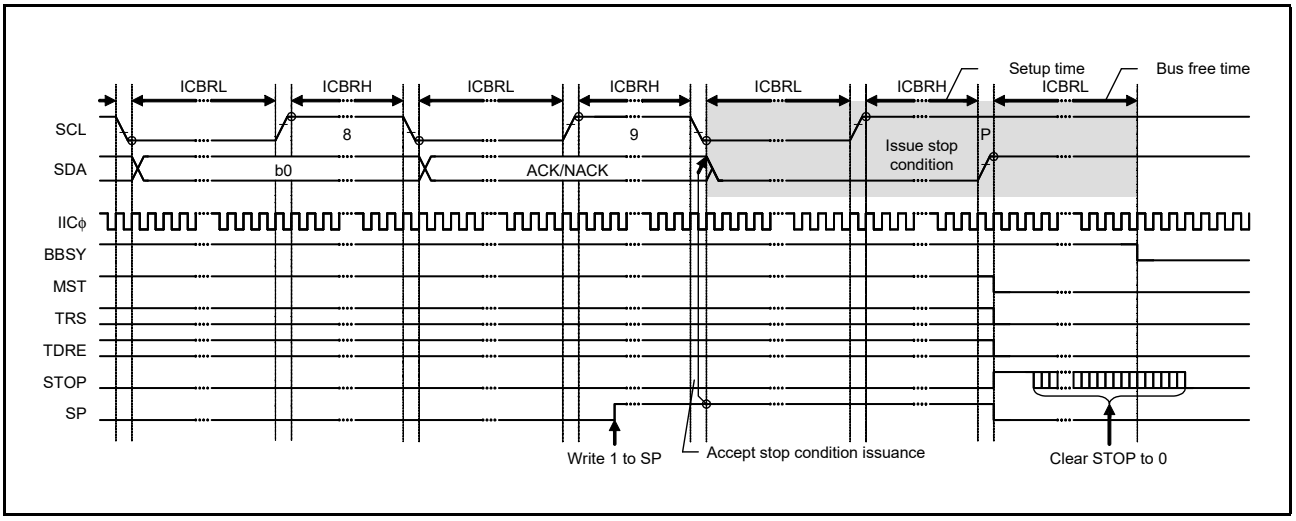


Figure 26.37 Stop Condition Issue Timing (SP Bit)

26.11 Bus Hanging

If the clock signals from the master and slave devices go out of synchronization due to noise or other factors, the I²C bus might hang with a fixed level on the SCL line and/or SDA line.
As measures against the bus hanging, the RIIC has a timeout function to detect hanging by monitoring the SCL line, a function for the output of an extra SCL clock cycle to release the bus from a hung state due to clock signals being out of synchronization, the RIIC reset function, and internal reset function.
By checking the SCLO, SDAO, SCLI, and SDAI bits in ICCR1, it is possible to see whether the RIIC or its partner in communications is placing the low level on the SCL or SDA line.

26.11.1 Timeout Function

The RIIC includes a timeout function for detecting when the SCL line has been stuck longer than the predetermined time. The RIIC can detect an abnormal bus state by monitoring that the SCL line is stuck low or high for a predetermined time. The timeout function monitors the SCL line state and counts the low-level period or high-level period using the internal counter. The timeout function resets the internal counter each time the SCL line changes (rising or falling), but continues to count unless the SCL line changes. If the internal counter overflows due to no SCL line change, the RIIC can detect the timeout and report the bus hung state.
This timeout function is enabled when the ICFER.TMOE bit is 1. It detects a hung state that the SCL line is stuck low or high during the following conditions:

- The bus is busy (ICCR2.BBSY flag is 1) in master mode (ICCR2.MST bit is 1).
- The RIIC's own slave address is detected (ICSR1 register is not 00h) and the bus is busy (ICCR2.BBSY flag is 1) in slave mode (ICCR2.MST bit is 0).
- The bus is free (ICCR2.BBSY flag is 0) while generation of a start condition is requested (ICCR2.ST bit is 1).

The internal counter of the timeout function works using the internal reference clock (IICϕ) set by the CKS[2:0] bits in ICMR1 as a count source. It functions as a 16-bit counter when long mode is selected (TMOS bit = 0 in ICMR2) or a 14-bit counter when short mode is selected (TMOS bit = 1).
The SCL line level (low/high or both levels) during which this counter is activated can be selected by the setting of the TMOH and TMOL bits in ICMR2. If both TMOL and TMOH bits are set to 0, the internal counter does not work.

Note: When the timeout function for detecting is to be used, see section 26.2.4, I²C Bus Mode Register 2 (ICMR2), section 26.3, Operation, and section 26.3.2, Initial Settings.

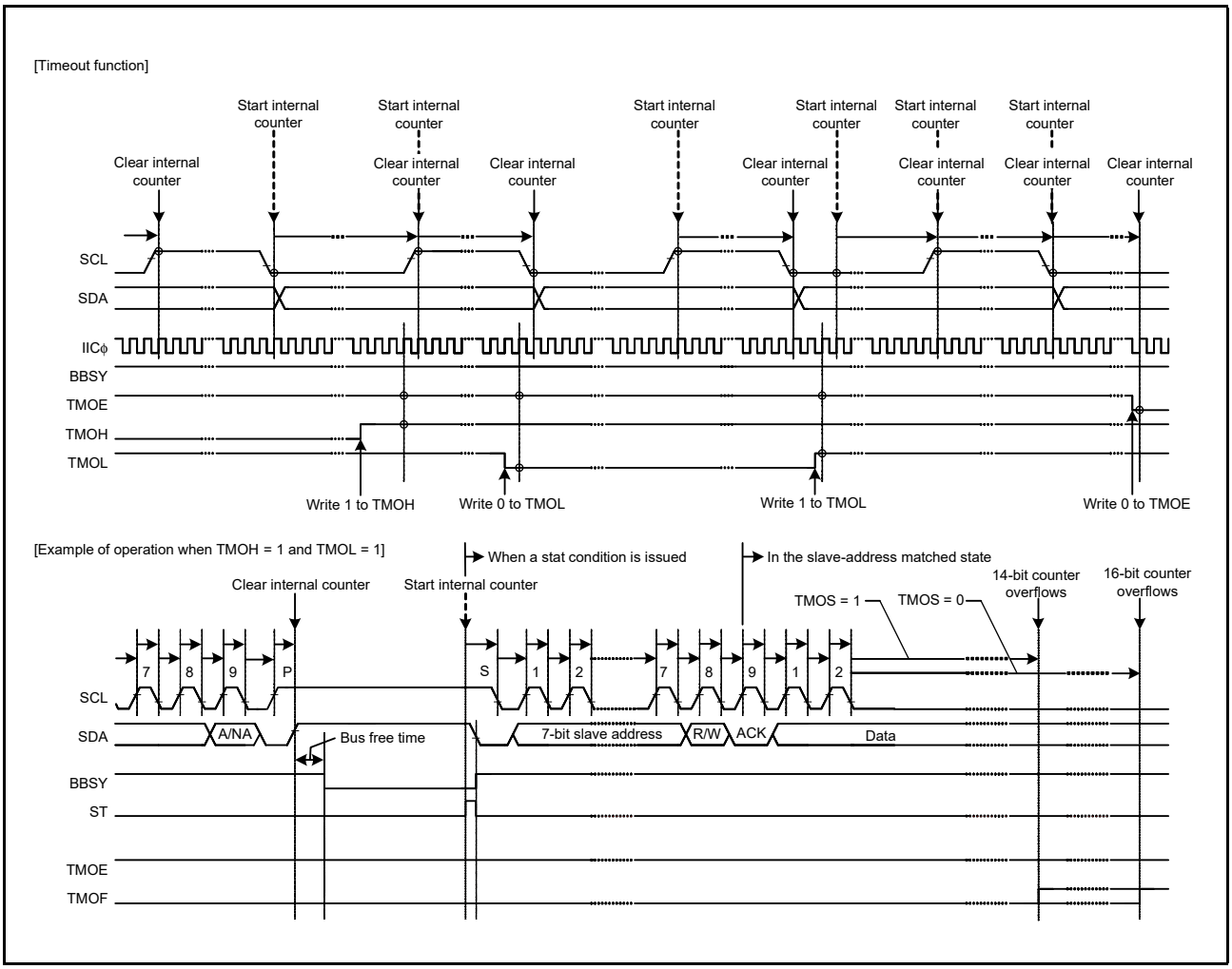


Figure 26.38 Timeout Function (TMOE, TMOS, TMOH, and TMOL Bits)

26.11.2 Extra SCL Clock Cycle Output Function

In master mode, the RIIC module has a facility for the output of extra SCL clock cycles to release the SDA line of the slave device from being held at the low level due to the master being out of synchronization with the slave device.
This function is mainly used in master mode to release the SDA line of the slave device from the state of being fixed to the low level by including extra cycles of SCL output from the RIIC with single cycles of the SCL clock as the unit in the case of a bus error where the RIIC cannot issue a stop condition because the slave device is holding the SDA line at the low level. Do not use this facility in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.
When the CLO bit in ICCR1 is set to 1 in master mode, a single cycle of the SCL clock at the frequency corresponding to the transfer rate settings (settings of the CKS[2:0] bits in ICMR1, and of the ICBRH and ICBRL registers) is output as an extra clock cycle. After output of this single cycle of the SCL clock, the CLO bit is automatically set to 0. Therefore, further extra clock cycles can be output consecutively by the software program writing 1 to the CLO bit after having read CLO = 0.
When the RIIC module is in master mode and the slave device is holding the SDA line at the low level because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a stop condition is not possible. The facility for output of an extra cycle of the SCL clock can be used to output extra cycles of SCL one by one to make the slave device release the SDA line from being held at the low level, thus recovering the bus from an unusable state. Release of the SDA line by the slave device can be monitored by reading the SDAI bit in ICCR1. After confirming release of the SDA line by the slave device, complete communications by reissuing the stop condition.
Use this facility with the MALE bit (master arbitration-lost detection disabled) in ICFER set to 0. If the MALE bit is set to 1 (master arbitration-lost detection enabled), arbitration is lost when the value of the SDAO bit in ICCR1 does not match the state of the SDA line, so take care on this point.

[Output conditions for using the CLO bit in ICCR1]

- When the bus is free (BBSY flag in ICCR2 = 0) or in master mode (MST bit = 1 and BBSY flag = 1 in ICCR2)
- When the communication device does not hold the SCL line low

Figure 26.39 shows the operation timing of the extra SCL clock cycle output function (CLO bit).

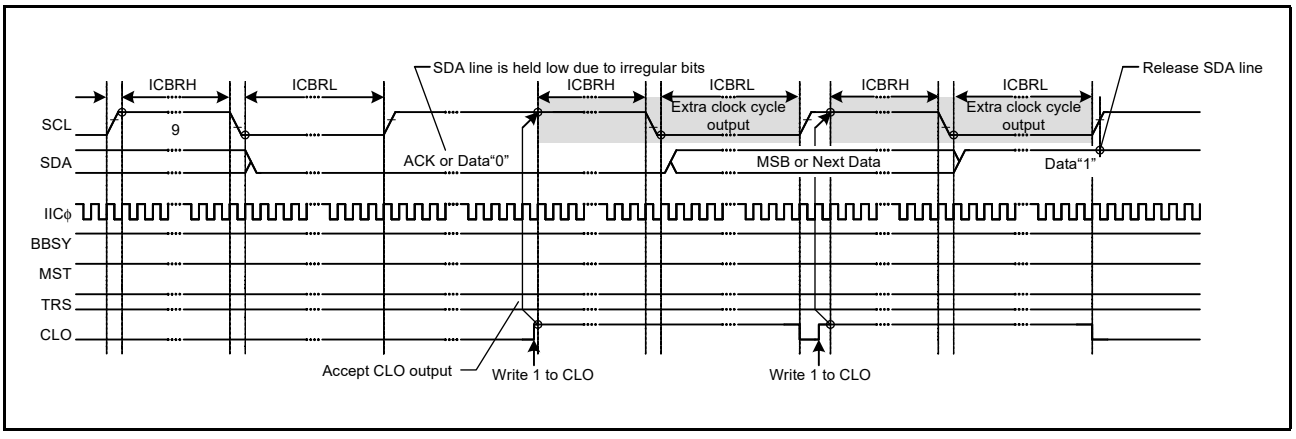


Figure 26.39 Extra SCL Clock Cycle Output Function (CLO Bit)

26.11.3 RIIC Reset and Internal Reset

The RIIC module incorporates a function for resetting itself. There are two types of reset. One is referred to as an RIIC reset; this initializes all registers including the BBSY flag in ICCR2. The other is referred to as an internal reset; this releases the RIIC from the slave-address matched state and initializes the internal counter while retaining other settings. After issuing a reset, be sure to set the IICRST bit in ICCR1 to 0.

Both types of reset are effective for release from bus-hung states since both restore the output state of the SCL and SDA pins to the high impedance state.

Do not issue a reset during slave operation because it may lead to a loss of synchronization between the master device clock and the slave device clock. Note that monitoring of the bus state, such as for the presence of a start condition, is not possible during an RIIC reset (ICE and IICRST bits = 01b in ICCR1).

For a detailed description of the RIIC and internal resets, refer to section 26.13, Resets and Register and Function States When Issuing Each Condition.

26.12 Interrupt Sources

The RIIC issues four types of interrupt request: transfer error or event generation (arbitration-lost, NACK detection, timeout detection, start condition detection, and stop condition detection), receive data full, transmit data empty, and transmit end.

Table 26.6 lists details of the several interrupt requests. The receive data full and transmit data empty are both capable of activating data transfer by the DMAC.

Table 26.6 Interrupt Sources

Symbol	Interrupt Source	Interrupt Flag	DMAC Activation	Interrupt Condition
EEI	Transfer error/ event generation	AL	Not possible	AL = 1 • ALIE = 1
		NACKF		NACKF = 1 • NAKIE = 1
		TMOF		TMOF = 1 • TMOIE = 1
		START		START = 1 • STIE = 1
		STOP		STOP = 1 • SPIE = 1
RXI*1	Receive data full	—	Possible	RDRF = 1 • RIE = 1
TXI*2	Transmit data empty	—	Possible	TDRE = 1 • TIE = 1
TEI*3	Transmit end	TEND	Not possible	TEND = 1 • TEIE = 1

Note: There is a delay time between the execution of a write instruction for a peripheral module by the CPU and actual writing to the module. Thus, when an interrupt flag has been cleared or masked, read the relevant flag again to check whether clearing or masking has been completed, and then return from interrupt handling. Returning from interrupt handling without checking that writing to the module has been completed creates a possibility of repeated processing of the same interrupt.

Note 1. The RDRF flag in ICSR2 (a condition for RXI) is automatically set to 0 when data are read from ICDRR.

Note 2. The TDRE flag in ICSR2 (a condition for TXI) is automatically set to 0 when data for transmission are written to ICDRT or a stop condition is detected (STOP flag = 1 in ICSR2).

Note 3. When using the TEI interrupt, clear the TEND flag in ICSR2 in the TEI interrupt handling.
Note that the TEND flag in ICSR2 is automatically set to 0 when data for transmission are written to ICDRT or a stop condition is detected (STOP flag = 1 in ICSR2).

Clear or mask the each flag during interrupt handling.

26.13 Resets and Register and Function States When Issuing Each Condition

The RIIC has reset, RIIC reset, and internal reset functions. Table 26.7 lists the resets and register and function states when issuing each condition.

Table 26.7 Resets and Register and Function States When Issuing Each Condition

		Chip Reset	RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/ Restart Condition Detection	Stop Condition Detection
ICCR1	ICE, IICRST	At a reset	Retained	Retained	Retained	Retained
	SCLO, SDAO		At a reset	At a reset		
	Others			Retained		
ICCR2	BBSY	At a reset	At a reset	Retained	Retained	Retained
	ST			At a reset	At a reset	Retained
	Others					At a reset
ICMR1	BC[2:0]	At a reset	At a reset	At a reset	At a reset	Retained
	Others			Retained	Retained	
ICMR2		At a reset	At a reset	Retained	Retained	Retained
ICMR3		At a reset	At a reset	Retained	Retained	Retained
ICFER		At a reset	At a reset	Retained	Retained	Retained
ICSER		At a reset	At a reset	Retained	Retained	Retained
ICIER		At a reset	At a reset	Retained	Retained	Retained
ICSR1		At a reset	At a reset	At a reset	Retained	At a reset
ICSR2	TDRE, TEND	At a reset	At a reset	At a reset	Retained	At a reset
	START				Retained	
	STOP				Retained	Retained
	Others				Retained	Retained
ICSARL0, ICSARL1, ICSARL2 ICSARU0, ICSARU1, ICSARU2		At a reset	At a reset	Retained	Retained	Retained
ICBRH, ICBRL		At a reset	At a reset	Retained	Retained	Retained
ICDRT		At a reset	At a reset	Retained	Retained	Retained
ICDRR		At a reset	At a reset	Retained	Retained	Retained
ICDRS		At a reset	At a reset	At a reset	Retained	Retained
Timeout function		At a reset	At a reset	Operation	Operation	Operation
Bus free time measurement		At a reset	At a reset	Operation	Operation	Operation

26.14 Usage Notes

26.14.1 Setting Module Stop Function

Module-stop state can be entered or canceled using module stop control register B (MSTPCRB). The initial setting is for operation of the RIIC to be stopped. RIIC register access is enabled by clearing the module-stop state.
For details on module stop control registers B, refer to section 9, Low-Power Consumption Function.

27. CAN Interface (RSCAN)

This section contains a generic description of the CAN Interface (RSCAN).

27.1 Overview

27.1.1 Functional Overview

This LSI incorporates one unit of the CAN interface (RSCAN) which has one channel (CAN1) of the CAN controller conforming to the ISO11898-1 specifications. Table 27.1 shows the specifications of the RSCAN and Figure 27.1 shows a block diagram of the RSCAN.

Table 27.1 Specifications of the RSCAN (1 / 2)

Item	Specification
Number of channels	1
Protocol	ISO11898-1 compliant (standard or extended frame)
Communication speed	• Up to 1 Mbps $\text{Communication speed (CAN1 bit time clock)} = \frac{1}{\text{CAN1 bit time}}$ CAN1 bit time = CAN1Tq × Tq count per bit $\text{CAN1Tq} = \frac{\text{(BRP[9:0] bits in the RSCAN0C1CFG register + 1)}}{\text{fCAN}}$ Tq: Time quantum (1 Tq per bit = 1 + TSEG1 + TSEG2) fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCAN0GCFG register)
Buffer	80 buffers in total • Transmit buffer: 16 buffers (shared with the transmit queue) Transmit queue: Single queue (shared with transmit buffers; up to 16 buffers allocatable) • Shared buffer: 64 buffers Receive buffer: 16 buffers (shared with receive buffers and transmit/receive buffers; up to 64 buffers allocatable to each) Receive FIFO buffer: 8 FIFO buffers (shared with receive buffers and transmit/receive buffers; up to 64 buffers allocatable to each) Transmit/receive FIFO buffer: 3 FIFO buffers (shared with receive buffers and receive buffers; up to 64 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 64 reception rules. • Sets the number of reception rules (0 to 64). • Acceptance filter processing: Sets ID and mask for each reception rule. • DLC filter processing: Enables DLC filter check for each acceptance rule.
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. • Transmission 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.

Table 27.1 Specifications of the RSCAN (2 / 2)

Item	Specification
Transmission history function	Stores the history information of transmission-completed messages
Gateway function	Transmits a received message automatically.
Bus off recovery mode selection	• Selects the method for returning from bus off state. • ISO11898-1 compliant • Automatic entry to channel halt mode at bus-off entry • Automatic entry to channel halt mode at bus-off end • Transition to channel standby 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	5 sources • Global Interrupts (2 sources) CAN receive FIFO interrupt CAN global error interrupt • Channel interrupts (3 sources) CAN1 transmit interrupt (CAN1 transmission complete, CAN1 transmission abort, etc.) CAN1 transmit/receive FIFO receive complete interrupt CAN1 error interrupt (bus error, bus lock, etc.)
Error source	An ECC error in the buffer RAM is detected and notified to the error control module (ECM). • 1-bit ECC error in the RSCAN RAM. • 2-bit ECC error in the RSCAN RAM. • RSCAN overflow error
Low-power consumption function	The module-stop state can be set.
CAN clock source	PLL0 frequency-division of CANCLKA (24 MHz) or the main clock signal (CANCLKB; 25 MHz) is selectable
Test function	Test function for user evaluation • Listen-only mode • Self-test mode 0 (external loopback) • Self-test mode 1 (internal loopback) • RAM test (read/write test)

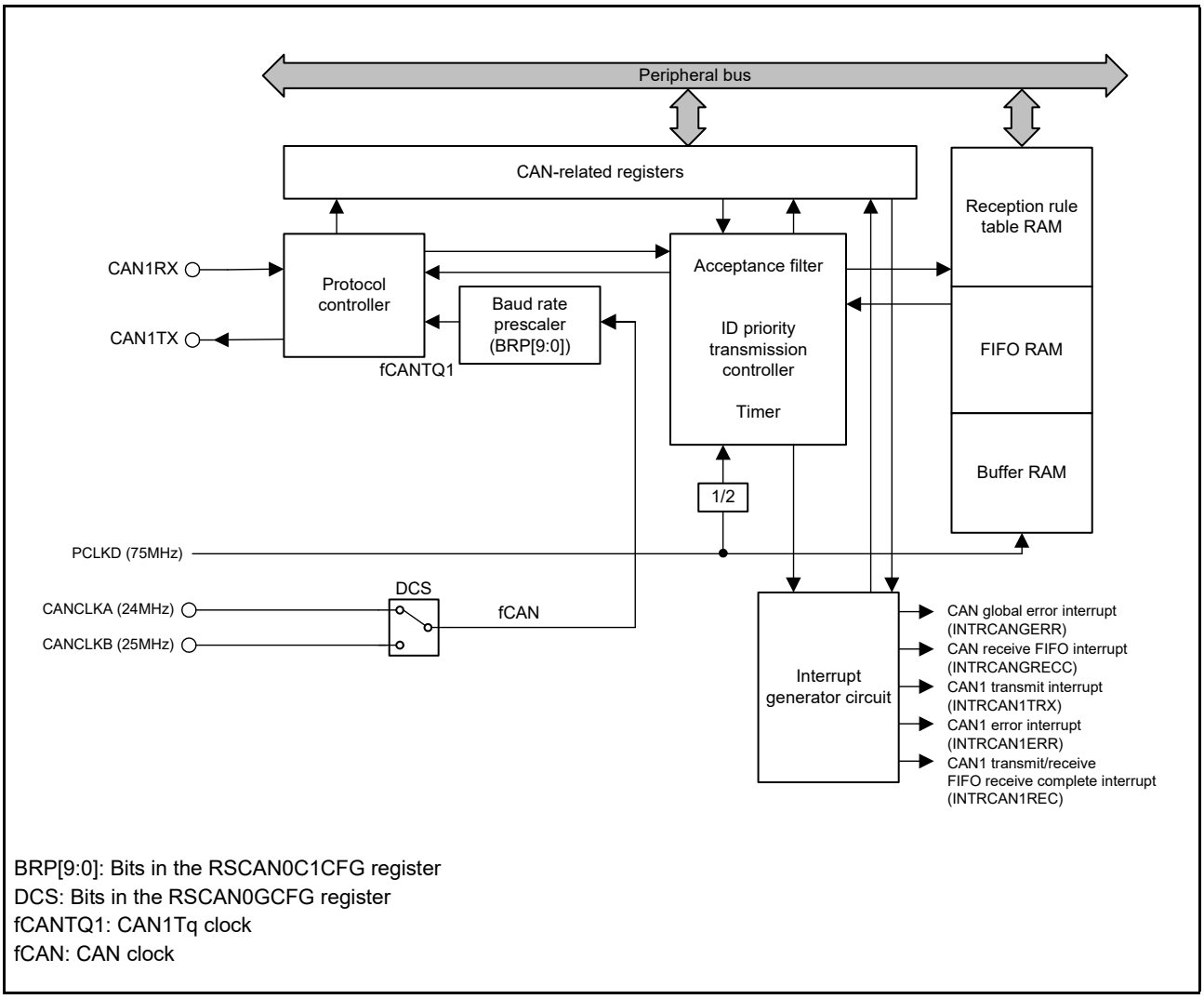


Figure 27.1 Block Diagram of the RSCAN

Table 27.2 lists the I/O pins used in the RSCAN.

Table 27.2 RSCAN Pin Configuration

Channel	Pin Name	I/O	Function
CAN1	CRXD1	Input	CAN1 receive data input pin
	CTXD1	Output	CAN1 transmit data output pin

Table 27.3 lists the meanings of indices used throughout this section.

Table 27.3 Index

Index	Meaning
j	The individual registers associated with reception rule table are generically indicated by the index “j” (j = 0 to 15); for example, RSCAN0GAFLIDj is the reception rule ID register.
k	The individual transmit/receive FIFO buffers are generically indicated by the index “k” (k = 3 to 5); for example, RSCAN0CFCCk is the transmit/receive FIFO buffer configuration/control register.
x	The individual receive FIFO buffers in the RSCAN units are identified by the index “x” (x = 0 to 7): for example, RSCAN0RFSTSx is the receive FIFO buffer status register in the RSCAN0 unit.
q	The individual receive buffers are generically indicated by the index “q” (q = 16 to 31); for example, RSCAN0RMIDq is the receive buffer ID register.
p	The individual transmit buffers are generically indicated by the index “p” (p = 16 to 31); for example, RSCAN0TMCp is the transmit buffer control register.
r	The individual RAM tests for CAN are generically indicated by the index “r” (r = 0 to 63); for example, RSCAN0RPGACCr is the RAM test page access register.
z	The individual ECC error address registers for CAN are generically indicated by the index “z” (z = 0 to 7).

27.2 Register Descriptions

27.2.1 Channel Configuration Register (RSCAN0C1CFG)

The RSCAN0C1CFG register control the settings for clock timing on channel 1 (CAN1).
Modify the RSCAN0C1CFG register in channel reset mode or channel halt mode. Set these registers in channel reset mode before shifting to channel communication mode or channel wait mode. For a description of the bit timing parameters and settings, see section 27.9.1, Initial Settings.

Address(es): RSCAN.RSCAN0C1CFG A007 8010h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	BRP[9:0]									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b9 to b0	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.	R/W
b15 to b10	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b19 to b16	TSEG1[3:0]	Time Segment 1 Control	b19 b18 b17 b16 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 Settings other than above are prohibited.	R/W
b22 to b20	TSEG2[2:0]	Time Segment 2 Control	b22 b21 b20 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 Settings other than above are prohibited.	R/W
b23	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b25, b24	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	R/W
b31 to b26	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

BRP[9:0] Bits (Prescaler Division Ratio Set)

The CAN1Tq clock (fCANTQ1) is calculated by dividing the CAN clock (fCAN) by the baud rate prescaler, ((BRP[9:0]) + 1). One clock cycle of the CAN1Tq clock is 1 Time Quantum (Tq).

TSEG1[3:0] Bits (Time Segment 1 Control)

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.

TSEG2[2:0] Bits (Time Segment 2 Control)

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.

SJW[1:0] Bits (Resynchronization Jump Width Control)

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. For the Tq value, see the description of BRP[9:0] bits.

27.2.2 Channel Control Registers (RSCAN0C1CTR)

The RSCAN0C1CTR registers control the operating mode of and the settings for interrupts from channel 1 (CAN1).

Address(es): RSCAN.RSCAN0C1CTR A007 8014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CHMDC[1:0]	Mode Select	b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode Settings other than above are prohibited.	R/W
b2	CSLPR	Channel Stop Mode	0: Other than channel stop mode 1: Channel stop mode	R/W
b3	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.	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	BEIE	Bus Error Interrupt Enable	0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.	R/W
b9	EWIE	Error Warning Interrupt Enable	0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.	R/W
b10	EPIE	Error Passive Interrupt Enable	0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.	R/W
b11	BOEIE	Bus Off Entry Interrupt Enable	0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.	R/W
b12	BORIE	Bus Off Recovery Interrupt Enable	0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.	R/W
b13	OLIE	Overload Frame Transmit Interrupt Enable	0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.	R/W
b14	BLIE	Bus Lock Interrupt Enable	0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.	R/W
b15	ALIE	Arbitration Lost Interrupt Enable	0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.	R/W
b16	TAIE	Transmission Abort Interrupt Enable	0: Transmission abort interrupt is disabled. 1: Transmission abort interrupt is enabled.	R/W
b20 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b22, b21	BOM[1:0]	Bus Off Recovery Mode Select	b22 b21 0 0: ISO11898-1 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	R/W
b23	ERRD	Error Display Mode Select	0: Error flags are displayed only for the first error information after bits 14 to 8 in RSCAN0C1ERFL are all cleared. 1: Error flags for all error information are displayed.	R/W

Bit	Symbol	Bit Name	Description	R/W
b24	CTME	Communication Test Mode Enable	0: Communication test mode is disabled. 1: Communication test mode is enabled.	R/W
b26, b25	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)	R/W
b31 to b27	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

CHMDC[1:0] Bits (Mode Select)

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see section 27.4.2, 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 11b. When the CAN 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 10b.

CSLPR Bit (Channel Stop Mode)

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 wait mode.

RTBO Bit (Forcible Return from Bus-off)

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 RSCAN0C1STS register to 00h and also clears the BOSTS flag in the RSCAN0C1STS 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 RSCAN0C1CTR register are 00b (ISO11898-1 compliant).
A delay of up to one CAN1 bit time occurs after the RTBO bit is set to 1 until the RSCAN module transitions to the error active state. Set this bit to 1 in channel communication mode.

BEIE Bit (Bus Error Interrupt Enable)

When the BEF flag in the RSCAN0C1ERFL register is set to 1 with the BEIE bit set to 1, a CAN1 error interrupt request (bus error interrupt) is generated. Modify this bit in channel reset mode.

EWIE Bit (Error Warning Interrupt Enable)

When the EWF flag in the RSCAN0C1ERFL register is set to 1 with the EWIE bit set to 1, a CAN1 error interrupt request (error warning interrupt) is generated. Modify this bit in channel reset mode.

EPIE Bit (Error Passive Interrupt Enable)

When the EPF flag in the RSCAN0C1ERFL register is set to 1 with the EPIE bit set to 1, a CAN1 error interrupt request (error passive interrupt) is generated. Modify this bit in channel reset mode.

BOEIE Bit (Bus Off Entry Interrupt Enable)

When the BOEF flag in the RSCAN0C1ERFL register is set to 1 with the BOEIE bit set to 1, a CAN1 error interrupt request (bus off entry interrupt) is generated. Modify this bit in channel reset mode.

BORIE Bit (Bus Off Recovery Interrupt Enable)

When the BORF flag in the RSCAN0C1ERFL register is set to 1 with the BORIE bit set to 1, a CAN1 error interrupt request (bus off recovery interrupt) is generated. Modify this bit in channel reset mode.

OLIE Bit (Overload Frame Transmit Interrupt Enable)

When the OVLF flag in the RSCAN0C1ERFL register is set to 1 with the OLIE bit set to 1, a CAN1 error interrupt request (overload frame interrupt) is generated. Modify this bit in channel reset mode.

BLIE Bit (Bus Lock Interrupt Enable)

When the BLF flag in the RSCAN0C1ERFL register is set to 1 with the BLIE bit set to 1, a CAN1 error interrupt request (bus lock interrupt) is generated. Modify this bit in channel reset mode.

ALIE Bit (Arbitration Lost Interrupt Enable)

When the ALF flag in the RSCAN0C1ERFL register is set to 1 with the ALIE bit set to 1, a CAN1 error interrupt request (arbitration lost interrupt) is generated. Modify this bit in channel reset mode.

TAIE Bit (Transmission Abort Interrupt Enable)

When transmission abort of the transmit buffer is completed with the TAIE bit set to 1, a CAN1 transmission interrupt request (transmission abort interrupt) is generated. Modify this bit only in channel reset mode.

BOM[1:0] Bits (Bus Off Recovery Mode Select)

These bits are used to select the bus off recovery mode of the RSCAN module.

When the BOM[1:0] bits are set to 00b, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RSCAN 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 10b (channel halt mode) before recessive bits are detected 128 times, the RSCAN module does not transition to channel halt mode until recessive bits are detected 128 times.

When the RSCAN module reaches the bus off state when the BOM[1:0] bits are set to 01b, the CHMDC[1:0] bits in the RSCAN0C1CTR register are set to 10b and the RSCAN 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 in the RSCAN0C1STS register are cleared to 00h.

When the RSCAN module reaches the bus off state when the BOM[1:0] bits are set to 10b, the CHMDC[1:0] bits are set to 10b and the RSCAN 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 00h.

When the BOM[1:0] bits are set to 11b and the CHMDC[1:0] bits are set to 10b while the RSCAN module is in the bus off state, the RSCAN 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 00h. However, if 11 consecutive recessive bits are detected 128 times and the RSCAN module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10b, a bus off recovery interrupt request is generated.

If a program writes to the CHMDC[1:0] bit at the same time as the RSCAN module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01b or at bus off end when the BOM[1:0] bits are 10b), the program's writing takes precedence. Modify the BOM[1:0] bits in channel reset mode.

ERRD Bit (Error Display Mode Select)

This bit is used to control the display mode of bits 14 to 8 in the RSCAN0C1ERFL register.
When this bit is clear to 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 in channel reset mode or channel halt mode.

CTMS[1:0] Bits (Communication Test Mode Select)

These bits are used to select a communication test mode. Modify these bits in channel halt mode. These bits are set to 0 in channel reset mode.

27.2.3 Channel Status Register (RSCAN0C1STS)

The RSCAN0C1STS register is a status register that indicates the state of transfer on channel 1 (CAN1).

Address(es): RSCAN.RSCAN0C1STS A007 8018h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	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

Bit	Symbol	Bit Name	Description	R/W
b0	CRSTSTS	Channel Reset Status Flag	0: Not in channel reset mode 1: In channel reset mode	R
b1	CHLTSTS	Channel Halt Status Flag	0: Not in channel halt mode 1: In channel halt mode	R
b2	CSLPSTS	Channel Stop Status Flag	0: Not in channel stop mode 1: In channel stop mode	R
b3	EPSTS	Error Passive Status Flag	0: Not in error passive state 1: In error passive state	R
b4	BOSTS	Bus Off Status Flag	0: Not in bus off state 1: In bus off state	R
b5	TRMSTS	Transmission Status Flag	0: Bus idle or in reception 1: In transmission or bus off state	R
b6	RECSTS	Reception Status Flag	0: Bus idle, in transmission or bus off state 1: In reception	R
b7	COMSTS	Communication Status Flag	0: Communication is not ready. 1: Communication is ready.	R
b15 to b8	—	Reserved	These bits are read as 0.	R
b23 to b16	REC[7:0]	Receive Error Counter	The receive error counter (REC) can be read.	R
b31 to b24	TEC[7:0]	Transmit Error Counter	The transmit error counter (TEC) can be read.	R

CRSTSTS Flag (Channel Reset Status Flag)

This flag is set to 1 when the CAN module has transitioned to channel reset mode, and is cleared to 0 when the CAN module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the CAN module transitions from channel reset mode to channel stop mode.

CHLTSTS Flag (Channel Halt Status Flag)

This flag is set to 1 when the CAN module has transitioned to channel halt mode, and is cleared to 0 when the CAN module has returned from channel halt mode.

CSLPSTS Flag (Channel Stop Status Flag)

This flag is set to 1 when the CAN module has transitioned to channel stop mode, and is cleared to 0 when the CAN module has returned from channel stop mode.

EPSTS Flag (Error Passive Status Flag)

This flag is set to 1 when the RSCAN module has entered the error passive state ((128 ≤ TEC[7:0] ≤ 255) or (128 ≤ REC[7:0])). It is cleared to 0 when the RSCAN module has exited the error passive state or has entered channel reset mode.

BOSTS Flag (Bus Off Status 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 CAN module has exited the bus off state.

TRMSTS Flag (Transmission Status 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.

RECSTS Flag (Reception Status 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.

COMSTS Flag (Communication Status Flag)

This bit indicates that communication is ready.
This flag becomes 1 when the CAN 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.

REC[7:0] Bits (Receive Error Counter)

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1).
These bits are cleared to 0 in channel reset mode.

TEC[7:0] Bits (Transmit Error Counter)

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1).
These bits are cleared to 0 in channel reset mode.

27.2.4 Channel Error Flag Registers (RSCAN0C1ERFL)

The RSCAN0C1ERFL register is a status register that indicates the state of error detection on channel 1 (CAN1). See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. Writing 1 to these flags does not change them to 1. 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 RSCAN0C1CTR register is set to 0 (i.e., only the flags for the first error event are displayed) and an error related to bits 14 to 8 of RSCAN0C1ERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the when time the error occurred.

Address(es): RSCAN.RSCAN0C1ERFL A007 801Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	CRCREG[14:0]														
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	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

Bit	Symbol	Bit Name	Description	R/W
b0	BEF	Bus Error Flag	0: No channel bus error is detected. 1: Channel bus error is detected.	R/W *1
b1	EWF	Error Warning Flag	0: No error warning is detected. 1: Error warning is detected.	R/W *1
b2	EPF	Error Passive Flag	0: No error passive is detected. 1: Error passive is detected.	R/W *1
b3	BOEF	Bus Off Entry Flag	0: No bus off entry is detected. 1: Bus off entry is detected.	R/W *1
b4	BORF	Bus Off Recovery Flag	0: No bus off recovery is detected. 1: Bus off recovery is detected.	R/W *1
b5	OVLF	Overload Flag	0: No overload is detected. 1: Overload is detected.	R/W *1
b6	BLF	Bus Lock Flag	0: No channel bus lock is detected. 1: Channel bus lock is detected.	R/W *1
b7	ALF	Arbitration-lost Flag	0: No arbitration-lost is detected. 1: Arbitration-lost is detected.	R/W *1
b8	SERR	Stuff Error Flag	0: No stuff error is detected. 1: Stuff error is detected.	R/W *1
b9	FERR	Form Error Flag	0: No form error is detected. 1: Form error is detected.	R/W *1
b10	AERR	ACK Error Flag	0: No ACK error is detected. 1: ACK error is detected.	R/W *1
b11	CERR	CRC Error Flag	0: No CRC error is detected. 1: CRC error is detected.	R/W *1
b12	B1ERR	Recessive Bit Error Flag	0: No recessive bit error is detected. 1: Recessive bit error is detected.	R/W *1
b13	B0ERR	Dominant Bit Error Flag	0: No dominant bit error is detected. 1: Dominant bit error is detected.	R/W *1
b14	ADERR	ACK Delimiter Error Flag	0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.	R/W *1
b15	—	Reserved	This bit is read as 0.	R

Bit	Symbol	Bit Name	Description	R/W
b30 to b16	CRCREG [14:0]	CRC Calculation Data	A CRC value calculated based on the transmit message or receive message is indicated.	R
b31	—	Reserved	This bit is read as 0.	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.

BEF Flag (Bus Error Flag)

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCAN0C1ERFL register is set to 1.

EWF Flag (Error Warning 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.

EPF Flag (Error Passive 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.

BOEF Flag (Bus Off Entry 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 RSCAN0C1CTR register are set to 01b (transition to channel halt mode at bus off entry).

BORF Flag (Bus Off Recovery Flag)

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the CAN module returns from the bus off state. However, this flag is not set to 1 if the CAN 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 RSCAN0C1CTR register are set to 01b (channel reset mode).
- The RTBO bit in the RSCAN0C1CTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCAN0C1CTR register are set to 01b (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCAN0C1CTR register are set to 10b (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11b (transition to channel halt mode upon a request from the program during bus off).

OVLF Flag (Overload Flag)

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BLF Flag (Bus Lock 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 CAN module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0.

ALF Flag (Arbitration-lost Flag)

This flag is set to 1 when an arbitration-lost has been detected.

SERR Flag (Stuff Error Flag)

This flag is set to 1 when a stuff error has been detected.

FERR Flag (Form Error Flag)

This flag is set to 1 when a form error has been detected.

AERR Flag (ACK Error Flag)

This flag is set to 1 when an ACK error has been detected.

CERR Flag (CRC Error Flag)

This flag is set to 1 when a CRC error has been detected.

B1ERR Flag (Recessive Bit Error Flag)

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

B0ERR Flag (Dominant Bit Error Flag)

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

ADERR Flag (ACK Delimiter Error Flag)

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

CRCREG[14:0] Flag (CRC Calculation Data)

When the CTME bit in the RSCAN0C1CTR 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.

27.2.5 Global Configuration Register (RSCAN0GCFG)

The RSCAN0GCFG register controls the settings for the clock signals in the RSCAN module as a whole, interval timers, etc.

Modify the RSCAN0GCFG register in global reset mode.

Address(es): RSCAN.RSCAN0GCFG A007 8084h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ITRCP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TSBTCS[2:0]			TSSS	TSP[3:0]				—	—	—	DCS	MME	DRE	DCE	TPRI
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TPRI	Transmit Priority Select	0: ID priority 1: Transmit buffer number priority	R/W
b1	DCE	DLC Check Enable	0: DLC check is disabled. 1: DLC check is enabled.	R/W
b2	DRE	DLC Replacement Enable	0: DLC replacement is disabled. 1: DLC replacement is enabled.	R/W
b3	MME	Mirror Function Enable	0: Mirror function is disabled. 1: Mirror function is enabled.	R/W
b4	DCS	CAN Clock Source Select	0: CANCLKA (24 MHz) 1: CANCLKB (25 MHz)	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b11 to b8	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	R/W
b12	TSSS	Timestamp Source Select	0: PCLKD (75 MHz)/2*1 1: Bit time clock (clock selected by the DCS bit)*2	R/W
b15 to b13	TSBTCS[2:0]	Timestamp Clock Source Select	b15 b14 b13 0 0 0: Value after reset 0 0 1: Channel 1 bit time clock Settings other than above are prohibited.	R/W
b31 to b16	ITRCP[15:0]	Interval Timer Prescaler Set	When these bits are set to M, the PCLKD (75 MHz) is divided by M. Setting 0000h is prohibited when the interval timer is in use.	R/W

Note 1. When specifying PCLKD/2 as the timestamp counter source, set bits TSBTCS[2:0] to 000b.
Note 2. When specifying the bit time clock as the timestamp counter source, set bits TSBTCS[2:0] to 001b.

TPRI Bit (Transmit Priority Select)

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 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.

DCE Bit (DLC Check Enable)

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 RSCAN0GAFLP0j register to 0000b before clearing the DCE bit in the RSCAN0GCFG register to 0.

DRE Bit (DLC Replacement Enable)

When the DRE bit is set to 1, the DLC value of the reception 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 00h is stored in each data byte beyond the DLC value of the reception rule.
The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

MME Bit (Mirror Function Enable)

Setting this bit to 1 makes the mirror function available.

DCS Bit (CAN Clock Source Select)

When this bit is set to 0, CANCLKA (24 MHz) is used as the clock source of the CAN clock (fCAN).
When this bit is set to 1, CANCLKB (25 MHz) is used as the clock source of the CAN clock (fCAN).

TSP[3:0] Bits (Timestamp Clock Source Division)

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 source.

TSSS Bit (Timestamp Source Select)

This bit is used to select a clock source of the timestamp counter.

TSBTCS[2:0] Bits (Timestamp Clock Source Select)

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.

ITRCP[15:0] Bits (Interval Timer Prescaler Set)

These bits are used to set a clock source division value of the interval timer for FIFO buffers. See section 27.6.3.1, Interval Transmission Function.

27.2.6 Global Control Register (RSCAN0GCTR)

The RSCAN0GCTR register controls the operating mode of the RSCAN module as a whole and global interrupts.

Address(es): RSCAN.RSCAN0GCTR A007 8088h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	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

Bit	Symbol	Bit Name	Description	R/W
b1, b0	GMDC[1:0]	Global Mode Select	b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode Settings other than above are prohibited.	R/W
b2	GSLPR	Global Stop Mode	0: Other than global stop mode 1: Global stop mode	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	DEIE	DLC Error Interrupt Enable	0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.	R/W
b9	MEIE	FIFO Message Lost Interrupt Enable	0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.	R/W
b10	THLEIE	Transmission History Buffer Overflow Interrupt Enable	0: Transmission history buffer overflow interrupt is disabled. 1: Transmission history buffer overflow interrupt is enabled.	R/W
b15 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	TSRST	Timestamp Counter Reset	Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.	R/W
b31 to b17	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GMDC[1:0] Bits (Global Mode Select)

These bits are used to select the mode of the entire RSCAN module (global operating mode, global reset mode, or global test mode). For details, see section 27.4.1, Global Modes. Setting the GSLPR bit to 1 when in global reset mode places the RSCAN module into global stop mode.

GSLPR Bit (Global Stop Mode)

Setting this bit to 1 places the RSCAN module into global stop mode.
Clearing this bit to 0 makes the RSCAN module leave from global stop mode.
This bit should not be modified in global operating mode or global test mode.

DEIE Bit (DLC Error Interrupt Enable)

When the DEIE bit is set to 1 and the DEF flag in the RSCAN0GERFL register is set to 1, a CAN global error interrupt request (DLC error interrupt) is generated. Modify this bit in global reset mode.

MEIE Bit (FIFO Message Lost Interrupt Enable)

When the MEIE bit is set to 1 and the MES flag in the RSCAN0GERFL register is set to 1, a CAN global error interrupt request (FIFO message lost interrupt) is generated. Modify this bit in global reset mode.

THLEIE Bit (Transmission History Buffer Overflow Interrupt Enable)

When the THLEIE bit is set to 1 and the THLES flag in the RSCAN0GERFL register is set to 1, a CAN global error interrupt request (transmission history buffer overflow interrupt) is generated. Modify this bit in global reset mode.

TSRST Bit (Timestamp Counter Reset)

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCAN0GTSC register is cleared to 0000h.

27.2.7 Global Status Register (RSCAN0GSTS)

The RSCAN0GSTS register is a status register that indicates the operating state of the RSCAN module as a whole.

Address(es): RSCAN.RSCAN0GSTS A007 808Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAMINIT	GSLPSTS	GHLTSTS	GRSTSTS
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	GRSTSTS	Global Reset Status Flag	0: Not in global reset mode 1: In global reset mode	R
b1	GHLTSTS	Global Test Status Flag	0: Not in global test mode 1: In global test mode	R
b2	GSLPSTS	Global Stop Status Flag	0: Not in global stop mode 1: In global stop mode	R
b3	GRAMINIT	CAN RAM Initialization Status Flag	0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.	R
b31 to b4	—	Reserved	These bits are read as 0.	R

GRSTSTS Flag (Global Reset Status Flag)

This flag is set to 1 when the CAN module has transitioned to global reset mode, and is cleared to 0 when the CAN module has exited global reset mode. This flag remains 1 even when the CAN module has transitioned from global reset mode to global stop mode.

GHLTSTS Flag (Global Test Status Flag)

This flag is set to 1 when the CAN module has transitioned to global test mode, and is cleared to 0 when the CAN module has exited global test mode.

GSLPSTS Flag (Global Stop Status Flag)

This flag is set to 1 when the CAN module has transitioned to global stop mode, and is cleared to 0 when the CAN module has returned from global stop mode.

GRAMINIT Flag (CAN RAM Initialization Status 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.

27.2.8 Global Error Flag Register (RSCAN0GERFL)

The RSCAN0GERFL register is a status register that indicates the error state of the RSCAN module as a whole.
All flags in the RSCAN0GERFL register are cleared to 0 in global reset mode.

Address(es): RSCAN.RSCAN0GERFL A007 8090h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	THLES	MES	DEF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DEF	DLC Error Flag	0: No DLC error has occurred. 1: A DLC error has occurred.	R/W *1
b1	MES	FIFO Message Lost Status Flag	0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.	R
b2	THLES	Transmission History Buffer Overflow Status Flag	0: No transmission history buffer overflow has occurred. 1: A transmission history buffer overflow has occurred.	R
b31 to b3	—	Reserved	These bits are read as 0.	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.

DEF Flag (DLC Error Flag)

The DEF flag is set to 1 when an error has been detected during the DLC check. This flag can be cleared by writing 0 to it.

MES Flag (FIFO Message Lost Status Flag)

The MES flag is set to 1 when any one of the RFMLT flags in the RSCAN0RFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCAN0CFSTSk register (k = 3 to 5) is set to 1.
This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

THLES Flag (Transmission History Buffer Overflow Status Flag)

The THLES flag is set to 1 when the THLELT flag in the RSCAN0THLSTS1 register is set to 1.
This flag is cleared to 0 when the THLELT flag is set to 0.

27.2.9 Global TX Interrupt Status Register 0 (RSCAN0GTINTSTS0)

The RSCAN0GTINTSTS0 register is a status register that indicates the state of interrupts in transmission by channel 1 (CAN1) of the RSCAN module.

Address(es): RSCAN.RSCAN0GTINTSTS0 A007 8460h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0.	R
b8	TSIF1	Channel 1 Transmit Buffer Interrupt Status Flag	0: Transmit buffer transmission complete interrupt is not requested. 1: Transmit buffer transmission complete interrupt is requested.	R*1
b9	TAIF1	Channel 1 Transmit Buffer Transmission Abort Interrupt Status Flag	0: Transmit buffer transmission abort interrupt is not requested. 1: Transmit buffer transmission abort interrupt is requested.	R*1
b10	TQIF1	Channel 1 Transmit Queue Interrupt Status Flag	0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.	R*1
b11	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.	R*1
b12	THIF1	Channel 1 Transmission History Interrupt Status Flag	0: Transmission history interrupt is not requested. 1: Transmission history interrupt is requested.	R*1
b31 to b13	—	Reserved	These bits are read as 0.	R

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

TSIF1 Bit (Channel 1 Transmit Buffer Interrupt Status Flag)

The TSIF1 bit is set to 1 when the TMIE bit in the RSCAN0TMIEC0 register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flag in the RCAN0TMSTSp (p = 16 to 31) register is set to 10b (transmission completed without abort request) or 11b (transmission completed with abort request).

When the TMTRF[1:0] flag is cleared to 00b under the condition that the TSIF1 bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIE bit to 0 also clears this flag to 0.

TAIF1 Bit (Channel 1 Transmit Buffer Transmission Abort Interrupt Status Flag)

The TAIF1 bit is set to 1 when the TAIE bit in the RSCAN0C1CTR register is 1 (transmission abort interrupt enabled) and the TMTRF[1:0] flag in the RCAN0TMSTSp (p = 16 to 31) register is set to 01b (transmission abort completed). This flag is cleared to 0 when the TMTRF[1:0] flag is cleared to 00b after the transmission abort is completed.

TQIF1 Bit (Channel 1 Transmit Queue Interrupt Status Flag)

When the TXQIE bit in the RSCAN0TXQCC1 register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCAN0TXQSTS1 register is set to 1 (transmit queue interrupt request), the TQIF1 bit is set to 1.
When the TXQIF bit (transmit queue interrupt request) in the RSCAN0TXQSTS1 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.

CFTIF1 Bit (Channel 1 Transmit/Receive FIFO Transmit Interrupt Status Flag)

When the CFTXIE bit in the RSCAN0CFCCk (k = 3 to 5) register is set to 1 (transmit/receive FIFO transmit interrupt enabled) and the CFTXIF bit in the RSCAN0CFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIF1 bit is set to 1.
When the CFTXIF bit is cleared to 0 under the conditions that the CFTIF1 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.

THIF1 Bit (Channel 1 Transmission History Interrupt Status Flag)

When the THLIE bit in the RSCAN0THLCC1 register is set to 1 (transmission history interrupt enabled) and the THLIF bit in the RSCAN0THLSTS1 register is set to 1 (transmission history interrupt request), the THIF1 bit is set to 1.
When the THLIF bit in the RSCAN0THLSTS1 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.

27.2.10 Global Timestamp Counter Register (RSCAN0GTSC)

The RSCAN0GTSC register is a 32-bit counter that indicates the timestamp counter value.

Address(es): RSCAN.RSCAN0GTSC A007 8094h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	TS[15:0]	Timestamp Value	The timestamp counter value can be read. Counter Value: 0000h to FFFFh	R
b31 to b16	—	Reserved	These bits are read as 0.	R

TS[15:0] Bits (Timestamp Value)

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. 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 RSCAN0GCFG register is 0 (PCLKD/2):
  The timestamp counter starts counting when the RSCAN module has transitioned to global operating mode.
  This counter stops counting when the RSCAN module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CAN1 bit time clock):
  The timestamp counter starts counting when channel 1 (CAN1) has transitioned to channel communication mode.
  This counter stops counting when channel 1 (CAN1) has transitioned to channel reset mode or channel halt mode.

27.2.11 Reception Rule Entry Control Register (RSCAN0GAFLECTR)

The RSCAN0GAFLECTR register controls the reception rule table.

Address(es): RSCAN.RSCAN0GAFLECTR A007 8098h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	AFLDAE	—	—	—	AFLPN[4:0]				
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b4 to b0	AFLPN[4:0]	Reception Rule Table Page Number Configuration	A page number can be selected from a range of page 0 (00000b) to page 3 (00011b).	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	AFLDAE	Reception Rule Table Write Enable	0: Writing to the reception rule table is disabled. 1: Writing to the reception rule table is enabled.	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

AFLPN[4:0] Bits (Reception Rule Table Write Enable)

These bits are used to set the page number of the reception rule table. Sixteen reception rules can be set per page.
Set these bits to a value within the range of 00000b to 00011b.

AFLDAE Bit (Reception Rule Table Page Number Configuration)

Setting this bit to 0 disables writing to the reception rule table. After writes to the reception rule table are completed, set this bit to 0 to disable writing to the table. The reception rule table can be read regardless of the value of this bit.
Set the AFLDAE bit to 1 only in global reset mode.

27.2.12 Reception Rule Configuration Register 0 (RSCAN0GAFLCFG0)

The RSCAN0GAFLCFG0 register controls the settings for the number of reception rules.
Modify the RSCAN0GAFLCFG0 register in global reset mode.
Up to 64 rules can be registered in the reception rule table as the entire unit. The number of reception rules should meet the following conditions.

- The number of rules is up to 64.

Address(es): RSCAN.RSCAN0GAFLCFG0 A007 809Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	RNC1[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b23 to b16	RNC1[7:0]	Channel 1 Rule Number Set	Set the number of reception rules exclusively used for channel 1.	R/W
b31 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RNC1[7:0] Bits (Channel 1 Rule Number Set)

These bits are used to set the number of rules to be registered in the channel 1 reception rule table.
Set these bits to a value within the range of 00h to 40h.

27.2.13 Reception Rule ID Register (RSCAN0GAFLIDj) (j = 0 to 15)

The RSCAN0GAFLIDj registers control the ID and frame formats of the reception rule.
Modify the RSCAN0GAFLIDj registers when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (enabling writing to the reception rule table) in global reset mode.

Address(es): RSCAN.RSCAN0GAFLID0 A007 8500h, RSCAN.RSCAN0GAFLID1 A007 8510h, RSCAN.RSCAN0GAFLID2 A007 8520h, RSCAN.RSCAN0GAFLID3 A007 8530h, RSCAN.RSCAN0GAFLID4 A007 8540h, RSCAN.RSCAN0GAFLID5 A007 8550h, RSCAN.RSCAN0GAFLID6 A007 8560h, RSCAN.RSCAN0GAFLID7 A007 8570h, RSCAN.RSCAN0GAFLID8 A007 8580h, RSCAN.RSCAN0GAFLID9 A007 8590h, RSCAN.RSCAN0GAFLID10 A007 85A0h, RSCAN.RSCAN0GAFLID11 A007 85B0h, RSCAN.RSCAN0GAFLID12 A007 85C0h, RSCAN.RSCAN0GAFLID13 A007 85D0h, RSCAN.RSCAN0GAFLID14 A007 85E0h, RSCAN.RSCAN0GAFLID15 A007 85F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:0]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	GAFLID[28:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b28 to b0	GAFLID[28:0]	ID Configuration	Set the ID of the reception rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.	R/W
b29	GAFLLB	Reception Rule Target Message Select	0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received	R/W
b30	GAFLRTR	RTR Select	0: Data frame 1: Remote frame	R/W
b31	GAFLIDE	IDE Select	0: Standard ID 1: Extended ID	R/W

GAFLID[28:0] Bits (ID Configuration)

These bits are used to set the ID field of the reception rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

GAFLLB Bit (Reception Rule Target Message Select)

When this bit is set to 0, data processing using the reception 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 reception rule is performed when the CAN node is receiving its own transmitted messages.

GAFLRTR Bit (RTR Select)

This bit is used to select the frame format (data frame or remote frame) of the reception rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLIDE Bit (IDE Select)

This bit is used to select the ID format (standard ID or extended ID) of the reception rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

27.2.14 Reception Rule Mask Registers (RSCAN0GAFLMj) (j = 0 to 15)

The RSCAN0GAFLMj registers control the settings for masking of the reception rule.
Modify the RSCAN0GAFLMj registers when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (enabling writing to the reception rule table) in global reset mode.

Address(es): RSCAN.RSCAN0GAFLM0 A007 8504h, RSCAN.RSCAN0GAFLM1 A007 8514h, RSCAN.RSCAN0GAFLM2 A007 8524h, RSCAN.RSCAN0GAFLM3 A007 8534h, RSCAN.RSCAN0GAFLM4 A007 8544h, RSCAN.RSCAN0GAFLM5 A007 8554h, RSCAN.RSCAN0GAFLM6 A007 8564h, RSCAN.RSCAN0GAFLM7 A007 8574h, RSCAN.RSCAN0GAFLM8 A007 8584h, RSCAN.RSCAN0GAFLM9 A007 8594h, RSCAN.RSCAN0GAFLM10 A007 85A4h, RSCAN.RSCAN0GAFLM11 A007 85B4h, RSCAN.RSCAN0GAFLM12 A007 85C4h, RSCAN.RSCAN0GAFLM13 A007 85D4h, RSCAN.RSCAN0GAFLM14 A007 85E4h, RSCAN.RSCAN0GAFLM15 A007 85F4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	GAFLIDEM	GAFLRTRM	—	GAFLIDM[28:0]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	GAFLIDM[28:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b28 to b0	GAFLIDM [28:0]	ID Mask	0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.	R/W
b29	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b30	GAFLRTRM	RTR Mask	0: The RTR bit is not compared. 1: The RTR bit is compared	R/W
b31	GAFLIDEM	IDE Mask	0: The IDE bit is not compared. 1: The IDE bit is compared.	R/W

GAFLIDM[28:0] Bits (ID Mask)

These bits are used to mask the corresponding ID bit of the reception rule.

GAFLRTRM Bit (RTR Mask)

This bit is used to mask the RTR bit of the reception rule.

GAFLIDEM Bit (IDE Mask)

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 RSCAN0GAFLIDj 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.

27.2.15 Reception Rule Pointer 0 Registers (RSCAN0GAFLP0j) (j = 0 to 15)

The RSCAN0GAFLP0j registers control the minimum data length and labels for received messages and the settings for the receive buffer.

Modify the RSCAN0GAFLP0j registers when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (enabling writing to the reception rule table) in global reset mode.

Address(es): RSCAN.RSCAN0GAFLP00 A007 8508h, RSCAN.RSCAN0GAFLP01 A007 8518h, RSCAN.RSCAN0GAFLP02 A007 8528h, RSCAN.RSCAN0GAFLP03 A007 8538h, RSCAN.RSCAN0GAFLP04 A007 8548h, RSCAN.RSCAN0GAFLP05 A007 8558h, RSCAN.RSCAN0GAFLP06 A007 8568h, RSCAN.RSCAN0GAFLP07 A007 8578h, RSCAN.RSCAN0GAFLP08 A007 8588h, RSCAN.RSCAN0GAFLP09 A007 8598h, RSCAN.RSCAN0GAFLP010 A007 85A8h, RSCAN.RSCAN0GAFLP011 A007 85B8h, RSCAN.RSCAN0GAFLP012 A007 85C8h, RSCAN.RSCAN0GAFLP013 A007 85D8h, RSCAN.RSCAN0GAFLP014 A007 85E8h, RSCAN.RSCAN0GAFLP015 A007 85F8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	GAFLRMV	GAFLRMDP[6:0]							—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b14 to b8	GAFLRMDP[6:0]	Receive Buffer Number Select	Set the receive buffer number to store received messages.	R/W
b15	GAFLRMV	Receive Buffer Enable	0: No receive buffer is used. 1: A receive buffer is used.	R/W
b27 to b16	GAFLPTR[11:0]	Reception Rule Label	Set the 12-bit label information.	R/W
b31 to b28	GAFLDLC[3:0]	Reception 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	R/W

GAFLRMDP[6:0] Bits (Receive Buffer Number Select)

These bits are used to select the number of the receive buffer that stores received messages that have passed through the filter when the GAFLRMV bit is set to 1.

The value (number) to be set in these bits should satisfy the following condition:

16 ≤ GAFLRMDP[6:0] < (16 + (the setting of the NRXMB[7:0] bits in the RSCAN0RMNB register)

GAFLRMV Bit (Receive Buffer Enable)

When this bit is set to 1, received messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLPTR[11:0] Bits (Reception Rule Label)

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.

GAFLDLC[3:0] Bits (Reception Rule DLC)

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 0000b disables the DLC check function allowing messages with any data length to pass the DLC check.

27.2.16 Reception Rule Pointer 1 Registers (RSCAN0GAFLP1j) (j = 0 to 15)

The RSCAN0GAFLP1j registers control the settings for the FIFO buffers for storing received messages that have passed through the filter.

Modify the RSCAN0GAFLP1j registers when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (enabling writing to the reception rule table) in global reset mode.

Address(es): RSCAN.RSCAN0GAFLP10 A007 850Ch, RSCAN.RSCAN0GAFLP11 A007 851Ch, RSCAN.RSCAN0GAFLP12 A007 852Ch, RSCAN.RSCAN0GAFLP13 A007 853Ch, RSCAN.RSCAN0GAFLP14 A007 854Ch, RSCAN.RSCAN0GAFLP15 A007 855Ch, RSCAN.RSCAN0GAFLP16 A007 856Ch, RSCAN.RSCAN0GAFLP17 A007 857Ch, RSCAN.RSCAN0GAFLP18 A007 858Ch, RSCAN.RSCAN0GAFLP19 A007 859Ch, RSCAN.RSCAN0GAFLP110 A007 85ACh, RSCAN.RSCAN0GAFLP111 A007 85BCh, RSCAN.RSCAN0GAFLP112 A007 85CCh, RSCAN.RSCAN0GAFLP113 A007 85DCh, RSCAN.RSCAN0GAFLP114 A007 85ECh, RSCAN.RSCAN0GAFLP115 A007 85FCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	GAFLFDP[17:0]									
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	GAFLFDP[17:0]								GAFLFDPr[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	GAFLFDPr [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.	R/W
b25 to b8	GAFLFDP [17:0]	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.	R/W
b31 to b26	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

GAFLFDP[17:0] and GAFLFDPr[7:0] Bits

These bits are used to specify FIFO buffers for storing received messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCAN0GAFLP0j (j = 0 to 15) register is set to 1 (storing a message 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 RSCAN0CFCCk (k = 3 to 5) register are set to 00b (receive mode) or 10b (gateway mode) are selectable.

27.2.17 Receive Buffer Number Register (RSCAN0RMNB)

The RSCAN0RMNB register controls the number of receive buffers of the RSCAN module.
Modify the RSCAN0RMNB register in global reset mode.

Address(es): RSCAN.RSCAN0RMNB A007 80A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	NRXMB[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	NRXMB[7:0]	Receive Buffer Number Configuration	Set the number of receive buffers. Set a value between 0 to 16.	R/W
b31 to b8	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

NRXMB[7:0] Bits (Receive Buffer Number Configuration)

These bits are used to set the total number of receive buffers of the RSCAN module. The maximum value is 16.
Setting these bits all to 0 makes receive buffers unavailable.

27.2.18 Receive Buffer New Data Register 0 (RSCAN0RMND0)

The RSCAN0RMND0 register is a status register that indicates completion of the reception of messages by the receive buffer.

Write 0 to the RSCAN0RMND0 register in global operating mode or global test mode.

Address(es): RSCAN.RSCAN0RMND0 A007 80A8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RMNSq_h (q = 31 to 16)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31 to b16	RMNSq_h	Receive Buffer Receive Complete Flag q _h (q = 31 to 16)	0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.	R/W

RMNSq_h Flags (q = 16 to 31)

These flags are 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 PCLKD to store a message.

These flags are cleared to 0 in global reset mode.

27.2.19 Receive Buffer ID Registers (RSCAN0RMIDq) (q = 16 to 31)

The RSCAN0RMIDq registers are status registers that indicate the state of the ID and frame formats of the messages stored in the receive buffers (q = 16 to 31).

Address(es): RSCAN.RSCAN0RMID16 A007 8700h, RSCAN.RSCAN0RMID17 A007 8710h, RSCAN.RSCAN0RMID18 A007 8720h, RSCAN.RSCAN0RMID19 A007 8730h, RSCAN.RSCAN0RMID20 A007 8740h, RSCAN.RSCAN0RMID21 A007 8750h, RSCAN.RSCAN0RMID22 A007 8760h, RSCAN.RSCAN0RMID23 A007 8770h, RSCAN.RSCAN0RMID24 A007 8780h, RSCAN.RSCAN0RMID25 A007 8790h, RSCAN.RSCAN0RMID26 A007 87A0h, RSCAN.RSCAN0RMID27 A007 87B0h, RSCAN.RSCAN0RMID28 A007 87C0h, RSCAN.RSCAN0RMID29 A007 87D0h, RSCAN.RSCAN0RMID30 A007 87E0h, RSCAN.RSCAN0RMID31 A007 87F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RMIDE	RMRTR	—	RMID[28:0]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RMID[28:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b28 to b0	RMID[28:0]	Receive Buffer ID Data	These bits contain the standard ID or extended ID of the received messages. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.	R
b29	—	Reserved	This bit is read as 0.	R
b30	RMRTR	Receive Buffer RTR	0: Data frame 1: Remote frame	R
b31	RMIDE	Receive Buffer IDE	0: Standard ID 1: Extended ID	R

RMID[28:0] Bits (Receive Buffer ID Data)

These bits contain the ID of the messages stored in the receive buffers.

RMRTR Bit (Receive Buffer RTR)

This bit indicates the frame format (data frame or remote frame) of the messages stored in the receive buffers.

RMIDE Bit (Receive Buffer IDE)

This bit indicates the ID format (standard ID or extended ID) of the messages stored in the receive buffers.

27.2.20 Receive Buffer Pointer Registers (RSCAN0RMPTRq) (q = 16 to 31)

The RSCAN0RMPTRq registers are status registers that indicate the data length, label information, and timestamp value of the messages stored in the receive buffers (q = 16 to 31).

Address(es): RSCAN.RSCAN0RMPTR16 A007 8704h, RSCAN.RSCAN0RMPTR17 A007 8714h, RSCAN.RSCAN0RMPTR18 A007 8724h, RSCAN.RSCAN0RMPTR19 A007 8734h, RSCAN.RSCAN0RMPTR20 A007 8744h, RSCAN.RSCAN0RMPTR21 A007 8754h, RSCAN.RSCAN0RMPTR22 A007 8764h, RSCAN.RSCAN0RMPTR23 A007 8774h, RSCAN.RSCAN0RMPTR24 A007 8784h, RSCAN.RSCAN0RMPTR25 A007 8794h, RSCAN.RSCAN0RMPTR26 A007 87A4h, RSCAN.RSCAN0RMPTR27 A007 87B4h, RSCAN.RSCAN0RMPTR28 A007 87C4h, RSCAN.RSCAN0RMPTR29 A007 87D4h, RSCAN.RSCAN0RMPTR30 A007 87E4h, RSCAN.RSCAN0RMPTR31 A007 87F4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RMTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	RMTS[15:0]	Receive Buffer Timestamp Data	Timestamp value of the received messages	R
b27 to b16	RMPTR[11:0]	Receive Buffer Label Data	Label information of the received messages	R
b31 to b28	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	R

RMTS[15:0] Bits (Receive Buffer Timestamp Data)

These bits indicate the timestamp value of the message stored in the receive buffer.

RMPTR[11:0] Bits (Receive Buffer Label Data)

These bits indicate the label information of the message stored in the receive buffer.

RMDLC[3:0] Bits (Receive Buffer DLC Data)

These bits indicate the data length of the message stored in the receive buffer.

27.2.21 Receive Buffer Data Field 0 Registers (RSCAN0RMDF0q) (q = 16 to 31)

The RSCAN0RMDF0q registers are data registers that hold the values in the receive buffers (q = 16 to 31).
When the value of the RMDLC[3:0] bits in the RSCAN0RMPTRq register is less than 1000b, data bytes for which no value is set are read as 00h.

Address(es): RSCAN.RSCAN0RMDF016 A007 8708h, RSCAN.RSCAN0RMDF017 A007 8718h, RSCAN.RSCAN0RMDF018 A007 8728h, RSCAN.RSCAN0RMDF019 A007 8738h, RSCAN.RSCAN0RMDF020 A007 8748h, RSCAN.RSCAN0RMDF021 A007 8758h, RSCAN.RSCAN0RMDF022 A007 8768h, RSCAN.RSCAN0RMDF023 A007 8778h, RSCAN.RSCAN0RMDF024 A007 8788h, RSCAN.RSCAN0RMDF025 A007 8798h, RSCAN.RSCAN0RMDF026 A007 87A8h, RSCAN.RSCAN0RMDF027 A007 87B8h, RSCAN.RSCAN0RMDF028 A007 87C8h, RSCAN.RSCAN0RMDF029 A007 87D8h, RSCAN.RSCAN0RMDF030 A007 87E8h, RSCAN.RSCAN0RMDF031 A007 87F8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	RMDB0[7:0]	Receive Buffer Data Byte 0	Values of messages stored in the corresponding receive FIFO buffer can be read from these bits.	R
b15 to b8	RMDB1[7:0]	Receive Buffer Data Byte 1		R
b23 to b16	RMDB2[7:0]	Receive Buffer Data Byte 2		R
b31 to b24	RMDB3[7:0]	Receive Buffer Data Byte 3		R

27.2.22 Receive Buffer Data Field 1 Registers (RSCAN0RMDF1q) (q = 16 to 31)

The RSCAN0RMDF1q registers are data registers that hold the values in the receive buffers (q = 16 to 31).
When the value of the RMDLC[3:0] bits in the RSCAN0RMPTRq register is less than 1000b, data bytes for which no value is set are read as 00h.

Address(es): RSCAN.RSCAN0RMDF116 A007 870Ch, RSCAN.RSCAN0RMDF117 A007 871Ch, RSCAN.RSCAN0RMDF118 A007 872Ch, RSCAN.RSCAN0RMDF119 A007 873Ch, RSCAN.RSCAN0RMDF120 A007 874Ch, RSCAN.RSCAN0RMDF121 A007 875Ch, RSCAN.RSCAN0RMDF122 A007 876Ch, RSCAN.RSCAN0RMDF123 A007 877Ch, RSCAN.RSCAN0RMDF124 A007 878Ch, RSCAN.RSCAN0RMDF125 A007 879Ch, RSCAN.RSCAN0RMDF126 A007 87ACh, RSCAN.RSCAN0RMDF127 A007 87BCh, RSCAN.RSCAN0RMDF128 A007 87CCh, RSCAN.RSCAN0RMDF129 A007 87DCh, RSCAN.RSCAN0RMDF130 A007 87ECh, RSCAN.RSCAN0RMDF131 A007 87FCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	RMDB4[7:0]	Receive Buffer Data Byte 4	Values of messages stored in the corresponding receive FIFO buffer can be read from these bits.	R
b15 to b8	RMDB5[7:0]	Receive Buffer Data Byte 5		R
b23 to b16	RMDB6[7:0]	Receive Buffer Data Byte 6		R
b31 to b24	RMDB7[7:0]	Receive Buffer Data Byte 7		R

27.2.23 Receive FIFO Buffer Configuration and Control Registers (RSCAN0RFCCx) (x = 0 to 7)

The RSCAN0RFCCx registers control receive FIFO buffer interrupts and the number of receive FIFO buffer stages.

Address(es): RSCAN.RSCAN0RFCC0 A007 80B8h, RSCAN.RSCAN0RFCC1 A007 80BCh, RSCAN.RSCAN0RFCC2 A007 80C0h, RSCAN.RSCAN0RFCC3 A007 80C4h, RSCAN.RSCAN0RFCC4 A007 80C8h, RSCAN.RSCAN0RFCC5 A007 80CCh, RSCAN.RSCAN0RFCC6 A007 80D0h, RSCAN.RSCAN0RFCC7 A007 80D4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b0	RFE	Receive FIFO Buffer Enable	0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.	R/W
b1	RFIE	Receive FIFO Interrupt Enable	0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.	R/W
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b10 to b8	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: Setting prohibited	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b12	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.	R/W
b15 to b13	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.	R/W
b31 to b16	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RFE Bit (Receive FIFO Buffer Enable)

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCAN0RFSTSx register to 1 (buffer empty). Modify this bit in global operating mode or global test mode.

RFIE Bit (Receive FIFO Interrupt Enable)

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).

RFDC[2:0] Bits (Receive FIFO Buffer Depth Configuration)

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 000b, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

RFIM Bit (Receive FIFO Interrupt Source Select)

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

RFIGCV[2:0] Bits (Receive FIFO Interrupt Request Timing Select)

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 001b (4 messages), set the RFIGCV[2:0] bits to 001b, 011b, 101b, or 111b. Modify these bits only in global reset mode.

27.2.24 Receive FIFO Buffer Status Registers (RSCAN0RFSTSx) (x = 0 to 7)

The RSCAN0RFSTSx registers are status registers that indicate whether the receive FIFO buffers are empty, the number of unread messages, etc.

Address(es): RSCAN.RSCAN0RFSTS0 A007 80D8h, RSCAN.RSCAN0RFSTS1 A007 80DCh, RSCAN.RSCAN0RFSTS2 A007 80E0h, RSCAN.RSCAN0RFSTS3 A007 80E4h, RSCAN.RSCAN0RFSTS4 A007 80E8h, RSCAN.RSCAN0RFSTS5 A007 80ECh, RSCAN.RSCAN0RFSTS6 A007 80F0h, RSCAN.RSCAN0RFSTS7 A007 80F4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b0	RFEMP	Receive FIFO Buffer Empty Status Flag	0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread messages (buffer empty).	R
b1	RFFLL	Receive FIFO Buffer Full Status Flag	0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.	R
b2	RFMLT	Receive FIFO Message Lost Flag	0: No receive FIFO message is lost. 1: A receive FIFO message is lost.	R/W *1
b3	RFIF	Receive FIFO Interrupt Request Flag	0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.	R/W *1
b7 to b4	—	Reserved	These bits are read as 0.	R
b15 to b8	RFMC[7:0]	Receive FIFO Unread Message Counter	The number of unread messages stored in the receive FIFO buffer is displayed.	R
b31 to b16	—	Reserved	These bits are read as 0.	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.

RFEMP Flag (Receive FIFO Buffer Empty Status 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 RSCAN0RFCCx 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.

RFFLL Flag (Receive FIFO Buffer Full Status 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 RSCAN0RFCCx 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 RSCAN0RFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFMLT Flag (Receive FIFO Message Lost 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”.

RFIF Flag (Receive FIFO Interrupt Request 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 RSCAN0RFCCx 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”.

RFMC[7:0] Flag (Receive FIFO Unread Message Counter)

This flag indicates the number of unread messages in the reception FIFO buffer. These flags become 00h when the RFE bit in the RSCAN0RFCCx register is set to 0.

27.2.25 Receive FIFO Buffer Pointer Control Registers (RSCAN0RFPCTRx) (x = 0 to 7)

The RSCAN0RFPCTRx registers control the pointer to the receive FIFO buffer.

Address(es): RSCAN.RSCAN0RFPCTR0 A007 80F8h, RSCAN.RSCAN0RFPCTR1 A007 80FCh, RSCAN.RSCAN0RFPCTR2 A007 8100h, RSCAN.RSCAN0RFPCTR3 A007 8104h, RSCAN.RSCAN0RFPCTR4 A007 8108h, RSCAN.RSCAN0RFPCTR5 A007 810Ch, RSCAN.RSCAN0RFPCTR6 A007 8110h, RSCAN.RSCAN0RFPCTR7 A007 8114h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	RFPC[7:0]	Receive FIFO Pointer Control	When these bits are set to FFh, the read pointer moves to the next unread message in the receive FIFO buffer.	W
b31 to b8	—	Reserved	The write value should be 0.	W

RFPC[7:0] Bits (Receive FIFO Pointer Control)

When the RFPC[7:0] bits are set to FFh, 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 RSCAN0RFSTSx register is decremented. Read the RSCAN0RFID, RSCAN0RFPTR, RSCAN0RFDF0, and RSCAN0RFDF1 registers to read messages in the receive FIFO buffer, and then write FFh to the RFPC[7:0] bits.

When writing FFh to these bits, make sure that the RFE bit in the RSCAN0RFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCAN0RFSTSx register is 0 (the receive FIFO buffer contains unread messages).

27.2.26 Receive FIFO Buffer Access ID Registers (RSCAN0RFIDx) (x = 0 to 7)

The RSCAN0RFIDx registers are status registers that indicate the state of the ID and frame formats of the messages stored in the receive FIFO buffers.

Address(es): RSCAN.RSCAN0RFID0 A007 8E00h, RSCAN.RSCAN0RFID1 A007 8E10h, RSCAN.RSCAN0RFID2 A007 8E20h, RSCAN.RSCAN0RFID3 A007 8E30h, RSCAN.RSCAN0RFID4 A007 8E40h, RSCAN.RSCAN0RFID5 A007 8E50h, RSCAN.RSCAN0RFID6 A007 8E60h, RSCAN.RSCAN0RFID7 A007 8E70h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	RFIDE	RFRTR	—	RFID[28:0]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RFID[28:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b28 to b0	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.	R
b29	—	Reserved	This bit is read as 0.	R
b30	RFRTR	Receive FIFO Buffer RTR	0: Data frame 1: Remote frame	R
b31	RFIDE	Receive FIFO Buffer IDE	0: Standard ID 1: Extended ID	R

RFID[28:0] Bits (Receive FIFO Buffer ID Data)

These bits indicate the ID of the messages stored in the receive FIFO buffers.

RFRTR Bit (Receive FIFO Buffer RTR)

This bit indicates the frame format (data frame or remote frame) of the messages stored in the receive FIFO buffers.

RFIDE Bit (Receive FIFO Buffer IDE)

This bit indicates the ID format (standard ID or extended ID) of the messages stored in the receive FIFO buffers.

27.2.27 Receive FIFO Buffer Access Pointer Registers (RSCAN0RFPTRx) (x = 0 to 7)

The RSCAN0RFPTRx registers are status registers that indicate the data length, label information, and timestamp value of the messages stored in the receive FIFO buffers.

Address(es): RSCAN.RSCAN0RFPTR0 A007 8E04h, RSCAN.RSCAN0RFPTR1 A007 8E14h, RSCAN.RSCAN0RFPTR2 A007 8E24h, RSCAN.RSCAN0RFPTR3 A007 8E34h, RSCAN.RSCAN0RFPTR4 A007 8E44h, RSCAN.RSCAN0RFPTR5 A007 8E54h, RSCAN.RSCAN0RFPTR6 A007 8E64h, RSCAN.RSCAN0RFPTR7 A007 8E74h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	RFTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	RFTS[15:0]	Receive FIFO Buffer Timestamp Data	Timestamp value of the received messages can be read.	R
b27 to b16	RFPTR[11:0]	Receive FIFO Buffer Label Data	Label information of the received messages can be read.	R
b31 to b28	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	R

RFTS[15:0] Bits (Receive FIFO Buffer Timestamp Data)

These bits contain the timestamp value of the messages stored in the receive FIFO buffers.

RFPTR[11:0] Bits (Receive FIFO Buffer Label Data)

These bits contain the label information of the messages stored in the receive FIFO buffers.

RFDLC[3:0] Bits (Receive FIFO Buffer DLC Data)

These bits contain the data length of the messages stored in the receive FIFO buffers.

27.2.28 Receive FIFO Buffer Access Data Field 0 Registers (RSCAN0RFDF0x) (x = 0 to 7)

The RSCAN0RFDF0x registers are data registers that hold the values in the receive buffers.
When the value of the RMDLC[3:0] bits in the RSCAN0RFPTRx register is less than 1000b, data bytes for which no value is set are read as 00h.

Address(es): RSCAN.RSCAN0RFDF00 A007 8E08h, RSCAN.RSCAN0RFDF01 A007 8E18h, RSCAN.RSCAN0RFDF02 A007 8E28h, RSCAN.RSCAN0RFDF03 A007 8E38h, RSCAN.RSCAN0RFDF04 A007 8E48h, RSCAN.RSCAN0RFDF05 A007 8E58h, RSCAN.RSCAN0RFDF06 A007 8E68h, RSCAN.RSCAN0RFDF07 A007 8E78h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	RFDB0[7:0]	Receive FIFO Buffer Data Byte 0	Values of messages stored in the corresponding receive FIFO buffer can be read from these bits.	R
b15 to b8	RFDB1[7:0]	Receive FIFO Buffer Data Byte 1		R
b23 to b16	RFDB2[7:0]	Receive FIFO Buffer Data Byte 2		R
b31 to b24	RFDB3[7:0]	Receive FIFO Buffer Data Byte 3		R

27.2.29 Receive FIFO Buffer Access Data Field 1 Registers (RSCAN0RFDF1x) (x = 0 to 7)

The RSCAN0RFDF1x registers are data registers that hold the values in the receive buffers.
When the value of the RMDLC[3:0] bits in the RSCAN0RFPTRx register is less than 1000b, data bytes for which no value is set are read as 00h.

Address(es): RSCAN.RSCAN0RFDF10 A007 8E0Ch, RSCAN.RSCAN0RFDF11 A007 8E1Ch, RSCAN.RSCAN0RFDF12 A007 8E2Ch, RSCAN.RSCAN0RFDF13 A007 8E3Ch, RSCAN.RSCAN0RFDF14 A007 8E4Ch, RSCAN.RSCAN0RFDF15 A007 8E5Ch, RSCAN.RSCAN0RFDF16 A007 8E6Ch, RSCAN.RSCAN0RFDF17 A007 8E7Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	RFDB4[7:0]	Receive FIFO Buffer Data Byte 4	Values of messages stored in the corresponding receive FIFO buffer can be read from these bits.	R
b15 to b8	RFDB5[7:0]	Receive FIFO Buffer Data Byte 5		R
b23 to b16	RFDB6[7:0]	Receive FIFO Buffer Data Byte 6		R
b31 to b24	RFDB7[7:0]	Receive FIFO Buffer Data Byte 7		R

27.2.30 Transmit/Receive FIFO Buffer Configuration and Control Registers k (RSCAN0CFCCk) (k = 3 to 5)

The RSCAN0CFCCk registers control the settings for the transmit/receive FIFO buffers (k = 3 to 5).

Address(es): RSCAN.RSCAN0CFCC3 A007 8124h, RSCAN.RSCAN0CFCC4 A007 8128h, RSCAN.RSCAN0CFCC5 A007 812Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b0	CFE	Transmit/Receive FIFO Buffer Enable	0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.	R/W
b1	CFRXIE	Transmit/Receive FIFO Receive Interrupt Enable	0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.	R/W
b2	CFTXIE	Transmit/Receive FIFO Transmit Interrupt Enable	0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b10 to b8	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: Setting prohibited	R/W
b11	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b12	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.	R/W
b15 to b13	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.	R/W

Bit	Symbol	Bit Name	Description	R/W
b17, b16	CFM[1:0]	Transmit/Receive FIFO Mode Select	b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode Settings other than above are prohibited.	R/W
b18	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.	R/W
b19	CFITR	Transmit/Receive FIFO Interval Timer Resolution Select	0: PCLKD/2 clock divided by the value of the ITRCP[15:0] bits 1: PCLKD/2 clock divided by (the value of the ITRCP[15:0] bits × 10)	R/W
b23 to b20	CFTML[3:0]	Transmit Buffer Link Configuration	Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.	R/W
b31 to b24	CFITT[7:0]	Message Transmission Interval Configuration	Set Value: 00h to FFh	R/W

CFE Bit (Transmit/Receive FIFO Buffer Enable)

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

CFRXIE Bit (Transmit/Receive FIFO Receive Interrupt Enable)

When this bit is set to 1 and the CFRXIF flag in the RSCAN0CFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.

Modify this bit with the CFE bit set to 0.

CFTXIE Bit (Transmit/Receive FIFO Transmit Interrupt Enable)

When this bit is set to 1 and the CFTXIF flag in the RSCAN0CFSTSk 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).

CFDC[2:0] Bits (Transmit/Receive FIFO Buffer Depth Configuration)

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 000b, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFIM Bit (Transmit/Receive FIFO Interrupt Source Select)

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

CFIGCV[2:0] Bits (Transmit/Receive FIFO Receive Interrupt Request Timing Select)

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 00b (receive mode) or 10b (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 001b (4 messages), set the CFIGCV[2:0] bits to 001b, 011b, 101b, or 111b.
Modify these bits only in global reset mode.

CFM[1:0] Bits (Transmit/Receive FIFO Mode Select)

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

CFITSS Bit (Transmit/Receive FIFO Interval Timer Clock Source Select)

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).

CFITR Bit (Transmit/Receive FIFO Interval Timer Resolution Select)

This bit is enabled when the CFITSS bit is 0.
When this bit is 0, the interval timer clock source is the PCLKD/2 clock divided by the value of the ITRCP[15:0] bits in the RSCAN0GCFG register.
When this bit is 1, the interval timer clock source is the PCLKD/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCAN0GCFG register × 10).
Modify this bit while the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

CFTML[3:0] Bits (Transmit Buffer Link Configuration)

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 01b (transmit mode) or 10b (gateway mode). There are three transmit/receive FIFO buffers on channel 1 (k = 3 to 5). Table 27.4 lists the actual transmit buffer numbers p linked to FIFO buffers k. See section 27.6, Transmission Functions, for operations of transmit/receive FIFO buffers k and transmit buffers p.
Setting the CFDC[2:0] bits to 001b 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 channel 1. Modify these bits only in global reset mode.

Table 27.4 Settings of the CFTML[3:0] Bits and Linking of Transmit Buffers p to Transmit/Receive FIFO Buffers k

Settings of CFTML[3:0] Bits (setting for each transmit/receive FIFO buffer k)	Transmit Buffers p Linked to Transmit/Receive FIFO Buffers (p = 16 to 31) Channel 1
0000b	Transmit buffer 16
0001b	Transmit buffer 17
0010b	Transmit buffer 18
0011b	Transmit buffer 19
0100b	Transmit buffer 20
0101b	Transmit buffer 21
0110b	Transmit buffer 22
0111b	Transmit buffer 23
1000b	Transmit buffer 24
1001b	Transmit buffer 25
1010b	Transmit buffer 26
1011b	Transmit buffer 27
1100b	Transmit buffer 28
1101b	Transmit buffer 29
1110b	Transmit buffer 30
1111b	Transmit buffer 31

CFITT[7:0] Bits (Message Transmission Interval Configuration)

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 01b (transmit mode) or 10b (gateway mode).
Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

27.2.31 Transmit/Receive FIFO Buffer Status Registers (RSCAN0CFSTSk) (k = 3 to 5)

The RSCAN0CFSTSk registers are status registers that indicate the state of the transmit/receive FIFO buffers (k = 3 to 5).

Address(es): RSCAN.RSCAN0CFSTS3 A007 8184h, RSCAN.RSCAN0CFSTS4 A007 8188h, RSCAN.RSCAN0CFSTS5 A007 818Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b0	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).	R
b1	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.	R
b2	CFMLT	Transmit/Receive FIFO Message Lost Flag	0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.	R/W *1
b3	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.	R/W *1
b4	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.	R/W *1
b7 to b5	—	Reserved	These bits are read as 0.	R
b15 to b8	CFMC[7:0]	Transmit/Receive FIFO Message Counter	The number of messages stored in the transmit/receive FIFO buffer.	R
b31 to b16	—	Reserved	These bits are read as 0.	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.

CFEMP Flag (Transmit/Receive FIFO Buffer Empty Status 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 00b: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: 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 transmission 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 00b or 10b: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01b: A value of FFh has been written to the RSCAN0CFPCTRk register after data was written to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers.

CFFLL Flag (Transmit/Receive FIFO Buffer Full Status 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 RSCAN0CFCCk register.

The CFFLL 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 RSCAN0CFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmission abort
- When the CFM[1:0] bits are set to 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: In channel reset mode

CFMLT Flag (Transmit/Receive FIFO Message Lost 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 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: 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 (Transmit/Receive FIFO Receive Interrupt Request 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 00b or 10b, and the factor selected by the CFIM bit in the RSCAN0CFCCk 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 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: 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”.

CFTXIF Flag (Transmit/Receive FIFO Transmit Interrupt Request 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 01b or 10b, and the factor selected by the CFIM bit in the RSCAN0CFCCk 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 00b: In global reset mode
- When the CFM[1:0] bits are set to 01b or 10b: 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”.

CFMC[7:0] Bits (Transmit/Receive FIFO Message Counter)

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCAN0CFCCk register.

- When CFM[1:0] value is 01b (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00b (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10b (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 00b: In global reset mode
- When CFM[1:0] value is 01b or 10b: In channel reset mode

27.2.32 Transmit/Receive FIFO Buffer Pointer Control Registers (RSCAN0CFPCTRk) (k = 3 to 5)

The RSCAN0CFPCTRk registers control the pointers to the transmit/receive FIFO buffers (k = 3 to 5).

Address(es): RSCAN.RSCAN0CFPCTR3 A007 81E4h, RSCAN.RSCAN0CFPCTR4 A007 81E8h, RSCAN.RSCAN0CFPCTR5 A007 81ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CFPC[7:0]	Transmit/Receive FIFO Pointer Control	• Receive mode: Writing FFh to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. • Transmit mode: Writing FFh to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. • Gateway mode: Setting prohibited	W
b31 to b8	—	Reserved	The write value should be 0.	W

CFPC[7:0] Bits (Transmit/Receive FIFO Pointer Control)

- Receive mode (CFM[1:0] value in the RSCAN0CFCCk register is 00b):
  Writing FFh 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 RSCAN0CFSTSk register is decremented. Read the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers to read messages from the transmit/receive FIFO buffer, and then write FFh to the CFPC[7:0] bits.
  When writing FFh to these bits, make sure that the CFE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCAN0CFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).
- Transmit mode (CFM[1:0] value in the RSCAN0CFCCk register is 01b):
  Writing FFh to the CFPC[7:0] bits stores the data written to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k 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 RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers before writing FFh to the CFPC[7:0] bits.
  When writing FFh to these bits, make sure that the CFE bit in the RSCAN0CFCCk register is set to 1 and the CFFLL flag in the RSCAN0CFSTSk register is 0 (the transmit/receive FIFO buffer is not full).
- Gateway mode (CFM[1:0] value in the RSCAN0CFCCk register is 10b):
  Setting prohibited

27.2.33 Transmit/Receive FIFO Buffer Access ID Registers (RSCAN0CFIDk) (k = 3 to 5)

The RSCAN0CFIDk registers are status registers that indicate the state of the ID and data formats of the received messages stored in the transmit/receive FIFO buffers (k = 3 to 5).
These registers are writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01b (transmit mode) and readable only when the CFM[1:0] value is 00b (receive mode). These registers should not be read or written when the CFM[1:0] value is 10b (gateway mode).

Address(es): RSCAN.RSCAN0CFID3 A007 8EB0h, RSCAN.RSCAN0CFID4 A007 8EC0h, RSCAN.RSCAN0CFID5 A007 8ED0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CFIDE	CFRTR	THLEN	CFID[28:0]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CFID[28:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b28 to b0	CFID[28:0]	Transmit/Receive FIFO Buffer ID Data Set	• When CFM[1:0] value is 01b (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 00b (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.	R/W
b29	THLEN	Transmission History Data Storage Enable	This bit is valid only when the CFM[1:0] value is 01b (transmit mode). 0: Transmission history data is not stored in the buffer. 1: Transmission history data is stored in the buffer.	R/W
b30	CFRTR	Transmit/Receive FIFO Buffer RTR	0: Data frame 1: Remote frame	R/W
b31	CFIDE	Transmit/Receive FIFO Buffer IDE	0: Standard ID 1: Extended ID	R/W

CFID[28:0] Bits (Transmit/Receive FIFO Buffer ID Data Set)

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00b.
When the CFM[1:0] value is 01b, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

THLEN Bit (Transmission History Data Storage Enable)

When this bit is set to 1, the transmission history data (label information, buffer number, and buffer type) of transmit messages is stored in the transmission history buffer after transmission is completed.
This bit is enabled when the CFM[1:0] value is 01b (transmit mode).

CFRTR Bit (Transmit/Receive FIFO Buffer RTR)

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 00b. When the CFM[1:0] value is 01b, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer.

CFIDE Bit (Transmit/Receive FIFO Buffer IDE)

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 00b. When the CFM[1:0] value is 01b, these bits are used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

27.2.34 Transmit/Receive FIFO Buffer Access Pointer Registers (RSCAN0CFPTRk) (k = 3 to 5)

The RSCAN0CFPTRk registers are status registers that indicate the data length, label information, and timestamp value of the received messages stored in the transmit/receive buffers (k = 3 to 5).
These registers are writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01b (transmit mode) and readable only when the CFM[1:0] value is 00b (receive mode). These registers should not be read or written when the CFM[1:0] value is 10b (gateway mode).

Address(es): RSCAN.RSCAN0CFPTR3 A007 8EB4h, RSCAN.RSCAN0CFPTR4 A007 8EC4h, RSCAN.RSCAN0CFPTR5 A007 8ED4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CFTS[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	CFTS[15:0]	Transmit/Receive FIFO Buffer Timestamp Data Indication	These bits are valid only when the CFM[1:0] value is 00b (receive mode). The timestamp value of the received message can be read.	R/W
b27 to b16	CFPTR[11:0]	Transmit/Receive FIFO Buffer Label Data Indication	• When CFM[1:0] value is 01b (transmit mode): Set the label information to be stored in the transmission history buffer. Only bits CFPTR[7:0] are valid. • When CFM[1:0] value is 00b (receive mode): The label information of the received message can be read.	R/W
b31 to b28	CFDLC[3:0]	Transmit/Receive FIFO Buffer DLC Data Indication	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	R/W

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 00b.

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 00b. When the CFM[1:0] value is 01b, the CFPTR[7:0] value is stored in the transmission history buffer when message transmission has been completed.

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 00b. When the CFM[1:0] value is 01b, 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 9 bytes or more, the actual transmit data defaults to 8 bytes.

27.2.35 Transmit/Receive FIFO Buffer Access Data Field 0 Registers (RSCAN0CFDF0k) (k = 3 to 5)

The RSCAN0CFDF0k registers are data registers that hold the values in the transmit/receive FIFO buffers (k = 3 to 5). These registers are writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01b (transmit mode) and readable only when the CFM[1:0] value is 00b (receive mode). When the CFDLC[3:0] value in the RSCAN0CFPTRk register is smaller than 1000b, data bytes for which no data is set are read as 00h.
These registers should not be read or written when the CFM[1:0] value is 10b (gateway mode).

Address(es): RSCAN.RSCAN0CFDF03 A007 8EB8h, RSCAN.RSCAN0CFDF04 A007 8EC8h, RSCAN.RSCAN0CFDF05 A007 8ED8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CFDB0[7:0]	Transmit/Receive FIFO Buffer Data Byte 0	• When CFM[1:0] value is 01b (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00b (receive mode): Values of messages stored in the corresponding transmit/receive FIFO buffer can be read from these bits.	R/W
b15 to b8	CFDB1[7:0]	Transmit/Receive FIFO Buffer Data Byte 1		R/W
b23 to b16	CFDB2[7:0]	Transmit/Receive FIFO Buffer Data Byte 2		R/W
b31 to b24	CFDB3[7:0]	Transmit/Receive FIFO Buffer Data Byte 3		R/W

27.2.36 Transmit/Receive FIFO Buffer Access Data Field 1 Registers (RSCAN0CFDF1k) (k = 3 to 5)

The RSCAN0CFDF1k registers are data registers that hold the values in the transmit/receive buffers (k = 3 to 5). These registers are writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01b (transmit mode) and readable only when the CFM[1:0] value is 00b (receive mode). When the CFDLC[3:0] value in the RSCAN0CFPTRk register is smaller than 1000b, data bytes for which no data is set are read as 00h.
These registers should not be read or written when the CFM[1:0] value is 10b (gateway mode).

Address(es): RSCAN.RSCAN0CFDF13 A007 8EBCh, RSCAN.RSCAN0CFDF14 A007 8ECCh, RSCAN.RSCAN0CFDF15 A007 8EDCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	CFDB4[7:0]	Transmit/Receive FIFO Buffer Data Byte 4	• When CFM[1:0] value is 01b (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00b (receive mode): Values of messages stored in the corresponding transmit/receive FIFO buffer can be read from these bits.	R/W
b15 to b8	CFDB5[7:0]	Transmit/Receive FIFO Buffer Data Byte 5		R/W
b23 to b16	CFDB6[7:0]	Transmit/Receive FIFO Buffer Data Byte 6		R/W
b31 to b24	CFDB7[7:0]	Transmit/Receive FIFO Buffer Data Byte 7		R/W

27.2.37 FIFO Empty Status Register (RSCAN0FESTS)

The RSCAN0FESTS register is a status register that indicates whether receive FIFOx (x = 0 to 7) and transmit/receive FIFOk (k = 3 to 5) are empty.

The RSCAN0FESTS register is set to 03FF FFFFh in global reset mode.

Address(es): RSCAN.RSCAN0FESTS A007 8238h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	CF5EMP	CF4EMP	CF3EMP	—	—	—	RF7EMP	RF6EMP	RF5EMP	RF4EMP	RF3EMP	RF2EMP	RF1EMP	RF0EMP
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b0	RF0EMP	Receive FIFO Buffer Empty Status Flag	0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message (x = 0 to 7)	R
b1	RF1EMP			R
b2	RF2EMP			R
b3	RF3EMP			R
b4	RF4EMP			R
b5	RF5EMP			R
b6	RF6EMP			R
b7	RF7EMP			R
b10 to b8	—	Reserved	These bits are read as 1.	R
b11	CF3EMP	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 = 3 to 5)	R
b12	CF4EMP			R
b13	CF5EMP			R
b16 to b14	—	Reserved	These bits are read as 1.	R
b31 to b17	—	Reserved	These bits are read as 0.	R

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCAN0RFSTSx 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.

CFkEMP Flag (k = 3 to 5)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCAN0CFSTSk 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.

27.2.38 FIFO Full Status Register (RSCAN0FFSTS)

The RSCAN0FFSTS register is a status register that indicates whether receive FIFOx (x = 0 to 7) and transmit/receive FIFOk (k = 3 to 5) are full.

The RSCAN0FFSTS register is cleared to 0000 0000h in global reset mode.

Address(es): RSCAN.RSCAN0FFSTS A007 823Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	CF5FLL	CF4FLL	CF3FLL	—	—	—	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

Bit	Symbol	Bit Name	Description	R/W
b0	RF0FLL	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)	R
b1	RF1FLL			R
b2	RF2FLL			R
b3	RF3FLL			R
b4	RF4FLL			R
b5	RF5FLL			R
b6	RF6FLL			R
b7	RF7FLL			R
b10 to b8	—	Reserved	These bits are read as 0.	R
b11	CF3FLL	Transmit/Receive FIFO Buffer Full Status Flag	0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full. (k = 3 to 5)	R
b12	CF4FLL			R
b13	CF5FLL			R
b31 to b14	—	Reserved	These bits are read as 0.	R/(W)

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCAN0RFSTSx 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.

CFkFLL Flag (k = 3 to 5)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCAN0CFSTSk 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.

27.2.39 FIFO Message Lost Status Register (RSCAN0FMSTS)

The RSCAN0FMSTS register is a status register that indicates whether messages in receive FIFOx (x = 0 to 7) and transmit/receive FIFOk (k = 3 to 5) have been lost.
The RSCAN0FFSTS register is cleared to 0000 0000h in global reset mode.

Address(es): RSCAN.RSCAN0FMSTS A007 8240h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	CF5MLT	CF4MLT	CF3MLT	—	—	—	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

Bit	Symbol	Bit Name	Description	R/W
b0	RF0MLT	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)	R
b1	RF1MLT			R
b2	RF2MLT			R
b3	RF3MLT			R
b4	RF4MLT			R
b5	RF5MLT			R
b6	RF6MLT			R
b7	RF7MLT			R
b10 to b8	—	Reserved	These bits are read as 0.	R
b11	CF3MLT	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 = 3 to 5)	R
b12	CF4MLT			R
b13	CF5MLT			R
b31 to b14	—	Reserved	These bits are read as 0.	R

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCAN0RFSTSx 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.

CFkMLT Flag (k = 3 to 5)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCAN0CFSTSk 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.

27.2.40 Receive FIFO Buffer Interrupt Flag Status Register (RSCAN0RFISTS)

The RSCAN0RFISTS register is a flag register that indicates interrupts from the receive FIFO buffer.
The RSCAN0RFISTS register is cleared to 0000 0000h in global reset mode.

Address(es): RSCAN.RSCAN0RFISTS A007 8244h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	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

Bit	Symbol	Bit Name	Description	R/W
b0	RF0IF	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)	R
b1	RF1IF			R
b2	RF2IF			R
b3	RF3IF			R
b4	RF4IF			R
b5	RF5IF			R
b6	RF6IF			R
b7	RF7IF			R
b31 to b8	—	Reserved	These bits are read as 0.	R

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCAN0RFSTSx 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.

27.2.41 Transmit/Receive FIFO Buffer Receive Interrupt Flag Status Register (RSCAN0CFRISTS)

The RSCAN0CFRISTS register is a flag register that indicates reception interrupts from the receive/transmit FIFO buffer.

The RSCAN0CFRISTS register is cleared to 0000 0000h in global reset mode.

Address(es): RSCAN.RSCAN0CFRISTS A007 8248h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	CF5RXIF	CF4RXIF	CF3RXIF	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0.	R
b3	CF3RXIF	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 = 3 to 5)	R
b4	CF4RXIF			R
b5	CF5RXIF			R
b31 to b6	—	Reserved	These bits are read as 0.	R

CFkRXIF Flag (k = 3 to 5)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCAN0CFSTSk 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.

27.2.42 Transmit/Receive FIFO Buffer Transmit Interrupt Flag Status Register (RSCAN0CFTISTS)

The RSCAN0CFTISTS register is a flag register that indicates transmission interrupts from the receive/transmit FIFO buffer.

The RSCAN0CFTISTS register is cleared to 0000 0000h in global reset mode.

Address(es): RSCAN.RSCAN0CFTISTS A007 824Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	CF5TXIF	CF4TXIF	CF3TXIF	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	—	Reserved	These bits are read as 0.	R
b3	CF3TXIF	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 = 3 to 5)	R
b4	CF4TXIF			R
b5	CF5TXIF			R
b31 to b6	—	Reserved	These bits are read as 0.	R

CFkTXIF Flag (k = 3 to 5)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCAN0CFSTSk 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.

27.2.43 Transmit Buffer Control Register (RSCAN0TMCp) (p = 16 to 31)

The RSCAN0TMCp registers control the settings for the transmit buffers (p = 16 to 31).
Set the RSCAN0TMCp registers to 00h when any of the following conditions is met.

- The RSCAN0TMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCAN0CFCCk register (p = 16 + the value of CFTML[3:0] bits) (k = 3 to 5).
- The RSCAN0TMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCAN0TXQCC1 register (p = 31 to (31 – the value of TXQDC[3:0] bits)).

Bits in the RSCAN0TMCp registers are all cleared to 0 in channel reset mode.
Modify the RSCAN0TMCp registers in channel communication mode or channel halt mode.

Address(es): RSCAN.RSCAN0TMC16 A007 8260h, RSCAN.RSCAN0TMC17 A007 8261h, RSCAN.RSCAN0TMC18 A007 8262h, RSCAN.RSCAN0TMC19 A007 8263h, RSCAN.RSCAN0TMC20 A007 8264h, RSCAN.RSCAN0TMC21 A007 8265h, RSCAN.RSCAN0TMC22 A007 8266h, RSCAN.RSCAN0TMC23 A007 8267h, RSCAN.RSCAN0TMC24 A007 8268h, RSCAN.RSCAN0TMC25 A007 8269h, RSCAN.RSCAN0TMC26 A007 826Ah, RSCAN.RSCAN0TMC27 A007 826Bh, RSCAN.RSCAN0TMC28 A007 826Ch, RSCAN.RSCAN0TMC29 A007 826Dh, RSCAN.RSCAN0TMC30 A007 826Eh, RSCAN.RSCAN0TMC31 A007 826Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMTR	Transmission Request	0: Transmission is not requested. 1: Transmission is requested.	R/W *1
b1	TMTAR	Transmission Abort Request	0: Transmission abort is not requested. 1: Transmission abort is requested.	R/W *1
b2	TMOM	One-Shot Transmission Enable	0: One-shot transmission is disabled. 1: One-shot transmission is enabled.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

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.

TMTR Bit (Transmission Request)

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.
- Transmission 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 RSCAN0TMSTSp register is 00b.

TMTAR Bit (Transmission Abort Request)

Setting this bit to 1 generates a transmission 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.
- Transmission 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.

TMOM Bit (One-Shot Transmission Enable)

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 RSCAN0TMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

27.2.44 Transmit Buffer Status Registers (RSCAN0TMSTSp) (p = 16 to 31)

The RSCAN0TMSTSp registers are status registers that indicate the state of the transmit buffers (p = 16 to 31).
Bits in the RSCAN0TMSTSp registers are all cleared to 0 in channel reset mode.

Address(es): RSCAN.RSCAN0TMSTS16 A007 82E0h, RSCAN.RSCAN0TMSTS17 A007 82E1h, RSCAN.RSCAN0TMSTS18 A007 82E2h, RSCAN.RSCAN0TMSTS19 A007 82E3h, RSCAN.RSCAN0TMSTS20 A007 82E4h, RSCAN.RSCAN0TMSTS21 A007 82E5h, RSCAN.RSCAN0TMSTS22 A007 82E6h, RSCAN.RSCAN0TMSTS23 A007 82E7h, RSCAN.RSCAN0TMSTS24 A007 82E8h, RSCAN.RSCAN0TMSTS25 A007 82E9h, RSCAN.RSCAN0TMSTS26 A007 82EAh, RSCAN.RSCAN0TMSTS27 A007 82EBh, RSCAN.RSCAN0TMSTS28 A007 82ECh, RSCAN.RSCAN0TMSTS29 A007 82EDh, RSCAN.RSCAN0TMSTS30 A007 82EEh, RSCAN.RSCAN0TMSTS31 A007 82EFh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	TMTSTS	Transmit Buffer Transmission Status Flag	0: Transmission is not in progress. 1: Transmission is in progress.	R
b2, b1	TMTRF[1:0]	Transmit Buffer Transmission Result Status Flag	b2 b1 0 0: Transmission is in progress or no transmission request is present. 0 1: Transmission abort has been completed. 1 0: Transmission has been completed (without transmission abort request). 1 1: Transmission has been completed (with transmission abort request).	R/W
b3	TMTRM	Transmit Buffer Transmission Request Status Flag	0: No transmission request is present. 1: A transmission request is present.	R
b4	TMTARM	Transmit Buffer Transmission Abort Request Status Flag	0: No transmission abort request is present. 1: A transmission abort request is present.	R
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TMTSTS Flag (Transmit Buffer Transmission Status 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.

TMTRF[1:0] Flag (Transmit Buffer Transmission Result Status Flag)

This flag indicates the result of transmission from the transmit buffer.
00b: Transmission is in progress or no transmission request is present.
01b: Transmission from the transmit buffer was aborted.
10b: Transmission has been completed with the TMTAR bit in the RSCAN0TMCp register set to 0 (transmission abort is not requested).
11b: Transmission has been completed with the TMTAR bit in the RSCAN0TMCp register set to 1 (transmission abort is requested).
Write 00b to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00b to this flag.

TMTRM Flag (Transmit Buffer Transmission Request Status Flag)

The TMTRM flag is set to 1 when the TMTR bit in the RSCAN0TMCp register is set to 1.
The TMTRM flag is set to 0 when the TMTR bit in the RSCAN0TMCp register is set to 0.

TMTARM Flag (Transmit Buffer Transmission Abort Request Status Flag)

The TMTARM flag is set to 1 when the TMTAR bit in the RSCAN0TMCp register is set to 1.
The TMTARM flag is set to 0 when the TMTAR bit in the RSCAN0TMCp register is set to 0.

27.2.45 Transmit Buffer Transmission Request Status Register 0 (RSCAN0TMTRSTS0)

The RSCAN0TMTRSTS0 register is a status register that indicates requests for the transmission by the transmit buffer.

Address(es): RSCAN.RSCAN0TMTRSTS0 A007 8350h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TMTRSTSp (p = 31 to 16)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0.	R
b31 to b16	TMTRSTSp	Transmit Buffer Transmission Request Status Flag p (p = 31 to 16)	0: No transmission request is present. 1: A transmission request is present.	R

TMTRSTSp Flags (p = 16 to 31) (Transmit Buffer Transmission Request Status Flags p)

These flags indicate the status of the TMTR bit in the RSCAN0TMCp 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 27.5 shows the bit assignment.

Table 27.5 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
16	1	16
.	.	.
.	.	.
30	1	30
31	1	31

27.2.46 Transmit Buffer Transmission Abort Request Status Register 0 (RSCAN0TMTARSTS0)

The RSCAN0TMTARSTS0 register is a status register that indicates requests for aborting transmission by the transmit buffers.

Address(es): RSCAN.RSCAN0TMTARSTS0 A007 8360h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TMTARSTSp (p = 31 to 16)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0.	R
b31 to b16	TMTARSTSp	Transmit Buffer Transmission Abort Request Status Flag p (p = 31 to 16)	0: No transmission abort request is present. 1: A transmission abort request is present.	R

TMTARSTSp Flags (Transmit Buffer Transmission Abort Request Status Flags p) (p = 16 to 31)

These flags indicate the status of the TMTAR bit in the RSCAN0TMCp register.

When the TMTAR bit is set to 1 (transmission 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 (transmission abort is not requested) or in channel reset mode.

Table 27.6 shows the bit assignment.

Table 27.6 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
16	1	16
. .	. .	. .
30	1	30
31	1	31

27.2.47 Transmit Buffer Transmission Complete Status Register 0 (RSCAN0TMTCSTS0)

The RSCAN0TMTCSTS0 register is a status register that indicates requests for the completion of transmission by the transmit buffer.

Address(es): RSCAN.RSCAN0TMTCSTS0 A007 8370h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TMTCSTSp (p = 31 to 16)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0.	R
b31 to b16	TMTCSTSp	Transmit Buffer Transmission Complete Status Flag p (p = 31 to 16)	0: Transmission has not been completed. 1: Transmission has been completed.	R

TMTCSTSp Flags (Transmit Buffer Transmission Complete Status Flags p) (p = 16 to 31)

When the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 10b (transmission has been completed (without transmission abort request)) or 11b (transmission has been completed (with transmission 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 00b or in channel reset mode.

Table 27.7 shows the bit assignment.

Table 27.7 TMTCSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
16	1	16
. .	. .	. .
30	1	30
31	1	31

27.2.48 Transmit Buffer Transmission Abort Status Register 0 (RSCAN0TMTASTS0)

The RSCAN0TMTASTS0 register is a status register that indicates requests for aborting transmission by the transmit buffers.

Address(es): RSCAN.RSCAN0TMTASTS0 A007 8380h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TMTASTSp (p = 31 to 16)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0.	R
b31 to b16	TMTASTSp	Transmit Buffer Transmission Abort Status Flag p (p = 31 to 16)	0: Transmission is not aborted 1: Transmission is aborted	R

TMTASTSp Flags (Transmit Buffer Transmission Abort Status Flags p) (p = 16 to 31)

When the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 01b (transmission 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 00b or in channel reset mode.

Table 27.8 shows the bit assignment.

Table 27.8 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
16	1	16
. .	. .	. .
30	1	30
31	1	31

27.2.49 Transmit Buffer Interrupt Enable Configuration Register 0 (RSCAN0TMIEC0)

The RSCAN0TMIEC0 register enables or disables requests for transmit buffer interrupts.

Address(es): RSCAN.RSCAN0TMIEC0 A007 8390h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TMIEp (p = 31 to 16)															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0.	R/W
b31 to b16	TMIEp	Transmit Buffer Interrupt Enable p (p = 31 to 16)	0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled	R/W

TMIEp Bits (Transmit Buffer Interrupt Enable p) (p = 16 to 31)

When any of these bits is set to 1 and the corresponding transmission has been completed, a CAN1 transmission interrupt request (transmit buffer interrupt) is generated.

Modify these bits when the TMTRM flag in the corresponding RSCAN0TMSTSp register is 0 (no transmission 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 27.9 shows the bit assignment.

Table 27.9 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
16	1	16
.	.	.
.	.	.
30	1	30
31	1	31

27.2.50 Transmit Buffer ID Registers (RSCAN0TMIDp) (p = 16 to 31)

The RSCAN0TMIDp registers control the ID and data formats of the messages for transmission (p = 16 to 31). Modify these registers when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmission request is present). If these registers are linked to transmit/receive FIFO buffers, do not write data to them. If these registers are allocated to the transmit queue, write data only to transmit buffer 31.

Address(es): RSCAN.RSCAN0TMID16 A007 9100h, RSCAN.RSCAN0TMID17 A007 9110h, RSCAN.RSCAN0TMID18 A007 9120h, RSCAN.RSCAN0TMID19 A007 9130h, RSCAN.RSCAN0TMID20 A007 9140h, RSCAN.RSCAN0TMID21 A007 9150h, RSCAN.RSCAN0TMID22 A007 9160h, RSCAN.RSCAN0TMID23 A007 9170h, RSCAN.RSCAN0TMID24 A007 9180h, RSCAN.RSCAN0TMID25 A007 9190h, RSCAN.RSCAN0TMID26 A007 91A0h, RSCAN.RSCAN0TMID27 A007 91B0h, RSCAN.RSCAN0TMID28 A007 91C0h, RSCAN.RSCAN0TMID29 A007 91D0h, RSCAN.RSCAN0TMID30 A007 91E0h, RSCAN.RSCAN0TMID31 A007 91F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	TMIDE	TMRTR	THLEN	TMID[28:0]												
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	TMID[28:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b28 to b0	TMID[28:0]	Transmit Buffer ID Data Configuration	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.	R/W
b29	THLEN	Transmission History Data Storage Enable	0: Transmission history data is not stored in the buffer. 1: Transmission history data is stored in the buffer.	R/W
b30	TMRTR	Transmit Buffer RTR	0: Data frame 1: Remote frame	R/W
b31	TMIDE	Transmit Buffer IDE	0: Standard ID 1: Extended ID	R/W

TMID[28:0] Bits (Transmit Buffer ID Data Configuration)

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

THLEN Bit (Transmission History Data Storage Enable)

With this bit set to 1, the transmission history data of the message transmitted (label information and the number and type) are stored in the transmission history buffer after transmission is completed.

TMRTR Bit (Transmit Buffer RTR)

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

TMIDE Bit (Transmit Buffer IDE)

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

27.2.51 Transmit Buffer Pointer Registers (RSCAN0TMPTRp) (p= 16 to 31)

The RSCAN0TMPTRp registers control the data length and label information for the messages for transmission (p = 16 to 31).

Modify these registers when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmission request is present). If these registers are linked to transmit/receive FIFO buffers, do not write data to them. If these registers are allocated to the transmit queue, write data only to transmit buffer 31.

Address(es): RSCAN.RSCAN0TMPTR16 A007 9104h, RSCAN.RSCAN0TMPTR17 A007 9114h, RSCAN.RSCAN0TMPTR18 A007 9124h, RSCAN.RSCAN0TMPTR19 A007 9134h, RSCAN.RSCAN0TMPTR20 A007 9144h, RSCAN.RSCAN0TMPTR21 A007 9154h, RSCAN.RSCAN0TMPTR22 A007 9164h, RSCAN.RSCAN0TMPTR23 A007 9174h, RSCAN.RSCAN0TMPTR24 A007 9184h, RSCAN.RSCAN0TMPTR25 A007 9194h, RSCAN.RSCAN0TMPTR26 A007 91A4h, RSCAN.RSCAN0TMPTR27 A007 91B4h, RSCAN.RSCAN0TMPTR28 A007 91C4h, RSCAN.RSCAN0TMPTR29 A007 91D4h, RSCAN.RSCAN0TMPTR30 A007 91E4h, RSCAN.RSCAN0TMPTR31 A007 91F4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b23 to b16	TMPTR[7:0]	Transmit Buffer Label Data Configuration	Set the label information to be stored in the transmission history buffer.	R/W
b27 to b24	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b31 to b28	TMDLC[3:0]	Transmit Buffer DLC Data Configuration	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	R/W

TMPTR[7:0] Bits (Transmit Buffer Label Data Configuration)

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmission history buffer.

TMDLC[3:0] Bits (Transmit Buffer DLC Data Configuration)

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 RSCAN0TMIDp register is set to 0 (data frame). If the data length is set to 9 bytes 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.

27.2.52 Transmit Buffer Data Field 0 Registers (RSCAN0TMDF0p) (p = 16 to 31)

The RSCAN0TMDF0p registers are data registers for writing data for transmission from the transmit buffer (p = 16 to 31).

Modify these registers when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmission request is present). If these registers are linked to transmit/receive FIFO buffers, do not write data to them. If these registers are allocated to the transmit queue, write data only to transmit buffer 31.

Address(es): RSCAN.RSCAN0TMDF016 A007 9108h, RSCAN.RSCAN0TMDF017 A007 9118h, RSCAN.RSCAN0TMDF018 A007 9128h, RSCAN.RSCAN0TMDF019 A007 9138h, RSCAN.RSCAN0TMDF020 A007 9148h, RSCAN.RSCAN0TMDF021 A007 9158h, RSCAN.RSCAN0TMDF022 A007 9168h, RSCAN.RSCAN0TMDF023 A007 9178h, RSCAN.RSCAN0TMDF024 A007 9188h, RSCAN.RSCAN0TMDF025 A007 9198h, RSCAN.RSCAN0TMDF026 A007 91A8h, RSCAN.RSCAN0TMDF027 A007 91B8h, RSCAN.RSCAN0TMDF028 A007 91C8h, RSCAN.RSCAN0TMDF029 A007 91D8h, RSCAN.RSCAN0TMDF030 A007 91E8h, RSCAN.RSCAN0TMDF031 A007 91F8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	TMDB0[7:0]	Transmit Buffer Data Byte 0	Set the transmit buffer data.	R/W
b15 to b8	TMDB1[7:0]	Transmit Buffer Data Byte 1		R/W
b23 to b16	TMDB2[7:0]	Transmit Buffer Data Byte 2		R/W
b31 to b24	TMDB3[7:0]	Transmit Buffer Data Byte 3		R/W

27.2.53 Transmit Buffer Data Field 1 Registers (RSCAN0TMDF1p) (p = 16 to 31)

The RSCAN0TMDF1p registers are data registers for writing data for transmission from the transmit buffer (p = 16 to 31).

Modify these registers when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmission request is present). If these registers are linked to transmit/receive FIFO buffers, do not write data to them. If these registers are allocated to the transmit queue, write data only to transmit buffer 31.

Address(es): RSCAN.RSCAN0TMDF116 A007 910Ch, RSCAN.RSCAN0TMDF117 A007 911Ch, RSCAN.RSCAN0TMDF118 A007 912Ch, RSCAN.RSCAN0TMDF119 A007 913Ch, RSCAN.RSCAN0TMDF120 A007 914Ch, RSCAN.RSCAN0TMDF121 A007 915Ch, RSCAN.RSCAN0TMDF122 A007 916Ch, RSCAN.RSCAN0TMDF123 A007 917Ch, RSCAN.RSCAN0TMDF124 A007 918Ch, RSCAN.RSCAN0TMDF125 A007 919Ch, RSCAN.RSCAN0TMDF126 A007 91ACh, RSCAN.RSCAN0TMDF127 A007 91BCh, RSCAN.RSCAN0TMDF128 A007 91CCh, RSCAN.RSCAN0TMDF129 A007 91DCh, RSCAN.RSCAN0TMDF130 A007 91ECh, RSCAN.RSCAN0TMDF131 A007 91FCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	TMDB4[7:0]	Transmit Buffer Data Byte 4	Set the transmit buffer data.	R/W
b15 to b8	TMDB5[7:0]	Transmit Buffer Data Byte 5		R/W
b23 to b16	TMDB6[7:0]	Transmit Buffer Data Byte 6		R/W
b31 to b24	TMDB7[7:0]	Transmit Buffer Data Byte 7		R/W

27.2.54 Transmit Queue Configuration and Control Register (RSCAN0TXQCC1)

The RSCAN0TXQCC1 register controls the settings for the transmit queue of channel 1 (CAN1).

Address(es): RSCAN.RSCAN0TXQCC1 A007 83A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	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

Bit	Symbol	Bit Name	Description	R/W
b0	TXQE	Transmit Queue Enable	0: The transmit queue is not used. 1: The transmit queue is used.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b11 to b8	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.	R/W
b12	TXQIE	Transmit Queue Interrupt Enable	0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.	R/W
b13	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.	R/W
b31 to b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

TXQE Bit (Transmit Queue Enable)

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 0010b or more.

TXQDC[3:0] Bits (Transmit Queue Depth Configuration)

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 31 to 16 (CAN1), as shown in Table 27.10. For examples of how buffer allocation is done, see Figure 27.9 in section 27.6, Transmission Functions. Modify these bits only in channel reset mode.

Table 27.10 Transmit Buffers p Allocated to the Transmit Queue of Channel 1 (CAN1)

	Transmit Buffers p Allocated to the Transmit Queue (p = 16 to 31)
Settings of TXQDC[3:0] Bits	**Channel 1**
0000b	Setting prohibited
0001b	Setting prohibited
0010b	Transmit buffer 31 to 29
0011b	Transmit buffer 31 to 28
0100b	Transmit buffer 31 to 27
0101b	Transmit buffer 31 to 26
0110b	Transmit buffer 31 to 25
0111b	Transmit buffer 31 to 24
1000b	Transmit buffer 31 to 23
1001b	Transmit buffer 31 to 22
1010b	Transmit buffer 31 to 21
1011b	Transmit buffer 31 to 20
1100b	Transmit buffer 31 to 19
1101b	Transmit buffer 31 to 18
1110b	Transmit buffer 31 to 17
1111b	Transmit buffer 31 to 16

TXQIE Bit (Transmit Queue Interrupt Enable)

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, a CAN1 transmission interrupt request (transmit queue interrupt) is generated.
Set the TXQE bit to 0 before modifying the TXQIE bit.

TXQIM Bit (Transmit Queue Interrupt Source Select)

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

27.2.55 Transmit Queue Status Registers (RSCAN0TXQSTS1)

The RSCAN0TXQSTS1 register is a status register that indicates the state of the transmit queue.

Address(es): RSCAN.RSCAN0TXQSTS1 A007 83C4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQFLL	TXQEMP
Value after reset:	0	0	0	x	x	x	x	x	0	0	0	0	0	0	0	1

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	TXQEMP	Transmit Queue Empty Status Flag	0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).	R
b1	TXQFLL	Transmit Queue Full Status Flag	0: The transmit queue is not full. 1: The transmit queue is full.	R
b2	TXQIF	Transmit Queue Interrupt Request Flag	0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.	R/W *1
b7 to b3	—	Reserved	These bits are read as 0.	R
b12 to b8	—	Reserved	These bits are read as an undefined value.	R
b31 to b13	—	Reserved	These bits are read as 0.	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.

TXQEMP Flag (Transmit Queue Empty Status 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

TXQFLL Flag (Transmit Queue Full Status 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 RSCAN0TXQCC1 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

TXQIF Flag (Transmit Queue Interrupt Request Flag)

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCAN0TXQCC1 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 RSCAN0TXQCC1 register to 0 (the transmit queue is not used).

27.2.56 Transmit Queue Pointer Control Registers (RSCAN0TXQPCTR1)

The RSCAN0TXQPCTR1 register controls the pointers to the transmit queue.

Address(es): RSCAN.RSCAN0TXQPCTR1 A007 83E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	TXQPC[7:0]	Transmit Queue Pointer Control	Writing FFh to these bits moves the write pointer to the transmit queue to the next queue buffer.	W
b31 to b8	—	Reserved	The write value should be 0.	W

TXQPC[7:0] Bits (Transmit Queue Pointer Control)

Writing FFh to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmission request of the message. Write transmit messages to the RSCAN0TMID31, RSCAN0TMPTR31, RSCAN0TMDF031, and RSCAN0TMDF131 registers before writing FFh to the TXQPC[7:0] bits.

When writing FFh to these bits, make sure that the TXQE bit in the RSCAN0TXQCC1 register is set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCAN0TXQSTS1 register is 0 (the transmit queue is not full).

27.2.57 Transmission History Configuration and Control Register (RSCAN0THLCC1)

The RSCAN0THLCC1 register controls the settings for transmission history.

Address(es): RSCAN.RSCAN0THLCC1 A007 8404h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	THLE	Transmission History Buffer Enable	0: Transmission history buffer is not used. 1: Transmission history buffer is used.	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b8	THLIE	Transmission History Interrupt Enable	0: Transmission history interrupt is disabled. 1: Transmission history interrupt is enabled.	R/W
b9	THLIM	Transmission History Interrupt Source Select	0: When 12 sets of data have been stored in the transmission history buffer 1: When a single set of transmission history data has been stored	R/W
b10	THLDTE	Transmission 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	R/W
b31 to b11	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

THLE Bit (Transmission History Buffer Enable)

Setting this bit to 1 makes the transmission history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmission history data of transmit messages is stored in the transmission history buffer.

Modify this bit in channel communication mode or channel halt mode.

THLIE Bit (Transmission History Interrupt Enable)

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a CAN1 transmission interrupt request (transmission history interrupt) is generated. Modify the THLIE bit only when the THLE bit set to 0.

THLIM Bit (Transmission History Interrupt Source Select)

This bit is used to select a transmission history interrupt source.

Modify this bit only in channel reset mode.

THLDTE Bit (Transmission History Target Buffer Select)

When this bit is set to 0, the transmission history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmission history buffer. When this bit is set to 1, the transmission history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmission history buffer.

Modify this bit only in channel reset mode.

27.2.58 Transmission History Status Register (RSCAN0THLSTS1)

The RSCAN0THLSTS1 register is a status register that indicates the state of transmission history.

Address(es): RSCAN.RSCAN0THLSTS1 A007 8424h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	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

Bit	Symbol	Bit Name	Description	R/W
b0	THLEMP	Transmission History Buffer Empty Status Flag	0: Transmission history buffer contains unread data. 1: Transmission history buffer contains no unread data (buffer empty).	R
b1	THLFLL	Transmission history Buffer Full Status Flag	0: Transmission history buffer is not full. 1: Transmission history buffer is full.	R
b2	THLELT	Transmission History Buffer Overflow Flag	0: Transmission history buffer overflow has not occurred. 1: Transmission history buffer overflow has occurred.	R/W *1
b3	THLIF	Transmission History Interrupt Request Flag	0: No transmission history interrupt request is present. 1: A transmission history interrupt request is present.	R/W *1
b7 to b4	—	Reserved	These bits are read as 0.	R
b12 to b8	THLMC[4:0]	Transmission History Buffer Unread Data Counter	These bits indicate the number of unread data sets stored in the transmission history buffer.	R
b31 to b13	—	Reserved	These bits are read as 0.	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.

THLEMP Flag (Transmission History Buffer Empty Status Flag)

The THLEMP flag is cleared to 0 when even a single set of transmission history data has been stored in the transmission history buffer.

This flag is set to 1 when all the data in the transmission history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCAN0THLCC1 register is set to 0 (the transmission history buffer is not used).

THLFLL Flag (Transmission History Buffer Full Status Flag)

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmission history buffer, and is cleared to 0 when the number of data sets stored in the transmission 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 RSCAN0THLCC1 register is set to 0 (the transmission history buffer is not used).

THLELT Flag (Transmission History Buffer Overflow Flag)

The THLELT flag is set to 1 when an attempt is made to store new transmission history data while the transmission 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”.

THLIF Flag (Transmission History Interrupt Request Flag)

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCAN0THLCC1 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”.

THLMC[4:0] Bits (Transmission History Buffer Unread Data Counter)

These bits indicate the number of unread data sets stored in the transmission history buffer.

27.2.59 Transmission History Access Register (RSCAN0THLACC1)

The RSCAN0THLACC1 register is a status register that indicates the contents of the transmission history data held in the transmission history buffer.

Address(es): RSCAN.RSCAN0THLACC1 A007 9804h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	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

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	BT[2:0]	Buffer Type Data Indication	b2 b1 b0 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 1 0 0: Transmit queue	R
b6 to b3	BN[3:0]	Buffer Number Data Indication	The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.	R
b7	—	Reserved	This bit is read as 0.	R
b15 to b8	TID[7:0]	Label Data Indication	The label information of stored data can be read.	R
b31 to b16	—	Reserved	These bits are read as 0.	R

BT[2:0] Bits (Buffer Type Data Indication)

These bits indicate the type of the transmit source buffer in the transmission history data stored in the transmission history buffer.

BN[3:0] Bits (Buffer Number Data Indication)

These bits indicate the transmit source buffer number in the transmission history data stored in the transmission history buffer.

TID[7:0] Bits (Label Data Indication)

These bits indicate the label information of transmission history data stored in the transmission history buffer.

27.2.60 Transmission History Pointer Control Register (RSCAN0THLPCTR1)

The RSCAN0THLPCTR1 register controls the pointers to the transmission history buffer.

Address(es): RSCAN.RSCAN0THLPCTR1 A007 8444h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	THLPC[7:0]	Transmission History List Pointer Control	Writing FFh to these bits moves the read pointer to the next unread data in the transmission history buffer.	W
b31 to b8	—	Reserved	The write value should be 0.	W

THLPC[7:0] Bits (Transmission History List Pointer Control)

When the THLPC[7:0] bits are set to FFh, the read pointer moves to the next data in the transmission history buffer. At this time, the THLMC[4:0] (transmission history buffer unread data counter) value in the RSCAN0THLSTS1 register is decremented. Write FFh to the THLPC[7:0] bits after reading from the RSCAN0THLACC1 register.

When writing FFh to these bits, make sure that the THLE bit in the RSCAN0THLCC1 register is set to 1 (the transmission history buffer is used) and the THLEMP flag in the RSCAN0THLSTS1 register is 0.

27.2.61 Global Test Configuration Register (RSCAN0GTSTCFG)

The RSCAN0GTSTCFG register controls the test settings for the entire RSCAN module.
Modify the RSCAN0GTSTCFG register only in global test mode.

Address(es): RSCAN.RSCAN0GTSTCFG A007 8468h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b22 to b16	RTMPS[6:0]	RAM Test Page Configuration	Set a value within a range of page 0 (00h) to page 28 (1Ch).	R/W
b31 to b23	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RTMPS[6:0] Bits (RAM Test Page Configuration)

These bits are used to set the RAM test target page number for RAM test. Set a value in the range of 00h to 1Ch, inclusive.

27.2.62 Global Test Control Register (RSCAN0GTSTCTR)

The RSCAN0GTSTCTR register controls operations for the RAM test and inter-channel communication test.

Address(es): RSCAN.RSCAN0GTSTCTR A007 846Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	RTME	RAM Test Enable	0: RAM test is disabled. 1: RAM test is enabled.	R/W
b31 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

RTME Bit (RAM Test Enable)

Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode.

1. Set the GMDC[1:0] bits in the RSCAN0GCTR register to 10b (Global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

For writing the RTME bit, unlock the write protection on this bit by the global lock key register (RSCAN0GLOCKK).

27.2.63 Global Lock Key Register (RSCAN0GLOCKK)

The RSCAN0GLOCKK register unlocks protection of the special test bit and is write only.
For the protection unlock data, see section 27.9.4.2, Procedure for Unlocking the Protection.

Address(es): RSCAN.RSCAN0GLOCKK A007 847Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	LOCK[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	LOCK[15:0]	Lock Key	These bits are key bits to unlock protection of test mode.	W*1
b31 to b16	—	Reserved	The write value should be 0.	W

Note 1. Writing to these bits is effective only when the RSCAN module is in global test mode.

LOCK[15:0] Bits (Lock Key)

Writing the sequence of protection unlock data to the LOCK[15:0] bits enables writing 1 to the RTME bit in the RSCAN0GTSTCTR register.
After the protection has been unlocked, writing to the I/O register area (A007 8000h to A007 84FFh) of the CAN (i.e. to a location that is not in the RAM) leads to re-enabling of protection.
Reading from the I/O register area of the CAN or reading from or writing to other areas does not lead to re-enabling of protection.

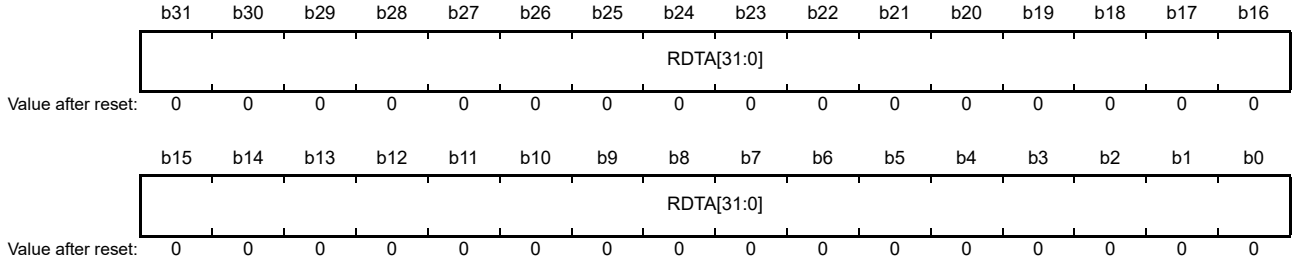
27.2.64 RAM Test Page Access Registers (RSCAN0RPGACCr) (r = 0 to 63)

The RSCAN0RPGACCr register controls test access to RAM data.

Modify the RSCAN0RPGACCr register in global test mode with the RTME bit in the RSCAN0GTSTCTR register set to 1 (RAM test is enabled).

The RSCAN0RPGACCr register is readable and writable when the RTME bit is set to 1.

Address(es): RSCAN.RSCAN0RPGACC0 A007 9900h, RSCAN.RSCAN0RPGACC1 A007 9904h,
RSCAN.RSCAN0RPGACC2 A007 9908h, RSCAN.RSCAN0RPGACC3 A007 990Ch,
RSCAN.RSCAN0RPGACC4 A007 9910h, RSCAN.RSCAN0RPGACC5 A007 9914h,
RSCAN.RSCAN0RPGACC6 A007 9918h, RSCAN.RSCAN0RPGACC7 A007 991Ch,
RSCAN.RSCAN0RPGACC8 A007 9920h, RSCAN.RSCAN0RPGACC9 A007 9924h,
RSCAN.RSCAN0RPGACC10 A007 9928h, RSCAN.RSCAN0RPGACC11 A007 992Ch,
RSCAN.RSCAN0RPGACC12 A007 9930h, RSCAN.RSCAN0RPGACC13 A007 9934h,
RSCAN.RSCAN0RPGACC14 A007 9938h, RSCAN.RSCAN0RPGACC15 A007 993Ch,
RSCAN.RSCAN0RPGACC16 A007 9940h, RSCAN.RSCAN0RPGACC17 A007 9944h,
RSCAN.RSCAN0RPGACC18 A007 9948h, RSCAN.RSCAN0RPGACC19 A007 994Ch,
RSCAN.RSCAN0RPGACC20 A007 9950h, RSCAN.RSCAN0RPGACC21 A007 9954h,
RSCAN.RSCAN0RPGACC22 A007 9958h, RSCAN.RSCAN0RPGACC23 A007 995Ch,
RSCAN.RSCAN0RPGACC24 A007 9960h, RSCAN.RSCAN0RPGACC25 A007 9964h,
RSCAN.RSCAN0RPGACC26 A007 9968h, RSCAN.RSCAN0RPGACC27 A007 996Ch,
RSCAN.RSCAN0RPGACC28 A007 9970h, RSCAN.RSCAN0RPGACC29 A007 9974h,
RSCAN.RSCAN0RPGACC30 A007 9978h, RSCAN.RSCAN0RPGACC31 A007 997Ch,
RSCAN.RSCAN0RPGACC32 A007 9980h, RSCAN.RSCAN0RPGACC33 A007 9984h,
RSCAN.RSCAN0RPGACC34 A007 9988h, RSCAN.RSCAN0RPGACC35 A007 998Ch,
RSCAN.RSCAN0RPGACC36 A007 9990h, RSCAN.RSCAN0RPGACC37 A007 9994h,
RSCAN.RSCAN0RPGACC38 A007 9998h, RSCAN.RSCAN0RPGACC39 A007 999Ch,
RSCAN.RSCAN0RPGACC40 A007 99A0h, RSCAN.RSCAN0RPGACC41 A007 99A4h,
RSCAN.RSCAN0RPGACC42 A007 99A8h, RSCAN.RSCAN0RPGACC43 A007 99ACh,
RSCAN.RSCAN0RPGACC44 A007 99B0h, RSCAN.RSCAN0RPGACC45 A007 99B4h,
RSCAN.RSCAN0RPGACC46 A007 99B8h, RSCAN.RSCAN0RPGACC47 A007 99BCh,
RSCAN.RSCAN0RPGACC48 A007 99C0h, RSCAN.RSCAN0RPGACC49 A007 99C4h,
RSCAN.RSCAN0RPGACC50 A007 99C8h, RSCAN.RSCAN0RPGACC51 A007 99CCh,
RSCAN.RSCAN0RPGACC52 A007 99D0h, RSCAN.RSCAN0RPGACC53 A007 99D4h,
RSCAN.RSCAN0RPGACC54 A007 99D8h, RSCAN.RSCAN0RPGACC55 A007 99DCh,
RSCAN.RSCAN0RPGACC56 A007 99E0h, RSCAN.RSCAN0RPGACC57 A007 99E4h,
RSCAN.RSCAN0RPGACC58 A007 99E8h, RSCAN.RSCAN0RPGACC59 A007 99ECh,
RSCAN.RSCAN0RPGACC60 A007 99F0h, RSCAN.RSCAN0RPGACC61 A007 99F4h,
RSCAN.RSCAN0RPGACC62 A007 99F8h, RSCAN.RSCAN0RPGACC63 A007 99FCh



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RDTA[31:0]	RAM Data Test Access	Data can be read and written in RSCAN RAM.	R/W

27.2.65 RSCAN ECC Control Register (ECCRCANCTL)

The ECCRCANCTL register controls the mode of the ECC for the RSCAN.
Bits 7 and 3 should be set (written) while the RSCAN is not operating.
Before writing to bit 7, set the EMCA1 and EMCA0 bits to 01b.

Address(es): RSCAN.ECCRCANCTL A007 B000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECDEDF7	ECSEDF7	ECDEDF6	ECSEDF6	ECDEDF5	ECSEDF5	ECDEDF4	ECSEDF4	ECDEDF3	ECSEDF3	ECDEDF2	ECSEDF2	ECDEDF1	ECSEDF1	ECDEDF0	ECSEDF0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	EMCA1	EMCA0	—	—	ECOVFF	ECER2C	ECER1C	—	ECTHM	ECERVF	EC1ECP	EC2EDIC	EC1EDIC	ECER2F	ECER1F	ECEMF
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECEMF	ECC Error Indication Flag	This flag indicates whether an error exists in the current read data. This bit is updated whenever the RAM reads data. This bit might be set if it is read before initialization of the RAM. This bit is cleared at the time of through mode enable selection (ECTHM = 1) and when there is no 1 bit error in the decode circuit input data. 0: The currently-read RAM data does not have bit errors. 1: The currently-read RAM data have bit errors.	R
b1	ECER1F	1-Bit Error Detection/Correction Flag	This flag indicates whether 1-bit error is detected during read access to the RAM when error detection is enabled (ECTHM = 0). Write 1 to the ECER1C bit (bit 9) to clear the flag. This bit is cleared at the time of through mode enable selection (ECTHM = 1). 0: 1-bit error has not occurred since this bit was cleared. 1: 1-bit error has occurred.	R
b2	ECER2F	2-Bit Error Detection Flag	This flag indicates whether 2-bit error is detected during read access to the RAM when error detection is enabled (ECTHM = 0). When 2-bit error interrupt is enabled (EC2EDIC = 1) and this flag is set, a 2-bit ECC error interrupt (INTECCDCNRAM) is output. Write 1 to the ECER2C bit (bit 10) to clear the flag. This bit is cleared at the time of through mode enable selection (ECTHM = 1). If 2-bit error is detected again while this bit is set, an interrupt signal will not be generated. 0: 2-bit error has not occurred since this bit was cleared. 1: 2-bit error has occurred.	R
b3	EC1EDIC	1-Bit Error Detection Interrupt Control	This bit controls whether to output a 1-bit ECC error source signal to the ECM when a 1-bit error is detected. 0: When a 1-bit error is detected, the error source signal is not output. 1: When a 1-bit error is detected, the error source signal is output.	R/W
b4	EC2EDIC	2-Bit Error Detection Interrupt Control	This bit controls whether to output a 2-bit ECC error source signal to the ECM when a 2-bit error is detected. 0: When a 2-bit error is detected, the error source signal is not output. 1: When a 2-bit error is detected, the error source signal is output.	R/W

Bit	Symbol	Bit Name	Description	R/W
b5	EC1ECP	1-Bit Error Correction Enable	This bit specifies whether to enable or disable 1-bit error correction when the ECC error detection/correction is enabled. 0: When a 1-bit error is detected, the error will be corrected. 1: When a 1-bit error is detected, the error will not be corrected.	R/W
b6	ECERVF	ECC Error Detection Enable Flag	This bit selects enabling or disabling of error detection. When writing to this bit, (0, 1) must be written to (EMCA1, EMCA0) at the same time. 0: Error detection is disabled. 1: Error detection is enabled.	R/W
b7	ECTHM	ECC Function Through Mode Select	This bit is used to set enabling and disabling of ECC. Setting this bit to 1 disables ECC. When writing to this bit, (0, 1) must be written to (EMCA1, EMCA0) at the same time. 0: Through mode is disabled (normal operation mode). 1: Through mode is enabled (ECC disabled).	R/W
b8	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b9	ECER1C	1-Bit ECC Error Correction Accumulation Flag Clear	This bit is used to clear the 1 bit error detection/correction flag of ECER1F (bit 1). This bit is always read as 0. Writing 0 is ignored. Write 1 to this bit while the ECER1F bit is set to clear the ECER1F bit. When a conflict between this bit writing and ECER1F bit setting occurs, writing to this bit is given priority.	R/W *1
b10	ECER2C	2-Bit ECC Error Detection Flag Clear	This bit is used to clear the 2 bit error detection flag of ECER2F (bit 2). This bit is always read as 0. Writing 0 is ignored. Write 1 to this bit while the ECER2F bit is set to clear the ECER2F bit. When a conflict between this bit writing and ECER2F bit setting occurs, writing to this bit is given priority.	R/W *1
b11	ECOVFF	ECC Overflow Detection Flag	When an ECC error is detected while all ECCRCANEADz registers (z = 0 to 7) hold addresses at which errors were detected, this bit is set and the RSCAN overflow error signal is output. 0: ECC overflow is not detected. 1: ECC overflow is detected.	R
b13, b12	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b14	EMCA0	Access Control 0 to ECC Mode Selection	These bits specify whether modifying the ECTHM bit (bit 7) is disabled or enabled. The value written to these bits is not retained. When these bits are read, the read value is always 0. When these bits are 01b, writing to bits 7 and 6 is enabled.	R/W *1
b15	EMCA1	Access Control 1 to ECC Mode Selection		R/W *1
b31 to b16	ECDEDFz	2-Bit ECC Error Detection Flag	This flag bit indicates whether the error stored in the ECCRCANEADz register (z = 0 to 7) is a 2-bit error. 0: 2-bit error has not occurred. 1: 2-bit error has occurred.	R
	ECSEDFz	1-Bit ECC Error Detection Flag	This flag bit indicates whether the error stored in the ECCRCANEADz register (z = 0 to 7) is a 1-bit error. 0: 1-bit error has not occurred. 1: 1-bit error has occurred.	R

Note 1. These bits are always read as 0.
Note: Bits 2 and 1 should be cleared when the ECC error indication flag (ECEMF) is not set.
We recommend initializing the RAM before clearing bits 2 and 1.

ECERVF Bit (ECC Error Detection Enable Flag)

This bit selects enabling or disabling of error detection. When writing to this bit, (0, 1) must be written to (EMCA1, EMCA0) at the same time. Error detection is disabled when the ECTHM bit is set to enable through mode. Table 27.11 shows the relationship between the ECERVF and ECTHM bits and the error detection setting.

Table 27.11 ECERVF and ECTHM Bits and Error Detection Setting

ECTHM Bit	ECERVF Bit	Error Detection Setting
0	0	Error detection disabled
0	1	Error detection enabled
1	0	Error detection disabled (through mode)
1	1	Error detection disabled (through mode)

ECOVFF Bit (ECC Overflow Detection Flag)

When a further ECC error is detected while all ECCRCANEADz registers (z = 0 to 7) hold addresses at which errors were detected, this bit is set and the RSCAN overflow error signal is output to the error control module (ECM). For details of the ECM, see section 32, Error Control Module (ECM).
Furthermore, if an overflow error is detected while this bit is set to 1, the RSCAN overflow error signal is output again.
This bit is cleared by writing 1 to both the ECER2C and ECER1C bits.

ECDEDFz Bit (2-Bit ECC Error Detection Flag) (z = 0 to 7)

This flag bit indicates the detection of 2-bit ECC errors and the address where the error was found is stored in the ECCRCANEADz register.
When an ECC overflow is detected (i.e., the ECOVFF bit = 1), this bit is not set even if the error is a 2-bit error.
This bit is cleared by writing 1 to both the ECER2C and ECER1C bits.

ECSEDFz Bit (1-Bit ECC Error Detection Flag) (z = 0 to 7)

This flag bit indicates the detection of 1-bit ECC errors and the address where the error was found is stored in the ECCRCANEADz register.
When an ECC overflow is detected (i.e., the ECOVFF bit = 1), this bit is not set even if the error is a 1-bit error.
This bit is cleared by writing 1 to both the ECER2C and ECER1C bits.

27.2.66 RSCAN ECC Error Address Register z (ECCRCANEADz) (z = 0 to 7)

ECCRCANEADz (z = 0 to 7) are read-only registers that automatically store and retain the values of addresses where ECC errors have occurred. If an ECC error occurs while ECC error detection is enabled, the buffer RAM addresses where errors are found are captured in order from z = 0.

Note 1. If an ECC error is detected again at the same address as that currently stored in the ECCRCANEADz register, the detected address is simply discarded.

Note 2. When 1-bit errors occur multiple times and all ECCRCANEADz registers hold only addresses where 1-bit errors were detected, if an overflow occurs due to the detection of a 2-bit error, the ECCRCANEAD7 register is overwritten with the address where the 2-bit error was detected and the value is retained.
Even if all ECCRCANEADz registers overflow due to the detection of a 1-bit error while only the addresses where 2-bit errors were detected are being held, the address where a 1-bit error was detected is discarded.

Address(es): RSCAN.ECCRCANEAD0 A007 B010h, RSCAN.ECCRCANEAD1 A007 B014h, RSCAN.ECCRCANEAD2 A007 B018h, RSCAN.ECCRCANEAD3 A007 B01Ch, RSCAN.ECCRCANEAD4 A007 B020h, RSCAN.ECCRCANEAD5 A007 B024h, RSCAN.ECCRCANEAD6 A007 B028h, RSCAN.ECCRCANEAD7 A007 B02Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	ECEADz[10:0]										
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b10 to b0	ECEADz [10:0]	ECC Error Address Storage	When an ECC error is detected in the buffer RAM of the RSCAN, the RAM address where the error was detected is stored and retained in these bits. The ECCRCANCTL.ECDEDFz and ECSEDFz bits can be used to judge whether the ECC error at the address was a 1-bit error or 2-bit error.	R
b31 to b11	—	Reserved	These bits are read as 0.	R

27.3 Interrupt Sources

The RSCAN module has five interrupt signals connected to the interrupt controller, and these are grouped into global interrupts and channel interrupts. Also, multiple interrupt sources in the RSCAN module are grouped into each of these five interrupt signals. Table 27.12 lists the CAN interrupt sources.

- Global interrupts (2 sources):
  1. CAN receive FIFO interrupt
  2. CAN global error interrupt
- Channel interrupts (3 sources):
  1. CAN1 transmit interrupt
     - CAN1 transmission complete interrupt
     - CAN1 transmission abort interrupt
     - CAN1 transmit/receive FIFO transmission complete interrupt (in transmit mode, gateway mode)
     - CAN1 transmission history interrupt
     - CAN1 transmit queue Interrupt
  2. CAN1 transmit/receive FIFO receive complete interrupt (in transmit mode, gateway mode)
  3. CAN1 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 RSCAN module to the interrupt controller. (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.

Figure 27.2 shows the CAN global interrupt block diagram. Figure 27.3 shows the CAN channel interrupt block diagram.

Table 27.12 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 RSCAN0RFSTS0 register	RFIE in the RSCAN0RFCC0 register
		Receive FIFO 1	RFIF in the RSCAN0RFSTS1 register	RFIE in the RSCAN0RFCC1 register
		Receive FIFO 2	RFIF in the RSCAN0RFSTS2 register	RFIE in the RSCAN0RFCC2 register
		Receive FIFO 3	RFIF in the RSCAN0RFSTS3 register	RFIE in the RSCAN0RFCC3 register
		Receive FIFO 4	RFIF in the RSCAN0RFSTS4 register	RFIE in the RSCAN0RFCC4 register
		Receive FIFO 5	RFIF in the RSCAN0RFSTS5 register	RFIE in the RSCAN0RFCC5 register
		Receive FIFO 6	RFIF in the RSCAN0RFSTS6 register	RFIE in the RSCAN0RFCC6 register
		Receive FIFO 7	RFIF in the RSCAN0RFSTS7 register	RFIE in the RSCAN0RFCC7 register
	Global error		DEF in the RSCAN0GERFL register MES in the RSCAN0GERFL register THLES in the RSCAN0GERFL register	DEIE in the RSCAN0GCTR register MEIE in the RSCAN0GCTR register THLEIE in the RSCAN0GCTR register
Channel interrupts	CAN1 transmit	CAN1 transmission complete	TMTRF[1:0] in the RSCAN0TMSTSp register	TMIE in the RSCAN0TMIEC0 register
		CAN1 transmission abort	TMTRF[1:0] in the RSCAN0TMSTSp register	TAIE in the RSCAN0C1CTR register
		CAN1 transmit/ receive FIFO transmission complete	CFTXIF in the RSCAN0CFSTSk register	CFTXIE in the RSCAN0CFCCk register
		CAN1 transmit queue	TXQIF in the RSCAN0TXQSTS1 register	TXQIE in the RSCAN0TXQCC1 register
		CAN1 transmission history	THLIF in the RSCAN0THLSTS1 register	THLIE in the RSCAN0THLCC1 register
	CAN1 transmit/receive FIFO receive complete		CFRXIF in the RSCAN0CFSTSk register	CFRXIE in the RSCAN0CFCCk register
	CAN1 error		BEF in the RSCAN0C1ERFL register ALF in the RSCAN0C1ERFL register BLF in the RSCAN0C1ERFL register OVLF in the RSCAN0C1ERFL register BORF in the RSCAN0C1ERFL register BOEF in the RSCAN0C1ERFL register EPF in the RSCAN0C1ERFL register EWF in the RSCAN0C1ERFL register	BEIE in the RSCAN0C1CTR register ALIE in the RSCAN0C1CTR register BLIE in the RSCAN0C1CTR register OLIE in the RSCAN0C1CTR register BORIE in the RSCAN0C1CTR register BOEIE in the RSCAN0C1CTR register EPIE in the RSCAN0C1CTR register EWIE in the RSCAN0C1CTR register

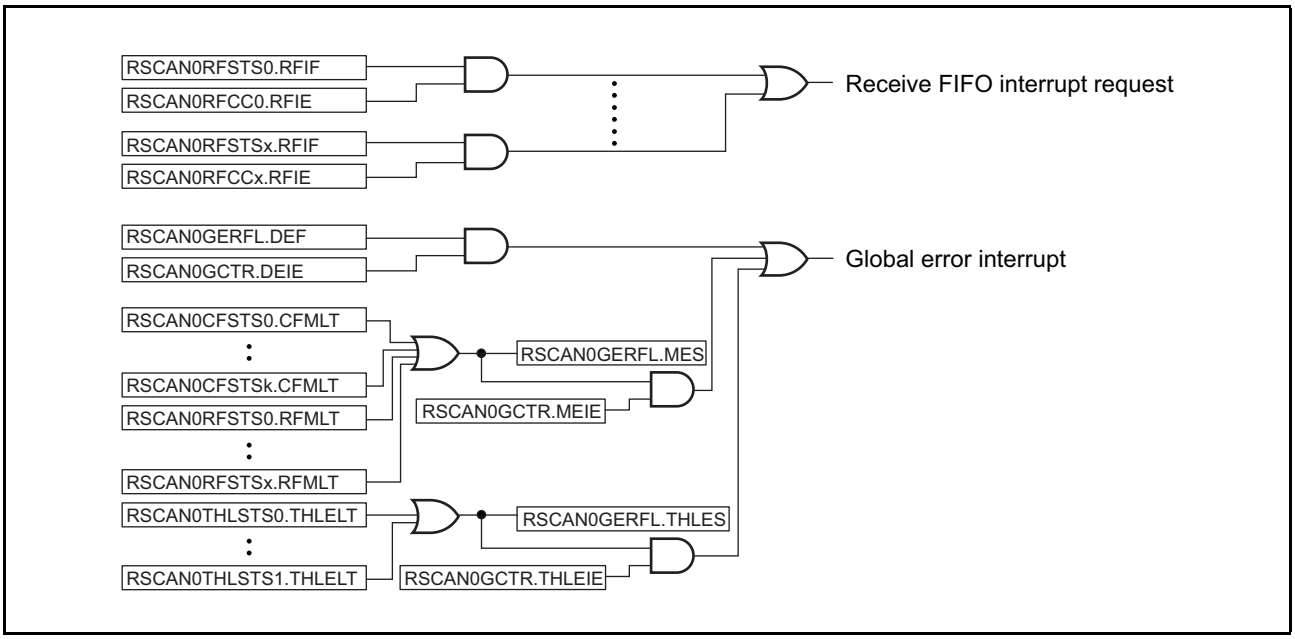


Figure 27.2 CAN Global Interrupt Block Diagram

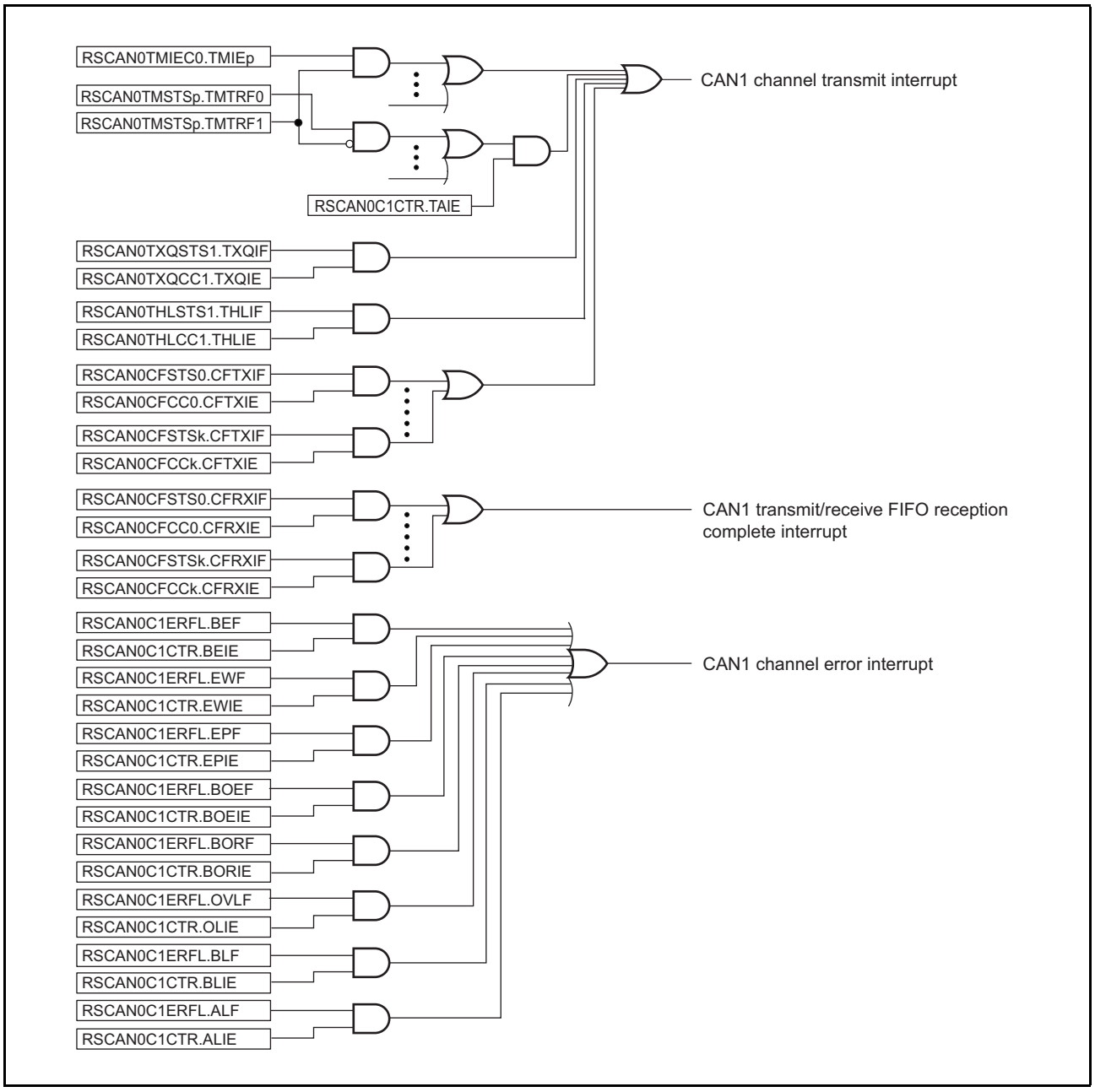


Figure 27.3 CAN Channel Interrupt Block Diagram

27.4 CAN Modes

The RSCAN module has four global modes to control the state of the entire module and four channel modes to control the state of channel 1 (CAN1). Details of global modes are described in section 27.4.1, Global Modes, and details of channel modes are described in section 27.4.2, 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.

27.4.1 Global Modes

Figure 27.4 shows the state transitions of global modes.

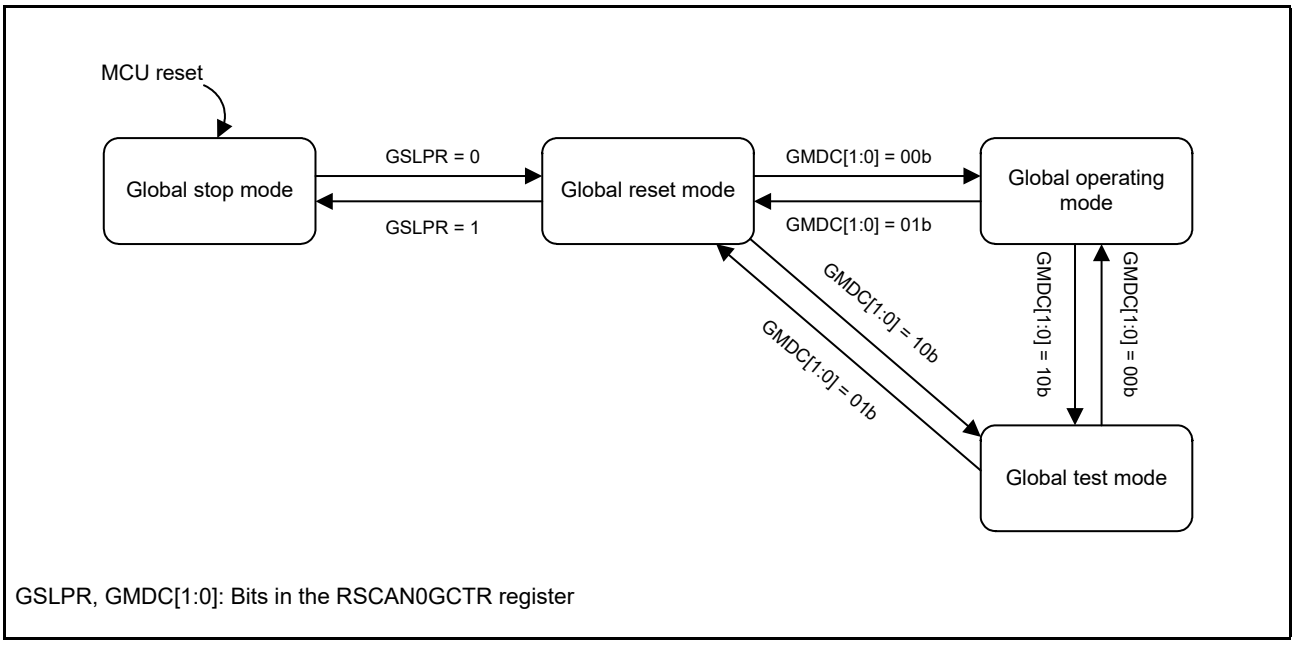


Figure 27.4 State Transitions of Global Modes

In some cases, global mode transitions also force channel mode transitions. Table 27.13 shows the channel mode transitions depending on the global mode setting dictated by the GMDC[1:0] bits and the GSLPR bit.

Table 27.13 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] = 00b GSLPR = 0 (Global Operation)**	**GMDC[1:0] = 10b GSLPR = 0 (Global Test)**	**GMDC[1:0] = 01b GSLPR = 0 (Global Reset)**	**GMDC[1:0] = 01b 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 1. GMDC[1:0], GSLPR: Bits in the RSCAN0GCTR register

Table 27.14 shows the global mode transition time.

Table 27.14 Global Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Global stop	Global reset	Three PCLKD cycles
Global reset	Global stop	Three PCLKD cycles
Global reset	Global test	Ten PCLKD cycles
Global reset	Global operating	Ten PCLKD cycles
Global test	Global reset	Three PCLKD cycles
Global test	Global operating	Three PCLKD cycles
Global operating	Global reset	Three PCLKD cycles
Global operating	Global test	Two CAN1 frames*1

Note 1. CAN1 frame (1 message) time of the lowest communication speed of the channels in use.

27.4.1.1 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 CAN module transitions to global stop mode. Setting the GSLPR bit in the RSCAN0GCTR register to 1 (in global stop mode) in global reset mode sets the CSLPR bit in the RSCAN0C1CTR register to 1 (channel stop mode). If it is forced to transition to channel stop mode, the CAN module transitions to global stop mode. The GSLPR bit should not be modified in global operating mode or global test mode.

27.4.1.2 Global Reset Mode

In global reset mode, RSCAN module settings are performed. When the RSCAN module transitions to global reset mode, some registers are initialized. Table 27.17 and Table 27.18 list the registers to be initialized.

Setting the GMDC[1:0] bits in the RSCAN0GCTR register to 01b sets the CHMDC[1:0] bits in the RSCAN0C1CTR register to 01b (channel reset mode). If it is forced to transition to channel reset mode, the CAN 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 01b).

27.4.1.3 Global Test Mode

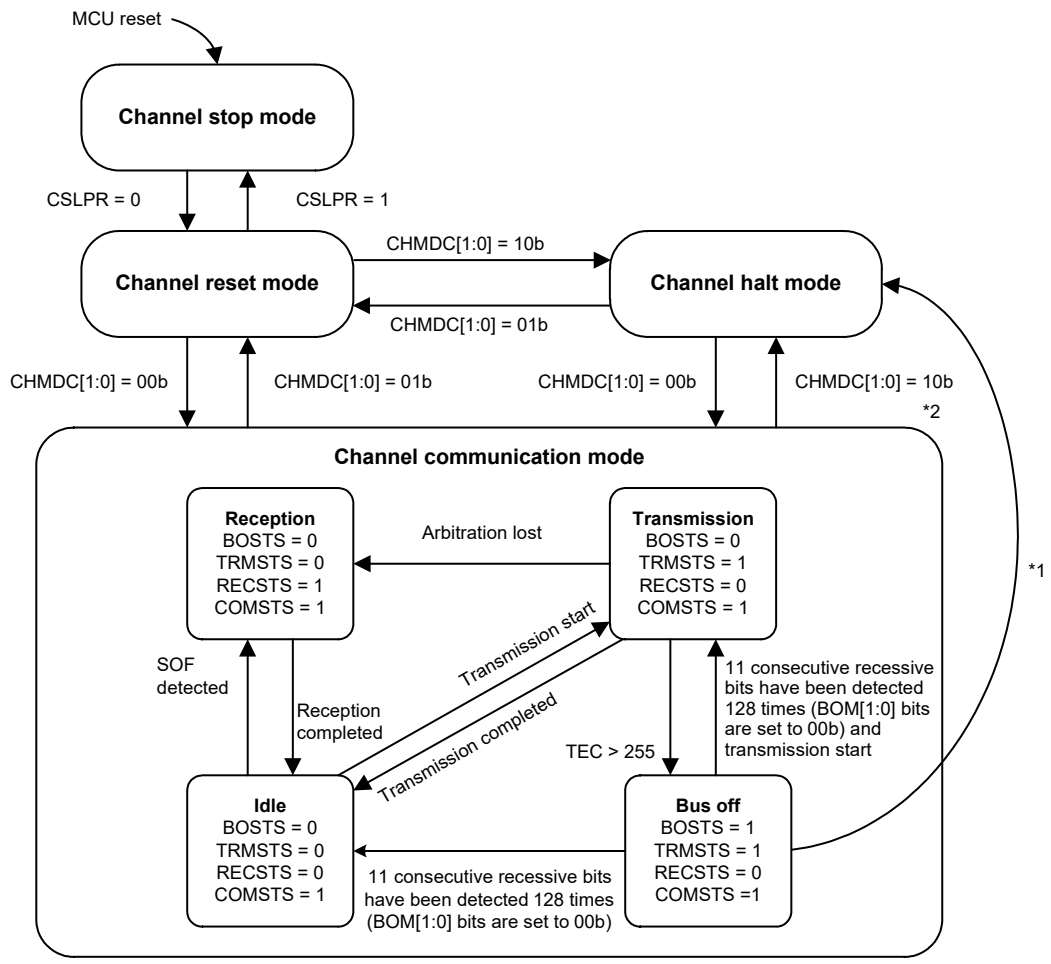
In global test mode, settings for test-related registers are performed. When the CAN module transitions to global test mode, all CAN communications are disabled.
Setting the GMDC[1:0] bits in the RSCAN0GCTR register to 10b sets the CHMDC[1:0] bits in the RSCAN0C1CTR register to 10b (channel halt mode). If it is are forced to transition to channel halt mode, the CAN module transitions to global test mode. Channels that are in channel stop mode, channel reset mode, or channel halt mode do not transition.

27.4.1.4 Global Operating Mode

The RSCAN module operates in global operating mode.
When the GMDC[1:0] bits in the RSCAN0GCTR register are set to 00b, the RSCAN module transitions to global operating mode.

27.4.2 Channel Modes

Figure 27.5 shows a channel mode state transition chart. Table 27.15 shows the channel mode transition time.



CHMDC[1:0], CSLPR, BOM[1:0]: Bits in the RSCAN0C1CTR register
BOSTS, TRMSTS, RECSTS, COMSTS: Bits in the RSCAN0C1STS register

Note 1. Timing of transition from bus off state to channel halt mode
- When BOM[1:0] = 01b: Transition to channel halt mode when TEC exceeds 255
- When BOM[1:0] = 10b: Transition to channel halt mode when 11 consecutive recessive bits have been detected 128 times
- When BOM[1:0] = 11b: Transition to channel halt mode when the CHMDC[1:0] bits are set to 10b

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.

Figure 27.5 Channel Mode State Transition Chart

Table 27.15 Channel Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Channel stop	Channel reset	Three PCLKD cycles
Channel reset	Channel stop	Three PCLKD cycles
Channel reset	Channel halt	Three CAN1 bit times
Channel reset	Channel communication	Two CAN1 bit times
Channel halt	Channel reset	Three PCLKD cycles
Channel halt	Channel communication	Three CAN1 bit times
Channel communication	Channel reset	Three PCLKD cycles
Channel communication	Channel halt	Two CAN1 frames (1 message)

27.4.2.1 Channel Stop Mode

In channel stop mode, clocks are not supplied to channels and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained.
Channel 1 (CAN1) enters channel stop mode after the MCU is reset. Channels also transition to channel stop mode when the GSLPR bit in the RSCAN0C1CTR register is set to 1 (channel stop mode) in channel reset mode. The GSLPR bit should not be modified in channel communication mode and channel halt mode.

27.4.2.2 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. Table 27.17 lists the registers to be initialized.
When the CHMDC[1:0] bits in the RSCAN0C1CTR register are set to 01b (channel reset mode) during CAN communication, communication is terminated before it is completed and the channel transitions to channel reset mode. Table 27.16 shows the operation when the CHMDC[1:0] bits are set to 01b (channel reset mode) during CAN communication.

27.4.2.3 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 27.16 shows operation when the CHMDC[1:0] bits are set to 10b (channel halt mode) during CAN communication.

Table 27.16 Operation a Channel Transitions to Channel Reset Mode/Channel Halt Mode

Mode	During Reception	During Transmission	Bus Off State
Channel reset (CHMDC[1:0] = 01b)	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] = 10b)	Transitions to channel halt mode after reception is completed.*2	Transitions to channel halt mode after transmission is completed.	[When BOM[1:0] = 00b] Transitions to channel halt mode (CHMDC[1:0] = 10b) only after bus off recovery. [When BOM[1:0] = 01b] Transitions to channel halt mode automatically when the condition for transition to bus off state is met. [When BOM[1:0] = 10b] Transitions to channel halt mode automatically after bus off recovery. [When BOM[1:0] = 11b] Transitions to channel halt mode immediately after the CHMDC[1:0] bits are set to 10b before bus off recovery.

Note 1. To allow transition to channel reset mode after communication is completed, set the CHMDC[1:0] bits to 10b 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 01b.
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 RSCAN0C1ERFL register that becomes 1 when dominant lock is detected.
Note 3. When the transition from channel reset mode to channel wait mode is to be made, set the RSCAN0C1CFG register in channel reset mode and then shift to channel wait mode.

27.4.2.4 Channel Communication Mode

In channel communication mode, CAN communication is performed. Channel 1 (CAN1) 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 RSCAN0C1CTR register are set to 00b, the channel transitions to channel communication mode. After that, once 11 consecutive recessive bits have been detected, the COMSTS flag in the RSCAN0C1STS register 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.

27.4.2.5 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 RSCAN0C1CTR register.

- When BOM[1:0] = 00b:
  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 RSCAN0C1STS register are initialized to 00h, the BORF flag in the RSCAN0C1ERFL register is set to 1 (bus off recovery is detected), and a CAN1 error interrupt request (bus off recovery interrupt) is generated. When the CHMDC[1:0] bits in the RSCAN0C1CTR register are set to 10b (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] = 01b:
  When a channel transitions to the bus off state, the CHMDC[1:0] bits are set to 10b and the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00h, but the BORF flag is not set to 1. Also, a bus off recovery interrupt request is not generated.
- When BOM[1:0] = 10b:
  When a channel has transitioned to the bus off state, the CHMDC[1:0] bits are set to 10b. 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 00h, the BORF flag is set to 1, and a CAN1 error interrupt request (bus off recovery interrupt) is generated.
- When BOM[1:0] = 11b:
  When the CHMDC[1:0] bits are set to 10b 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 00h, 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 CAN1 error interrupt request (bus off recovery interrupt) is generated if a CAN module transitions to error active state (by detecting 128 times of 11 consecutive recessive bits) before CHMDC[1:0] bits are set to 10b.

If the RSCAN 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 01b or 10b is made only when the CHMDC[1:0] bits are 00b (channel communication mode).
Furthermore, setting the RTBO bit in the RSCAN0C1CTR 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 CAN 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 00h. Write 1 to the RTBO bit only when the BOM[1:0] value is 00b.
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.

Table 27.17 Registers Initialized in Global Reset Mode or Channel Reset Mode

Register	Bit / Flag
RSCAN0C1CTR register	CTMS[1:0], CTME, CHMDC[1:0]
RSCAN0C1STS register	CHLTSTS, EPSTS, BOSTS, TRMSTS, RECSTS, COMSTS, REC[7:0], TEC[7:0]
RSCAN0C1ERFL register	CRCREG[14:0], ADERR, B0ERR, B1ERR, CERR, AERR, FERR, SERR, ALF, BLF, OVLF, BORF, BOEF, EPF, EWF, BEF
RSCAN0CFCCk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFE
RSCAN0CFSTSk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFMC[7:0], CFFLL, CFEMP, CFMLT, CFRXIF, CFTXIF
RSCAN0CFTISTS register	CFkTXIF
RSCAN0TMCp register	TMOM, TMTAR, TMTR
RSCAN0TMSTSp register	TMTARM, TMTRM, TMTRF[1:0], TMTSTS
RSCAN0TMTRSTS0 register	TMTRSTSp (Bits of channel 1 (CAN1) are initialized in channel reset mode.)
RSCAN0TMTARSTS0 register	TMTARSTSp (Bits of channel 1 (CAN1) are initialized in channel reset mode.)
RSCAN0TMTCSTS0 register	TMTCSTSp (Bits of channel 1 (CAN1) are initialized in channel reset mode.)
RSCAN0TMTASTS0 register	TMTASTSp (Bits of channel 1 (CAN1) are initialized in channel reset mode.)
RSCAN0TXQCC1 register	TXQE
RSCAN0TXQSTS1 register	TXQIF, TXQFLL, TXQEMP
RSCAN0THLCC1 register	THLE
RSCAN0THLSTS1 register	THLMC[4:0], THLIF, THLELT, THLFLL, THLEMP
RSCAN0GTINTSTS0 register	TSIF1, TAIF1, TQIF1, CFTIF1, THIF1

Note 1. Transmit/receive FIFO buffer: k = 3 to 5
Note 2. Transmit buffer number: p = 16 to 31

Table 27.18 Registers Initialized Only in Global Reset Mode

Register	Bit / Flag
RSCAN0GSTS register	GHLTSTS
RSCAN0GERFL register	THLES, MES, DEF
RSCAN0GTSC register	TS[15:0]
RSCAN0RMNDy register	RMNSq
RSCAN0RFCCx register	RFE
RSCAN0RFSTSx register	RFMC[7:0], RFIF, RFMLT, RFFLL, RFEMP
RSCAN0CFCCk register	When transmit/receive FIFO buffer is in receive mode: CFE
RSCAN0CFSTSk register	When transmit/receive FIFO buffer is in receive mode: CFMC[7:0], CFFLL, CFEMP, CFTXIF, CFRXIF, CFMLT
RSCAN0FESTS register	CFkEMP, RFxEMP
RSCAN0FFSTS register	CFkFLL, RFxFLL
RSCAN0FMSTS register	CFkMLT, RFxMLT
RSCAN0RFISTS register	RFxIF
RSCAN0CFRISTS register	CFkRXIF
RSCAN0GTSTCFG register	RTMPS[6:0]
RSCAN0GTSTCTR register	RTME

Note 1. Receive buffer number: q = 16 to 31
Note 2. Receive FIFO buffer number: x = 0 to 7

27.5 Reception Function

There are two reception types.

- Reception by receive buffers:
  Receive buffers from 16 to 31 can be used. 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 and three transmit/receive FIFO buffers are provided. 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.

27.5.1 Data Processing Using the Reception Rule Table

Data processing using the reception 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 64 reception rules can be registered in the entire module. If reception rules are not set, no messages can be received. Figure 27.6 illustrates how reception rules are registered.

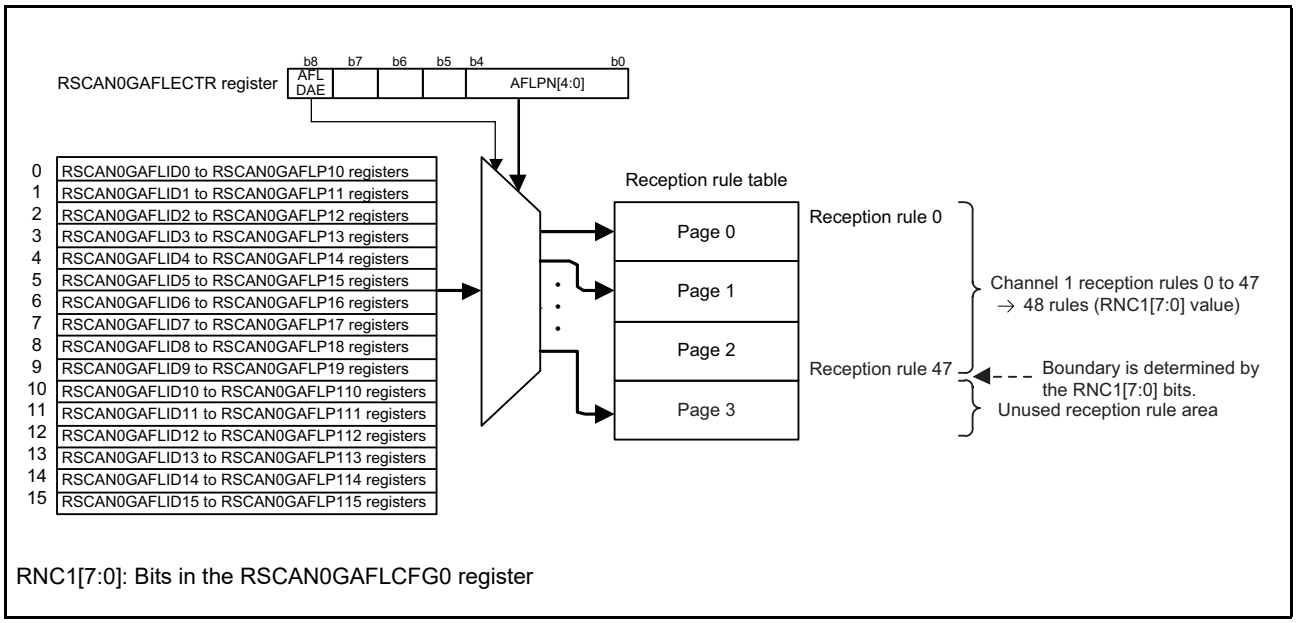


Figure 27.6 Entry of Reception Rules

Note: Reception rules must be set in contiguous blocks.

Each reception rule consists of 16 bytes in the RSCAN0GAFLIDj, RSCAN0GAFLMj, RSCAN0GAFLP0j, and RSCAN0GAFLP1j registers (j = 0 to 15). The RSCAN0GAFLIDj register is used to set GAFLID, GAFLIDE bit, GAFLRTR bit, and the mirror function, the RSCAN0GAFLMj register is used to set mask, the RSCAN0GAFLP0j register is used to set label information to be added, DLC value, and storage receive buffer, and the RSCAN0GAFLP1j register is used to set storage FIFO buffer. Up to 16 reception rules can be set per page.

27.5.1.1 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 reception rule of channel 1 (CAN1). 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 RSCAN0GAFLMj register are not compared and are regarded as matched. Check begins with the reception rule of the minimum number for channel 1 (CAN1). When all the bits to be compared in a received message match the bits set in the reception rule or when all the reception rules are compared without any match, filter processing stops. If there is no matching reception rule, the received message is not stored in the receive buffer or FIFO buffer.

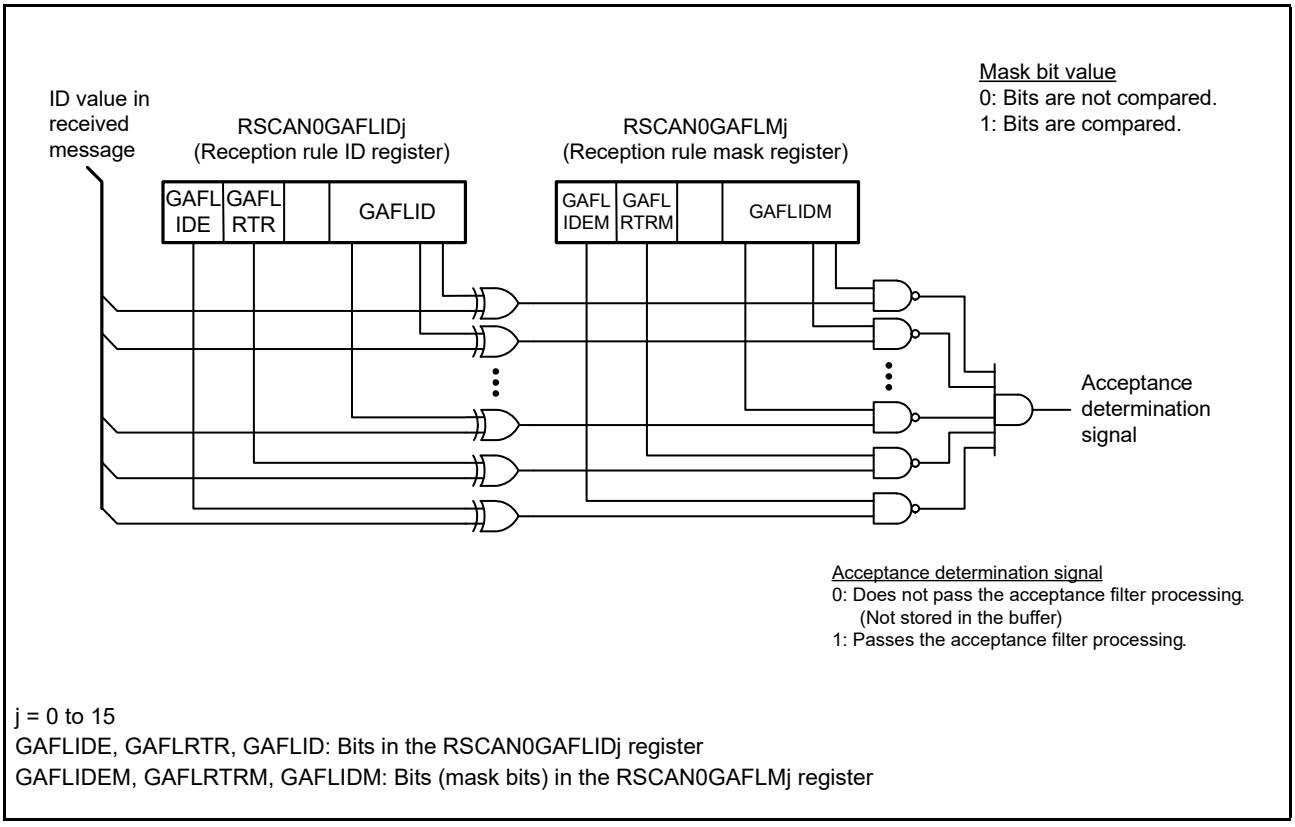


Figure 27.7 Acceptance Filter Function

27.5.1.2 DLC Filter Processing

When the DCE bit in the RSCAN0GCFG 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 reception 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 RSCAN0GCFG 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 RSCAN0GCFG register set to 1 (DLC replacement is enabled), the DLC value in the reception rule is stored in the buffer instead of the DLC value in the received message. In this case, a value of 00h is stored in each data byte beyond the number of bytes which is indicated by the DLC value in the reception rule.
When the DLC value in the received message is smaller than that in the reception 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 RSCAN0GERFL register is set to 1 (a DLC error is present).

27.5.1.3 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 RSCAN0GAFLP0j register (j = 0 to 15) and by the RSCAN0GAFLP1j register. Messages that passed through the acceptance filter processing and the DLC filter processing can be stored in up to eight buffers.

27.5.1.4 Label Addition Processing

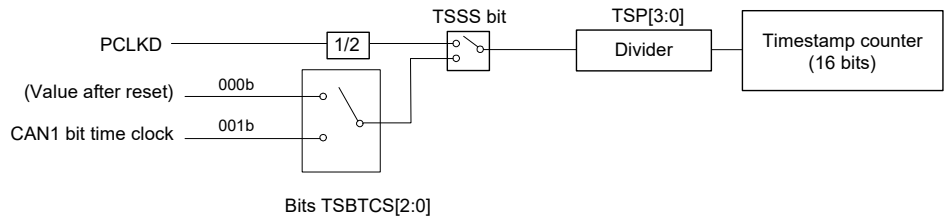
It is possible to add 12-bit label information to messages that passed through the filter processing and store them in buffers. This label information is set in the GAFLPTR[11:0] bits in the RSCAN0GAFLP0j register.

27.5.1.5 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 RSCAN0GCFG register to 1 (mirror function is enabled).
When the mirror function is in use, reception rules for which the GAFLLB bit in the RSCAN0GAFLIDj 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, reception rules for which the GAFLLB bit is set to 1 are used for data processing.

27.5.1.6 Timestamp

The timestamp counter is a 16-bit free-running counter used for recording message receive time. The timestamp counter value is fetched at the start-of-frame (SOF) timing of a message and is then stored in a receive buffer or a FIFO buffer together with the message ID and data. Either PCLKD/2 or the CAN1 bit time clock may be selected as a timestamp counter clock source using the TSBTCS[2:0] and TSSS bits in the RSCAN0GCFG register. The timestamp counter source is obtained by dividing the selected clock source by the TSP[3:0] value in the RSCAN0GCFG register.
When the CAN1 bit time clock is used as a clock source, the timestamp counter stops when channel 1 (CAN1) transitions to channel reset mode or channel halt mode. When the PCLKD/2 is used as a clock source, the timestamp function is not affected by channel mode.
The timestamp counter value is reset to 0000h by setting the TSRST bit in the RSCAN0GCTR register to 1.



TSBTCS[2:0], TSSS, TSP[3:0]: Bits in the RSCAN0GCFG register

Note: When specifying the bit time clock as the timestamp counter source, set bits TSBTCS[2:0] to 001b.

Figure 27.8 Timestamp Function Block Diagram

27.6 Transmission Functions

There are three types of transmission.

- Transmission using transmit buffers:
  Channel 1 has 16 buffers.
- Transmission using transmit/receive FIFO buffers (transmit mode):
  Channel 1 has three FIFO buffers. Up to 64 messages can be contained in a single FIFO buffer. 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 can be allocated to the transmit queues of channel 1. Transmit buffers 16 to 31 are used as an access window of channel 1. 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.

Table 27.19 Transmit Buffers p Allocated to the Channel (p = 16 to 31)

Channel	Allocation of Transmit Buffers
CAN1	Transmit buffers 16 to 31

Table 27.20 Transmit/Receive FIFO Buffers k Allocated to the Channel (k = 3 to 5)

Channel	Allocation of Transmit Buffers
CAN1	Transmit/receive FIFO buffers 3 to 5

Figure 27.9 shows the allocation of transmit queues and transmit/receive FIFO buffer link.

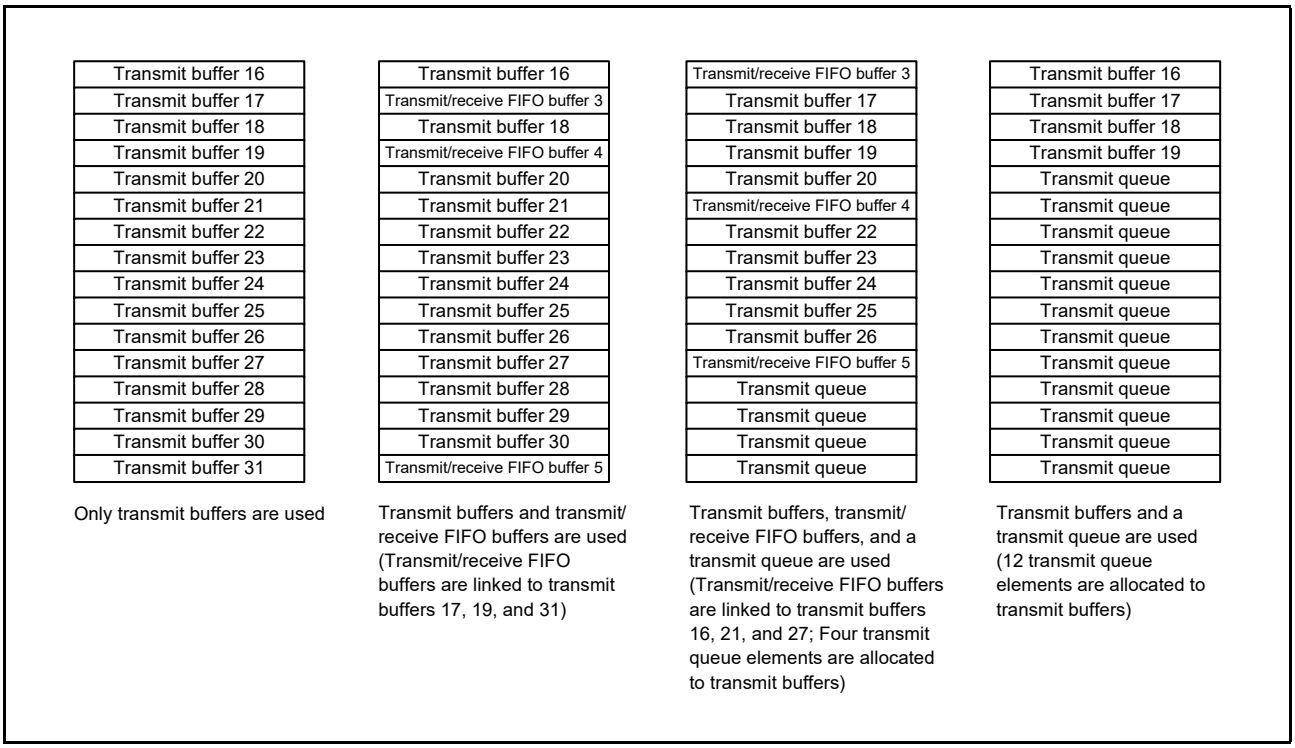


Figure 27.9 Allocation of Transmit Queues and Transmit/Receive FIFO Buffer Links

27.6.1 Transmit Priority Determination

If transmission requests are issued from multiple buffers or from the queue, 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)

The settings of the TPRI bit in the RSCAN0GCFG register are effective for channel 1 (CAN1).
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 transmission 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.

27.6.2 Transmission Using Transmit Buffers

Setting the transmission request bit (TMTR bit in the RSCAN0TMCp register) in a transmit buffer to 1 (transmission is requested) allows transmission of data frames or remote frames.
The transmission result is shown by the TMTRF[1:0] flag in the corresponding RSCAN0TMSTSp register (p = 16 to 31). When transmission completes successfully, the TMTRF[1:0] flag is set to 10b (transmission has been completed (without transmission abort request)) or 11b (transmission has been completed (with transmission abort request)).

27.6.2.1 Transmission Abort Function

With respect to transmit buffers for which the TMTRM bit in the RSCAN0TMSTSp register is set to 1 (a transmission request is present), when the TMTAR bit in the RSCAN0TMCp register is set to 1 (transmission abort is requested), the transmission request is canceled. When transmission abort is completed, the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 01b (transmission abort has been completed) and the transmission 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.

27.6.2.2 One-Shot Transmission Function (Retransmission Disabling Function)

When the TMOM bit in the RSCAN0TMCp 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 transmission result is shown by the TMTRF[1:0] flag in the corresponding RSCAN0TMSTSp register. When one-shot transmission completes successfully, the TMTRF[1:0] flag is set to 10b or 11b. When an arbitration-lost or an error occurs, the TMTRF[1:0] flag is set to 01b (transmission abort has been completed).

27.6.3 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 RSCAN0CFCCk register (k = 3 to 5). 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 RSCAN0CFCCk register. When the CFE bit in the RSCAN0CFCCk 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, or arbitration-lost 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.

27.6.3.1 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 RSCAN0CFCCk 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 RSCAN0CFCCk register. When the interval timer is not used, set the CFITT[7:0] bits to 00h.
Select an interval timer count source using the CFITR and CFITSS bits in the RSCAN0CFCCk register. When the CFITR and CFITSS bits are set to 00b, the count source is obtained by dividing PCLKD/2 by the value of the ITRCP[15:0] bits. When the CFITR and CFITSS bits are set to 10b, the count source is obtained by dividing PCLKD/2 by (the value of the ITRCP[15:0] bits in the RSCAN0GCFG register × 10). When the CFITR and CFITSS bits are set to x1b, the CAN1 bit time clock is used as a count source.
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 = 00b (fPBA is the frequency of PCLKD):

$$\frac{1}{f_{PBA}} \times 2 \times M \times N$$

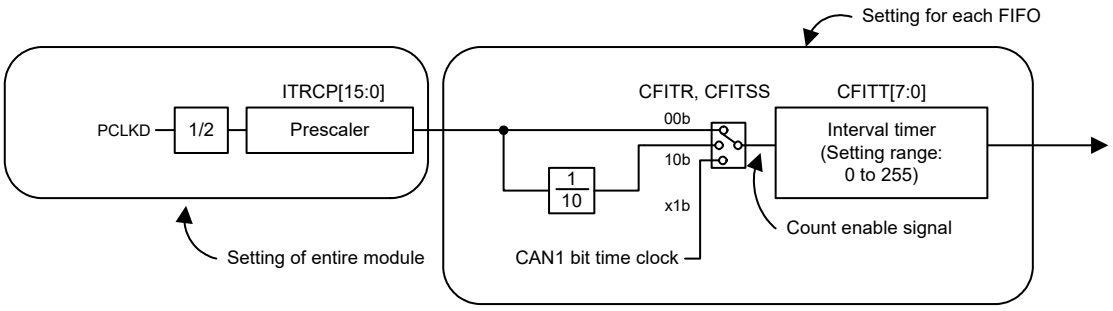
- When CFITR and CFITSS = 10b:

$$\frac{1}{f_{PBA}} \times 2 \times M \times 10 \times N$$

- When CFITR and CFITSS = x1b (fCANBIT is the frequency of CAN1 bit time clock):

$$\frac{1}{f_{CANBIT}} \times N$$

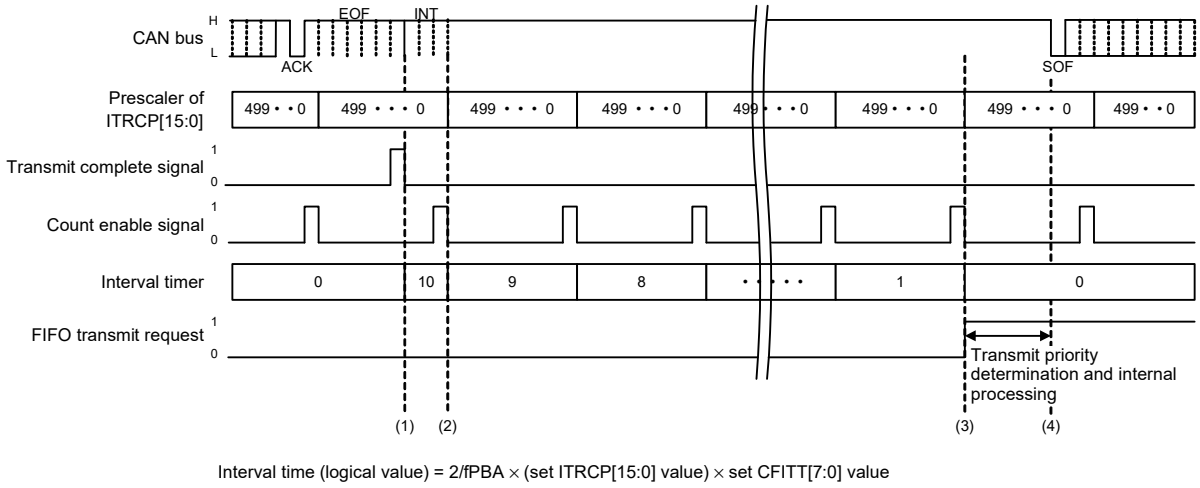
Figure 27.10 shows the interval timer block diagram.



ITRCP[15:0]: Bits in the RSCAN0GCFG register
CFITR, CFITSS, CFITT[7:0]: Bits in the RSCAN0CFCCk register
k = 3 to 5

Figure 27.10 Interval Timer Block Diagram

Figure 27.11 shows the interval timer timing diagram.



Interval time (logical value) = 2/fPBA × (set ITRCP[15:0] value) × set CFITT[7:0] value

fPBA: Frequency of PCLKD
ITRCP[15:0]: Bits in the RSCAN0GCFG register (The set value is 500 in this figure.)
CFITT[7:0]: Bits in the RSCAN0CFCCk register (The set value is 10 in this figure.)

Figure 27.11 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 transmission 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 CAN1 bit time clock cycles or less from the issue of transmission request. If multiple internal processes (such as receive filter processing, message routing, and transmit priority determination) take place, a delay of up to 582 cycles of the PCLKD may be generated.

27.6.4 Transmission Using Transmit Queues

Three to sixteen buffers are allocated to a transmit queue of channel 1 (CAN1), and transmit buffer 31 is used as an access window.
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 RSCAN0TXQCC1 register to 0 disables transmit queues. When the TXQE bit is set to 0, the TXQEMP flag in the RSCAN0TXQSTS1 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, or arbitration-lost 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.

27.6.5 Transmission History Function

Information about transmission-completed messages can be stored in the transmission history buffer. Channel 1 has a single transmission history buffer that can contain 16 sets of transmission history data.
A message transmit source buffer type can be selected by the THLDTE bit in the RSCAN0THLCC1 register. The THLEN bit in the RSCAN0CFIDk register (k = 3 to 5) determines whether transmission history data is stored for each message.
The following information on a transmitted message will be stored in the transmission history buffer after the successful completion of transmission.
Storage of the transmission history data after the successful completion of transmission may take up to 150 cycles of PCLKD.

- Buffer type 001b: Transmit buffer
  010b: Transmit/receive FIFO buffer
  100b: Transmit queue
- Buffer number Number of source transmit buffer, transmit queue, or transmit/receive FIFO buffer.
  This number depends on buffer types. See Table 27.21.
- Label data Label information of the transmit message

Table 27.21 Transmission History Data Buffer Numbers

Buffer No. / Buffer type	001b	010b	100b
0000b	Transmit buffer 16	Buffer numbers of the transmit buffer linked to the transmit/receive FIFO buffer by the CFTML[3:0] bits in the RSCAN0CFCCk register (k = 3 to 5)	Buffer numbers of the transmit buffer allocated to the transmit queue that performed transmission
0001b	Transmit buffer 17		
0010b	Transmit buffer 18		
0011b	Transmit buffer 19		
0100b	Transmit buffer 20		
0101b	Transmit buffer 21		
0110b	Transmit buffer 22		
0111b	Transmit buffer 23		
1000b	Transmit buffer 24		
1001b	Transmit buffer 25		
1010b	Transmit buffer 26		
1011b	Transmit buffer 27		
1100b	Transmit buffer 28		
1101b	Transmit buffer 29		
1110b	Transmit buffer 30		
1111b	Transmit buffer 31		

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.
Transmission history data can be read from the RSCAN0THLACC1 register. If an attempt is made to store new transmission history data while the buffer is full, the buffer overflows and the new data is discarded.

27.7 Gateway Function

When a transmit/receive FIFO buffer is set to gateway mode, received messages can be transmitted from channel 1 (CAN1) without CPU intervention.
When a transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCAN0CFCCk register are set to 10b (gateway mode) is selected by the RSCAN0GAFLP1j register (j = 0 to 15), messages that passed through the filter processing of the reception 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 RSCAN0CFCCk 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.

27.8 Test Function

The test function is classified into communication tests and global tests.

- Communication tests: Tests performed by channel 1 (CAN1).
  - Standard test mode
  - Listen-only mode
  - Self-test mode 0 (external loopback mode)
  - Self-test mode 1 (internal loopback mode)
- Global tests: Performed for the entire module
  - RAM test (read/write test)

27.8.1 Standard Test Mode

Standard test mode allows CRC test.

27.8.2 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 transmission request from any buffer or queue in listen-only mode.
Figure 27.12 shows the connection when listen-only mode is selected.

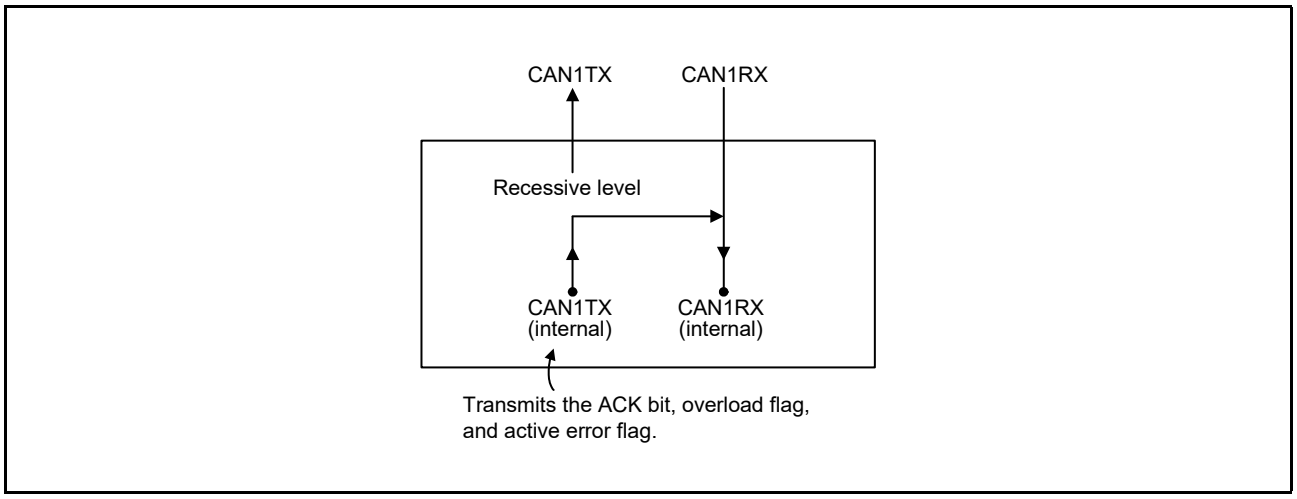


Figure 27.12 Connection when Listen-Only Mode is Selected

27.8.3 Self-Test Mode (Loopback Mode)

In self-test mode, transmitted messages are compared with the reception 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 reception rule for which the GAFLLB bit in the RSCAN0GAFLIDj 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.

27.8.3.1 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 27.13 shows the connection when self-test mode 0 is selected.

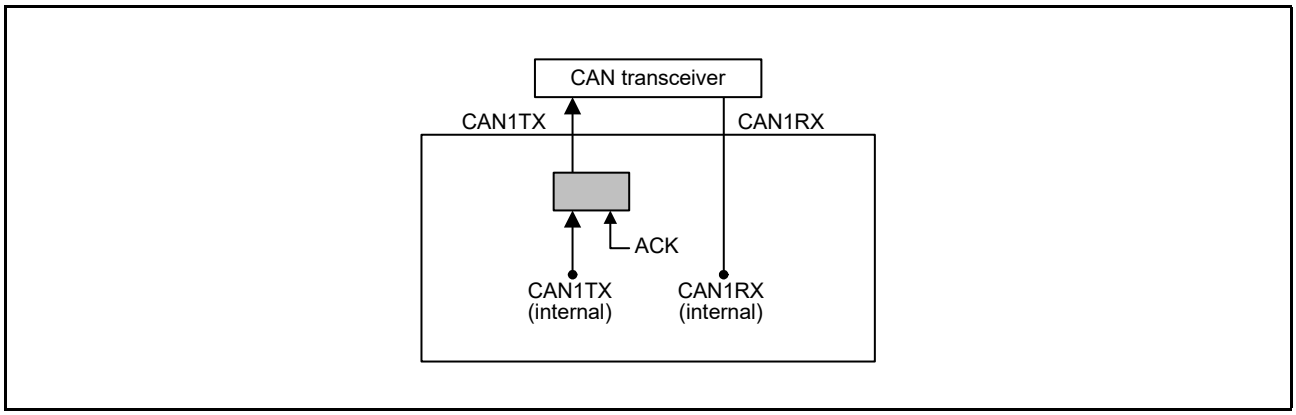


Figure 27.13 Connection when Self-Test Mode 0 is Selected

27.8.3.2 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 CAN1TX pin to the internal CAN1RX pin is performed. The external CAN1RX pin input is isolated. The external CAN1TX pin outputs only recessive bits.
Figure 27.14 shows the connection when self-test mode 1 is selected.

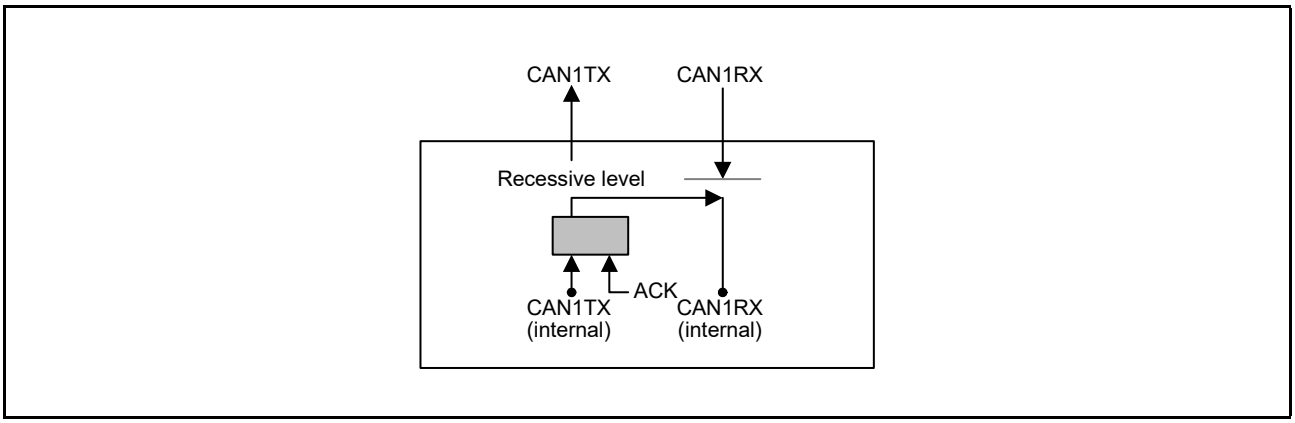


Figure 27.14 Connection when Self-Test Mode 1 is Selected

27.8.4 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 RSCAN0GTSTCFG register. Data in the set page can be read from and written to the RSCAN0RPGACCr register (r = 0 to 63). The available total RAM size is 14592 bytes (3900h).

27.9 RSCAN Setting Procedure

27.9.1 Initial Settings

The RSCAN module initializes the CAN RAM after the MCU is reset. The RAM initialization time is 7298 cycles of PCLKD. The GRAMINIT flag in the RSCAN0GSTS 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 27.15 shows the CAN setting procedure after the MCU is reset.

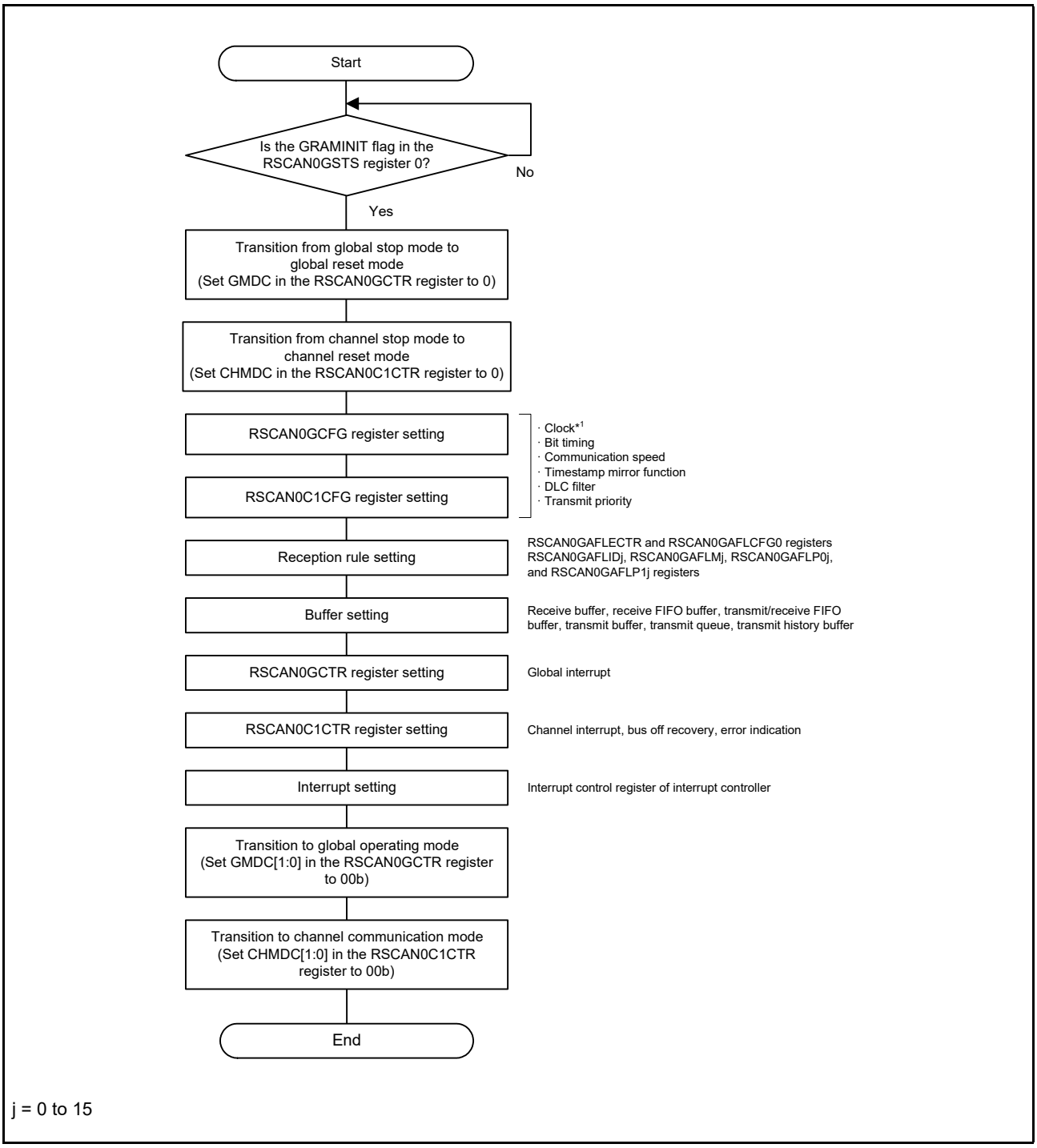


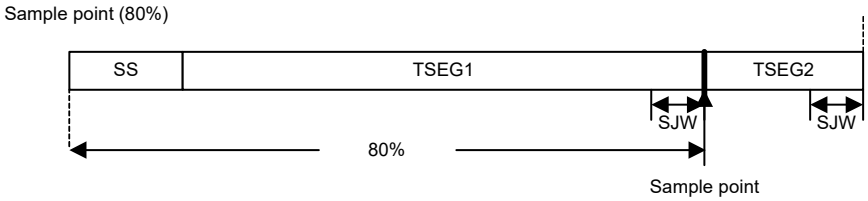
Figure 27.15 CAN Setting Procedure after the MCU is Reset

27.9.1.1 Clock Setting

Set the CAN clock (fCAN) as a clock source of the RSCAN module. Select CANCLKA (24 MHz) or CANCLKB (25 MHz) using the DCS bit in the RSCAN0GCFG register.

27.9.1.2 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 RSCAN0C1CFG 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). 1 Tq is equal to one CAN1Tq clock cycle. The CAN1Tq clock is obtained by selecting the clock source with the DCS bit in the RSCAN0GCFG register and selecting the clock division ratio with the BRP[9:0] bits in the RSCAN0C1CFG register. Figure 27.16 shows the bit timing chart. Table 27.22 shows an example of bit timing setting.



SS = 1 Tq fixed
Set TSEG1 to a range of 4 Tq to 16 Tq
Set TSEG2 to a range of 2 Tq to 8 Tq
Set SJW to a range of 1 Tq to 4 Tq
Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 25 Tq
TSEG1 > TSEG2 > SJW

- SS (synchronization segment):
  The SS is a segment that performs synchronization by monitoring the edge from recessive to dominant bits in the Interframe Space. Interframe Space consists of Intermission, Suspend Transmission, and Bus Idle. All nodes can start transmission during Bus Idle.
- TSEG1 (Time segment 1):
  TSEG1 is a segment that absorbs physical delay on the CAN network. The physical delay on the CAN network is twice the total of the delay on the CAN bus, the delay in the input comparator, and the delay in the output driver.
- TSEG2 (Time segment 2):
  TSEG2 is a segment that compensates phase error due to an error in frequency.
- SJW (Resynchronization jump width):
  The SJW is a length to extend or reduce the time segment to compensate for an error in phase due to phase error.

Figure 27.16 Bit Timing Chart

Table 27.22 Example of Bit Timing Setting

1 Bit	Set Value (Tq) SS	TSEG1	TSEG2	SJW	Sample Point (%) Note: See Figure 27.16.
8 Tq	1	4	3	1	62.50
	1	5	2	1	75.00
10 Tq	1	6	3	1	70.00
	1	7	2	1	80.00
12 Tq	1	8	3	1	75.00
	1	9	2	1	83.33
16 Tq	1	10	5	1	68.75
	1	11	4	1	75.00
20 Tq	1	12	7	1	65.00
	1	13	6	1	70.00
24 Tq	1	15	8	1	66.67
	1	16	7	1	70.83
25 Tq	1	16	8	1	68.00

27.9.1.3 Communication Speed Setting

Set the CAN communication speed using the fCAN, baud rate prescaler division value (BRP[9:0] bits in the RSCAN0C1CFG register), and Tq count per bit time.
Figure 27.17 shows the CAN clock control block diagram, and Table 27.23 shows an example of the communication speed setting.

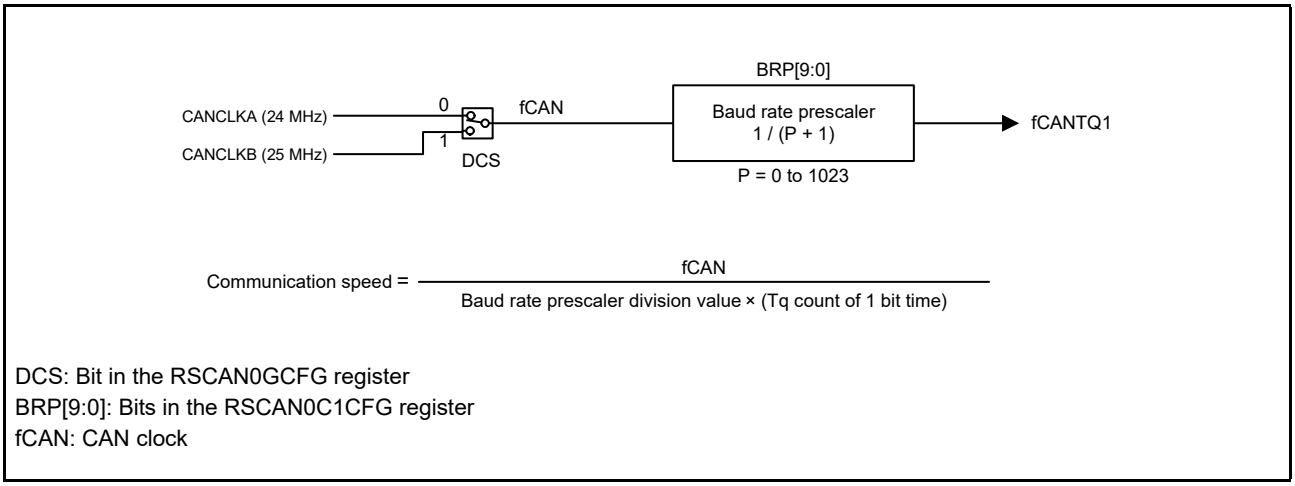


Figure 27.17 CAN Clock Control Block Diagram

Table 27.23 Example of Communication Speed Setting

Communication Speed \ fCAN	25 MHz	24 MHz
1 Mbps	25 Tq (1)	8 Tq (3) 12 Tq (2) 24 Tq (1)
500 kbps	25 Tq (2) 10 Tq (5)	8 Tq (6) 12 Tq (4) 24 Tq (2)
250 kbps	25 Tq (4) 10 Tq (10)	8 Tq (12) 12 Tq (8) 24 Tq (4)
125 kbps	25 Tq (8) 10 Tq (20)	8 Tq (24) 12 Tq (16) 24 Tq (8)

Note: Values in ( ) are baud rate prescaler division values.

27.9.1.4 Reception Rule Setting

Reception rules can be set using reception rule-related registers.
Up to 16 reception rules can be registered per page. Specify pages 0 to 3 by the AFLPN[4:0] bits in the RSCAN0GAFLECTR register. Enable or disable writing to the reception rule table using the AFLDAE bit.
Figure 27.18 shows the reception rule setting procedure.

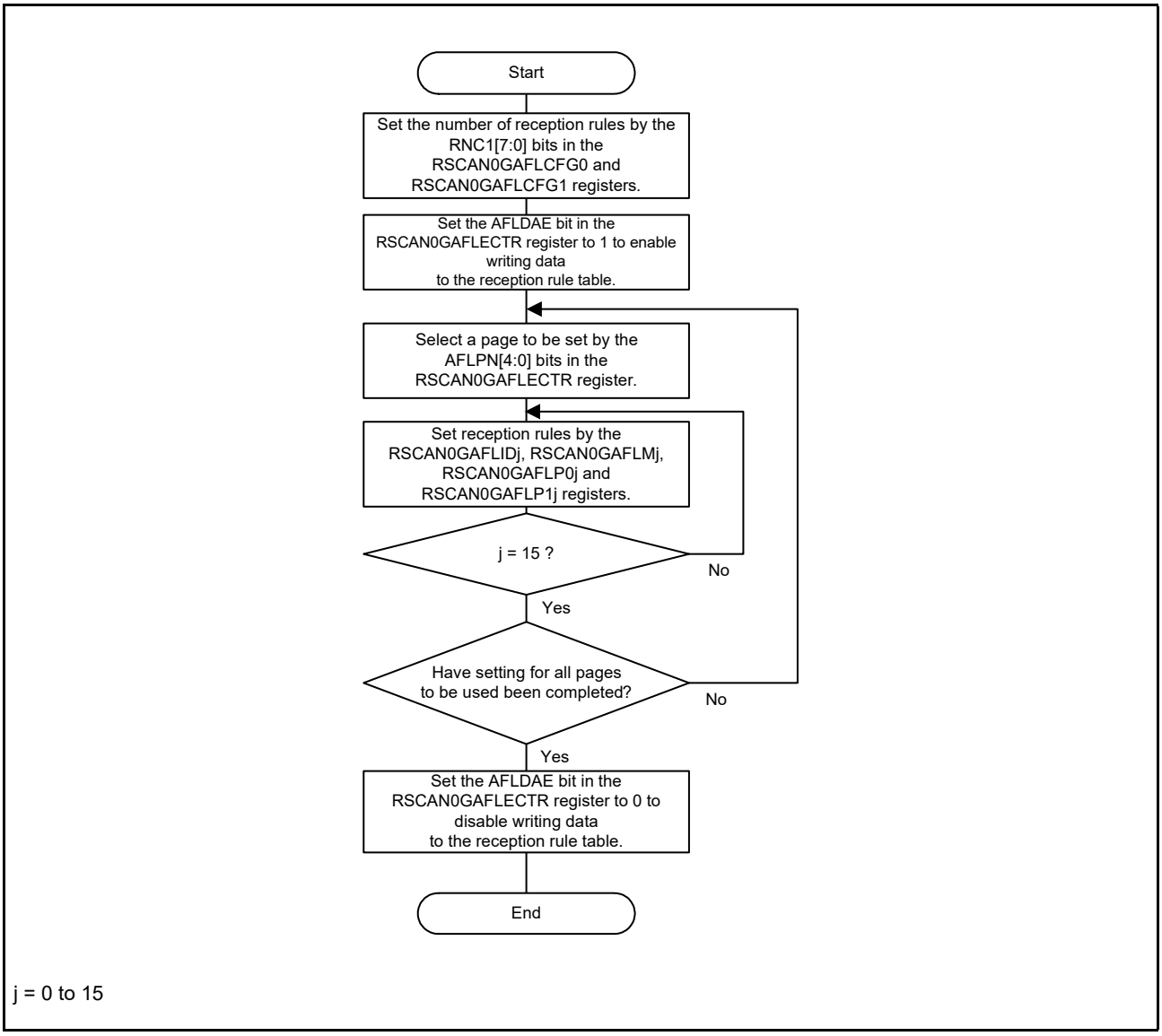


Figure 27.18 Reception Rule Setting Procedure

27.9.1.5 Buffer Setting

Set sizes and interrupt sources of buffers. For transmit/receive FIFO buffers that are set to transmit mode, set transmit buffers to be linked.

Figure 27.19 shows the buffer configuration. Figure 27.20 shows the buffer setting procedure.

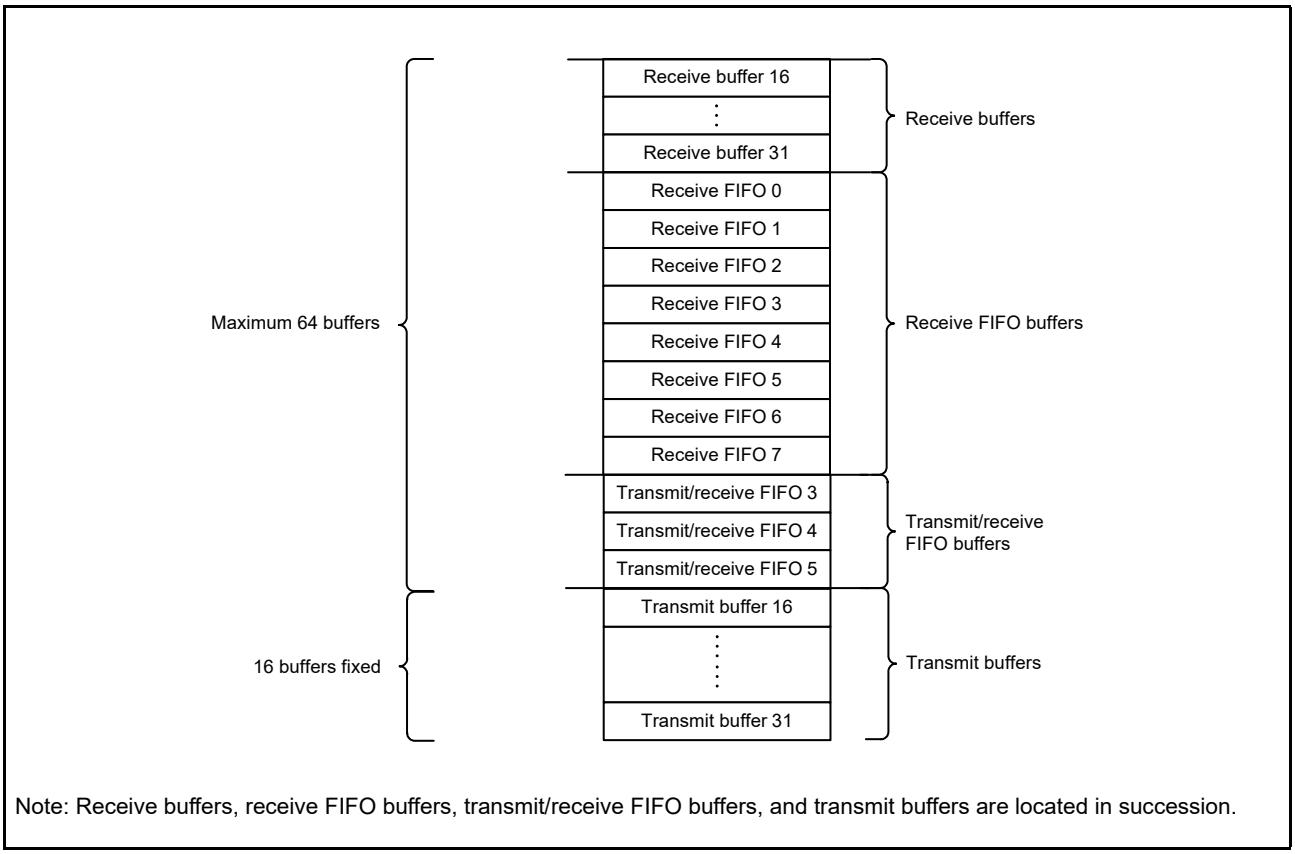


Figure 27.19 Buffer Configuration

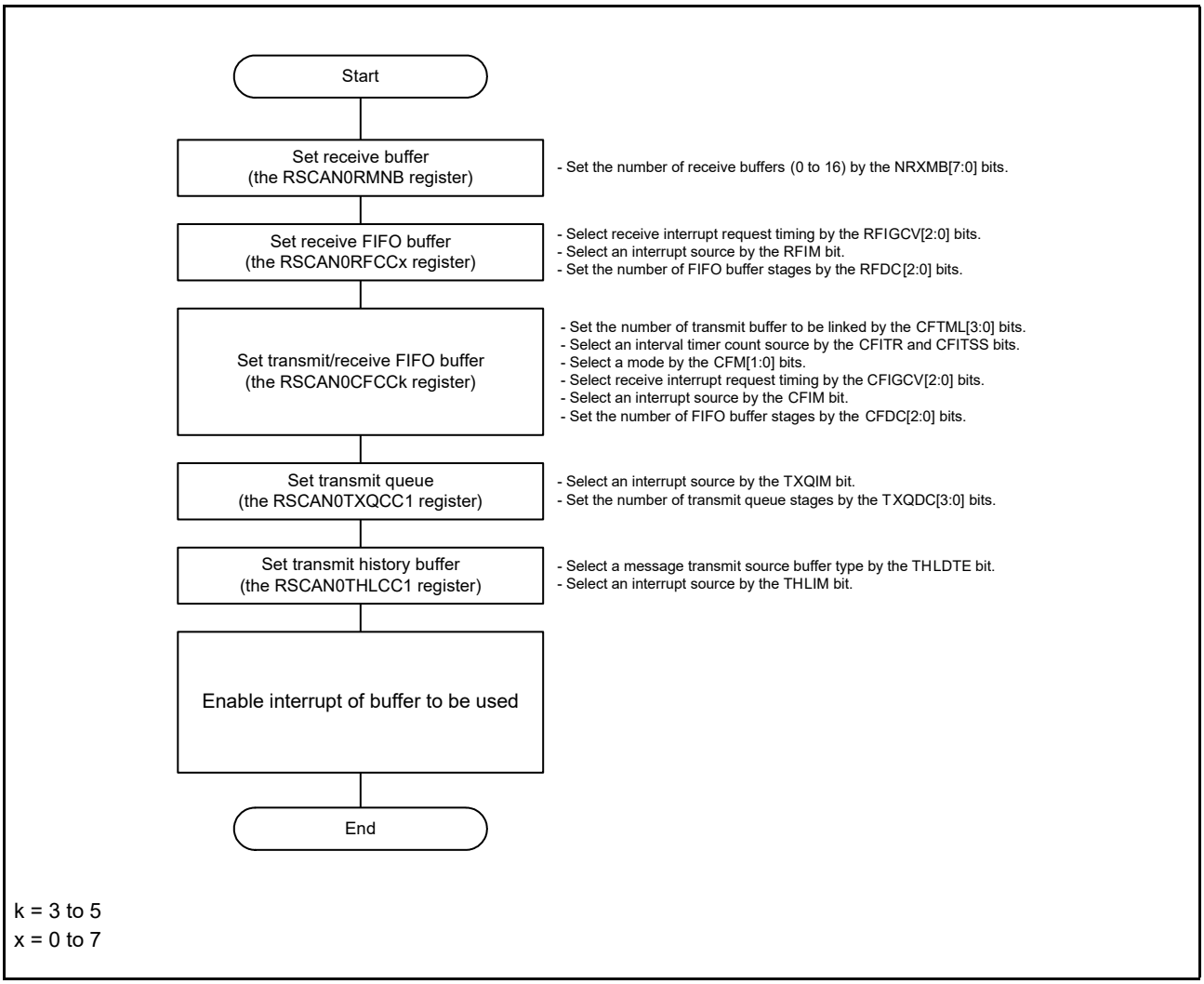


Figure 27.20 Buffer Setting Procedure

27.9.2 Reception Procedure

27.9.2.1 Receive Buffer Reading Procedure

When the processing to store received messages in a receive buffer starts, the RMNSq flag in the RSCAN0RMND0 register (q = 16 to 31) is set to 1 (receive buffer q contains a new message). Messages can be read from the RSCAN0RMIDq, RSCAN0RMPTRq, RSCAN0RMDF0q, and RSCAN0RMDF1q registers. If the next message has been received before the current message is read from the receive buffer, the message is overwritten. Figure 27.21 shows the receive buffer reading procedure.

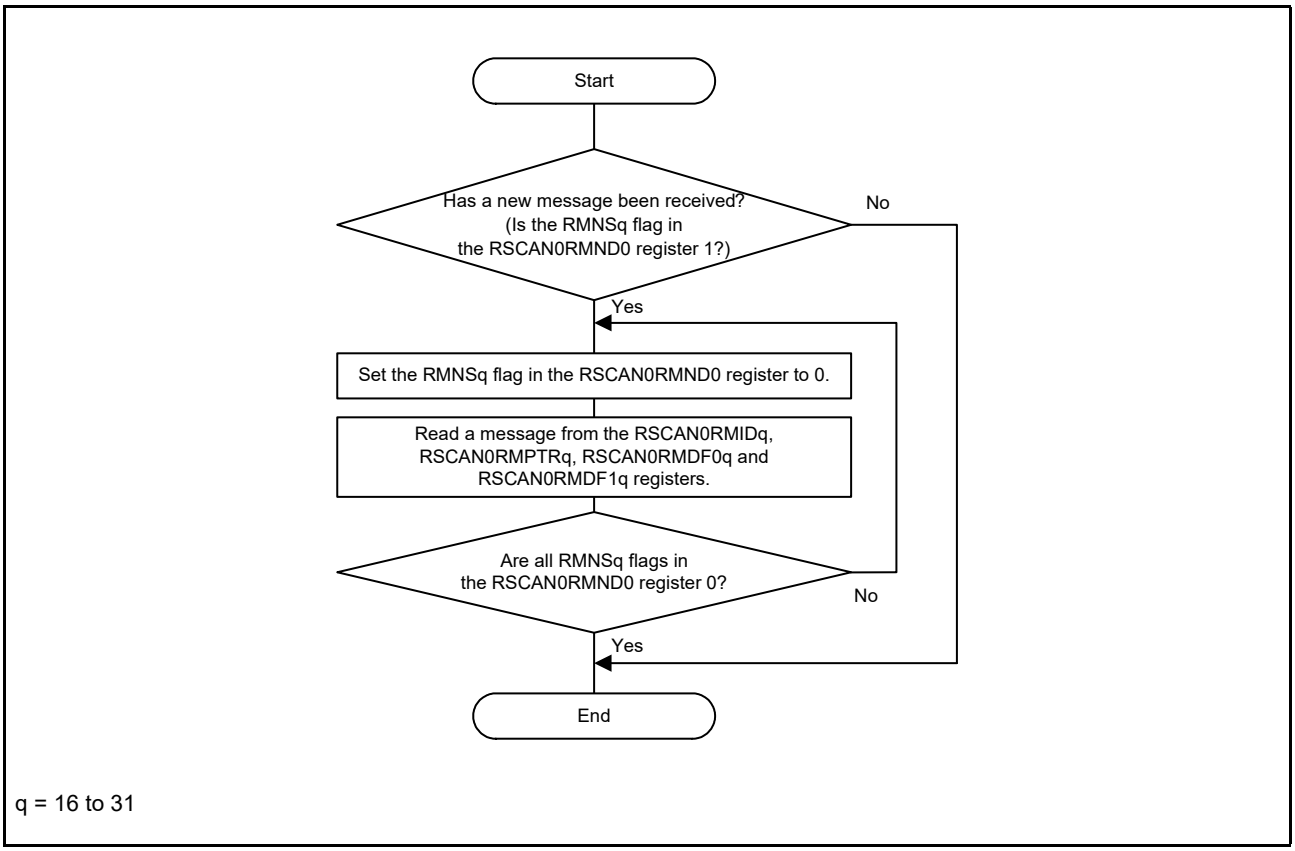
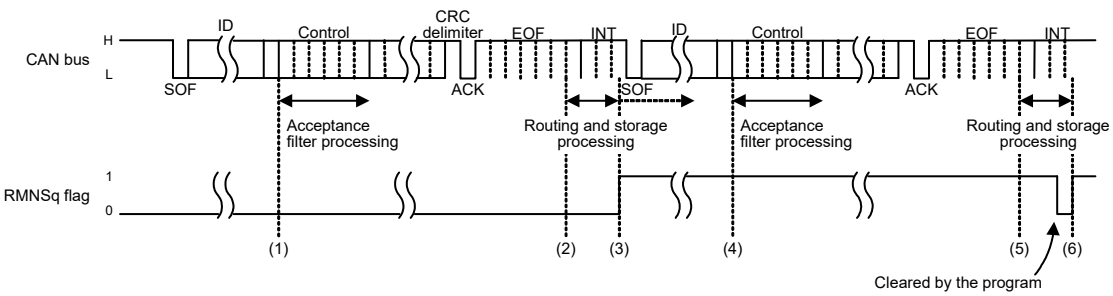


Figure 27.21 Receive Buffer Reading Procedure



q = 16 to 31
RMNSq: Flag in the RSCAN0RMND0 register

Figure 27.22 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 reception rule of channel 1 (CAN1) 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 RSCAN0GCFG 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 RSCAN0RMND0 register is set to 1 (the receive buffer contains a new message).

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) When the message matches the reception rule of channel 1 (CAN1) 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 RSCAN0GCFG 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.

27.9.2.2 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 RSCAN0RFSTSx register (x = 0 to 7) or CFMC[7:0] bits in the RSCAN0CFSTSk register (k = 3 to 5)) is incremented. At this time, when the RFIE bit (receive FIFO interrupt is enabled) in the RSCAN0RFCCx register or the CFRXIE bit (transmit/receive FIFO receive interrupt is enabled) in the RSCAN0CFCCk register is set to 1, an interrupt request is generated. Received messages can be read from the RSCAN0RFIDx, RSCAN0RFPTRx, RSCAN0RFDF0x, and RSCAN0RFDF1x registers for receive FIFO buffers, or from the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k 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 RSCAN0RFCCx register or the CFDC[2:0] bits in the RSCAN0CFCCk 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 RSCAN0RFSTSx register or the CFEMP flag in the RSCAN0CFSTSk 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 RSCAN0RFSTSx register or CFRXIF flag in the RSCAN0CFSTSk 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.

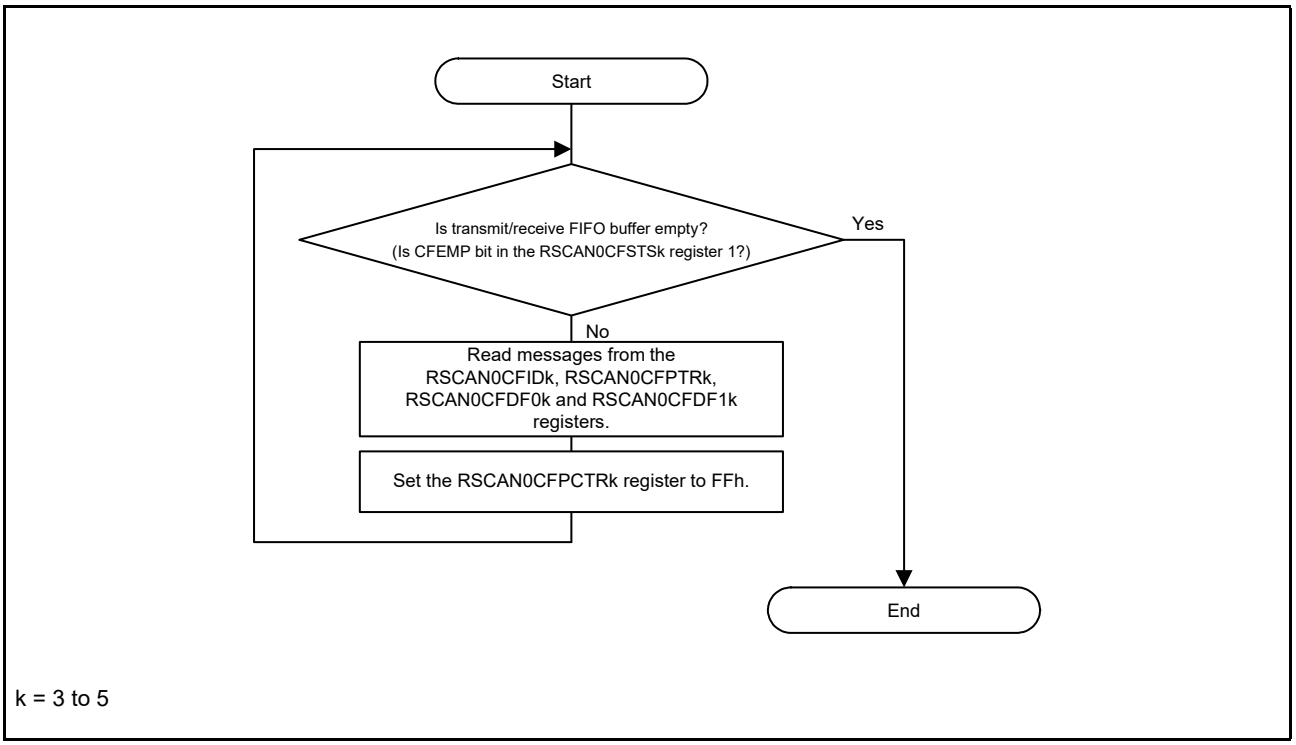
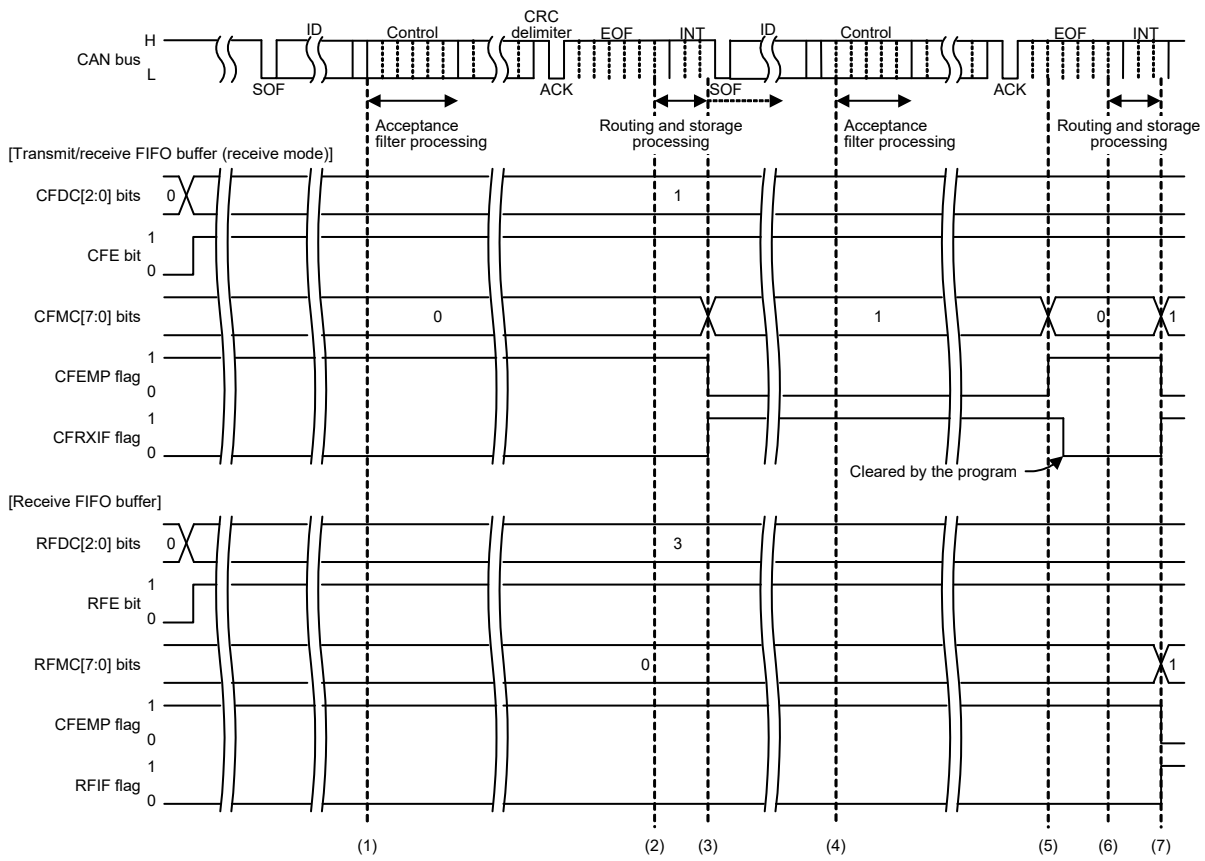


Figure 27.23 Transmit/Receive FIFO Buffer Reading Procedure



k = 3 to 5, x = 0 to 7
CFDC[2:0], CFE: Bits in the RSCAN0CFCCk register
CFMC[7:0], CFEMP, CFRXIF: Flags in the RSCAN0CFSTSk register
RFDC[2:0], RFE: Bits in the RSCAN0RFCCx register
RFMC[7:0], CFEMP, RFIF: Flags in the RSCAN0RFSTSx register

Figure 27.24 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 reception rule of channel 1 (CAN1) 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 RSCAN0GCFG 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 RSCAN0CFCCk register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCCk register is 001b 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 RSCAN0CFSTSk register is incremented and becomes 01h. When the CFIM bit in the RSCAN0CFCCk register is set to 1 (a FIFO receive interrupt request is generated each time a message has been received), the CFRXIF flag in the RSCAN0CFSTSk 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 RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers and write FFh to the RSCAN0CFPCTRk register. This causes the CFMC[7:0] bits in the RSCAN0CFSTSk register to be decremented. When CFMC[7:0] becomes 00h, the CFEMP flag in the RSCAN0CFSTSk register becomes 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

(6) When the message matches the reception rule of channel 1 (CAN1) 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 RSCAN0GCFG 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 001b or more. The CFMC[7:0] bit value is incremented by 1 to be 01h. 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 RSCAN0RFCCx register is set to 1 (receive FIFO buffers are used), and the RFDC[2:0] bits in the RSCAN0RFCCx register are set to 001b or more. The RFMC[7:0] bits in the RSCAN0RFSTSx register are set to 01h by being incremented by 1. When the RFIM bit in the RSCAN0RFCCx register is set to 1 (an interrupt occurs each time a message has been received), the RFIF flag in the RSCAN0RFSTSx register is set to 1 (a receive FIFO interrupt request is present).

27.9.3 Transmission Procedure

27.9.3.1 Procedure for Transmission from Transmit Buffers

Figure 27.25 shows the procedure for transmission from transmit buffers.
Figure 27.26 shows a timing chart where messages are transmitted from two transmit buffers and transmission has been successfully completed. Figure 27.27 shows a timing chart where messages are transmitted from two transmit buffers and transmission abort has been completed.

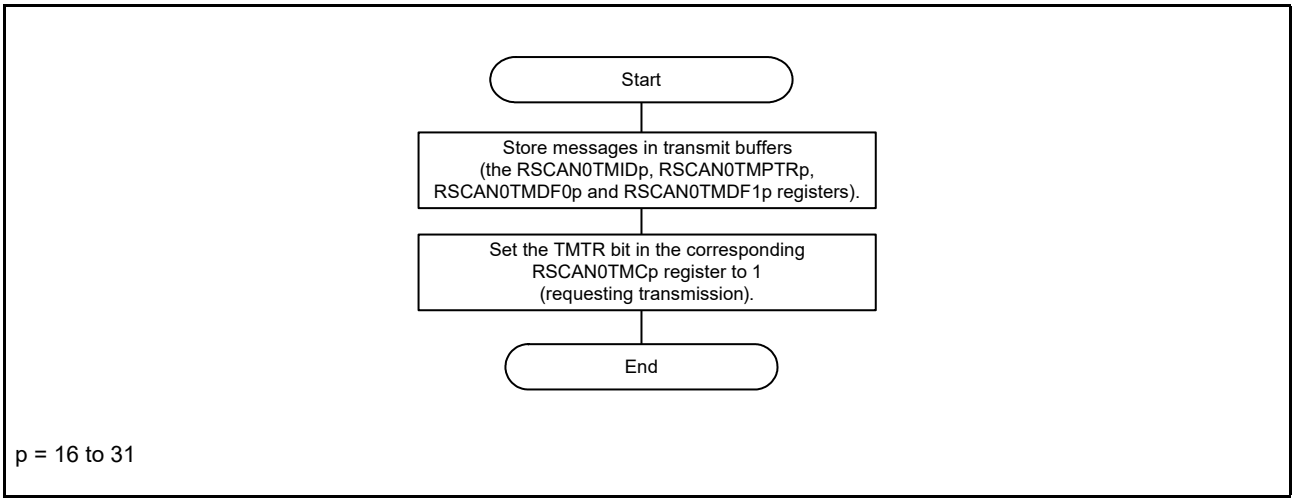
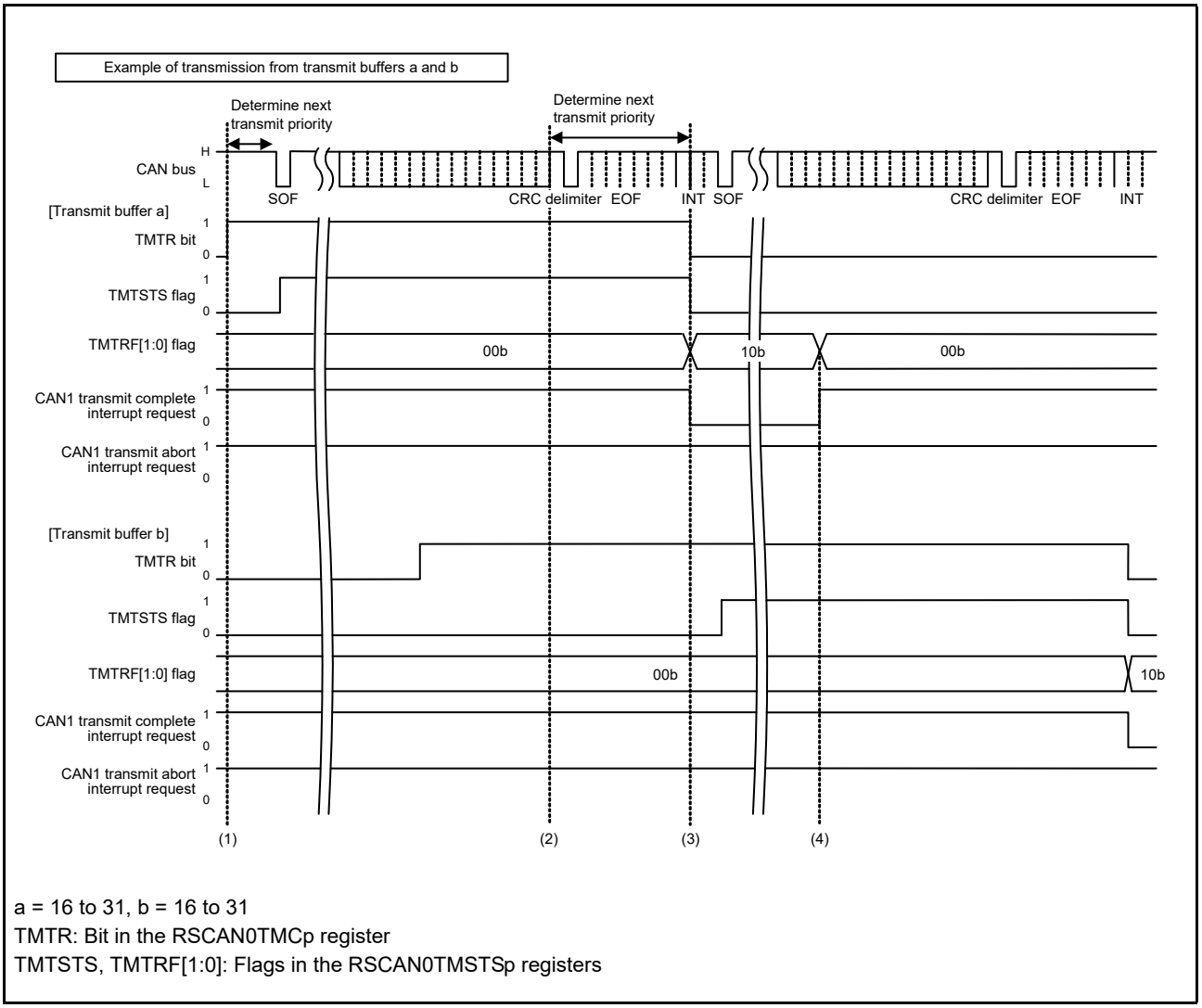


Figure 27.25 Procedure for Transmission from Transmit Buffers



a = 16 to 31, b = 16 to 31
TMTR: Bit in the RSCAN0TMCp register
TMTSTS, TMTRF[1:0]: Flags in the RSCAN0TMSTSp registers

Figure 27.26 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully)

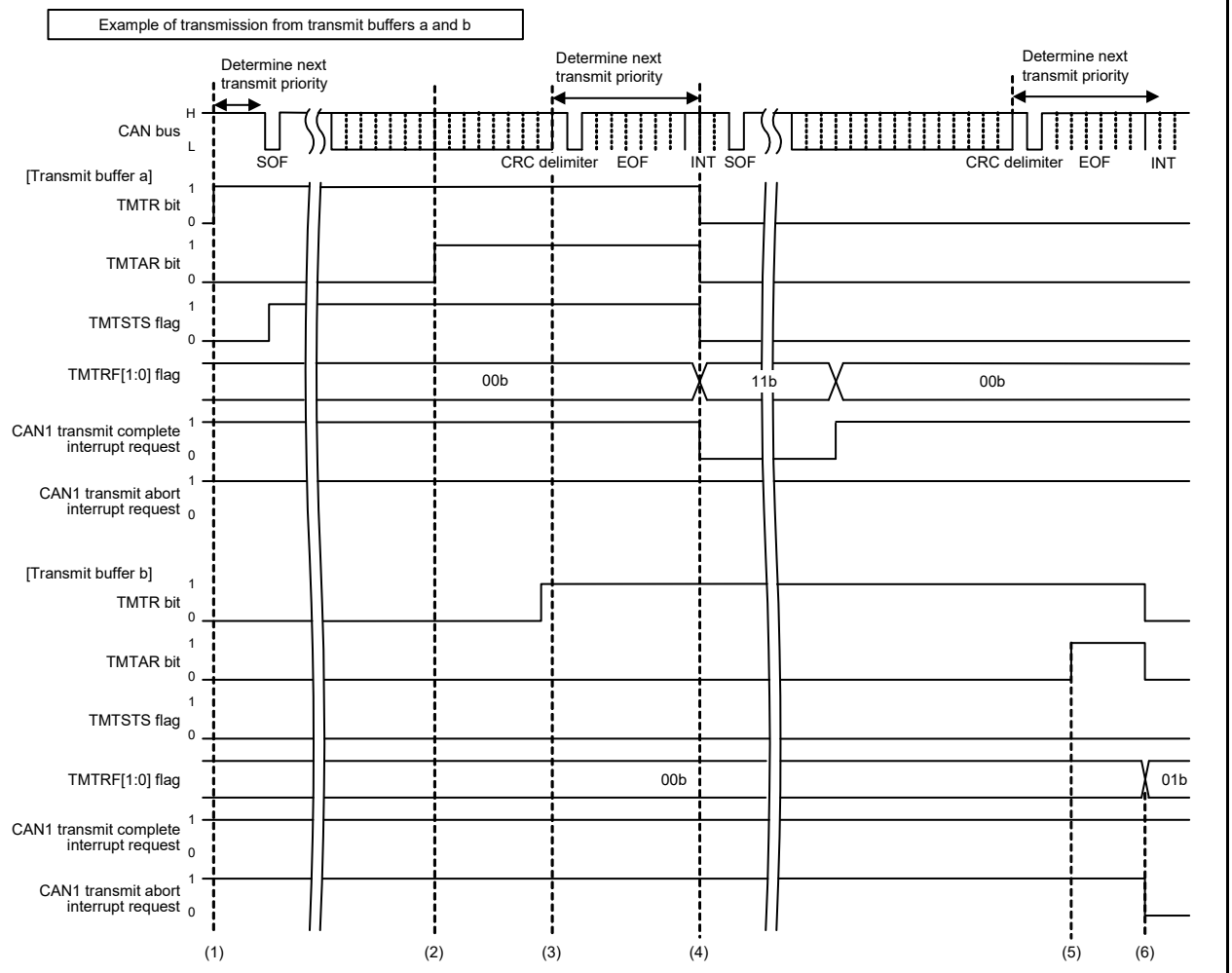
(1) When the TMTR bit in the RSCAN0TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCAN0TMSTSa register is set to 1 (transmission is in progress) and channel 1 (CAN1) starts to transmit data.

(2) When a transmission request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission.

(3) When transmission completes successfully, the TMTRF[1:0] flag in the RSCAN0TMSTSa register is set to 10b (transmission has been completed (without transmission abort request)) and the TMTSTS flag and the TMTR bit in the RSCAN0TMCa register are cleared to 0. When the TMIEa bit in the RSCAN0TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmission complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00b (transmission is in progress or no transmission request is present).

(4) Before starting the next transmission, set the TMTRF[1:0] flag to 00b. 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 00b.

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 delimiter 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.



a = 0 to 31, b = 0 to 31
TMTR, TMTAR: Bits in the RSCAN0TMCp register
TMTSTS, TMTRF[1:0]: Flags in the RSCAN0TMSTSp register

Figure 27.27 Transmit Buffer Transmission Timing Chart (Transmission Abort Completed)

(1) When the TMTR bit in the RSCAN0TMCa 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 RSCAN0TMSTSa register is set to 1 (transmission is in progress) and channel 1 (CAN1) starts to transmit 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 (transmission abort is requested).

(3) The priority determination starts with the CRC delimiter for the next transmission. In this timing chart, buffer b is not selected as the next transmit buffer.
(4) When transmission completes successfully, the TMTRF[1:0] flag in the RSCAN0TMSTSa register is set to 11b (transmission has been completed (with transmission abort request)) and the TMTSTS flag and the TMTR bit in the RSCAN0TMCa register are cleared to 0. When the TMIEa value in the RSCAN0TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmission complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00b (transmission is in progress or no transmission 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 channel 1 (CAN1) 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 01b. 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 01b. At this time, the TMTR and TMTAR bits are cleared to 0. When transmission abort is completed with the TAIE bit in the RSCAN0C1CTR register set to 1 (transmission abort interrupt is enabled), an interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00b.

If an arbitration loss has occurred after channel 1 (CAN1) started transmission, the TMTSTS bit is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter 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.

27.9.3.2 Procedure for Transmission from Transmit/Receive FIFO Buffers

Figure 27.28 shows the procedure for transmission from transmit/receive FIFO buffers.
Figure 27.29 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers and transmission has been successfully completed. Figure 27.30 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers and transmission abort has been completed.

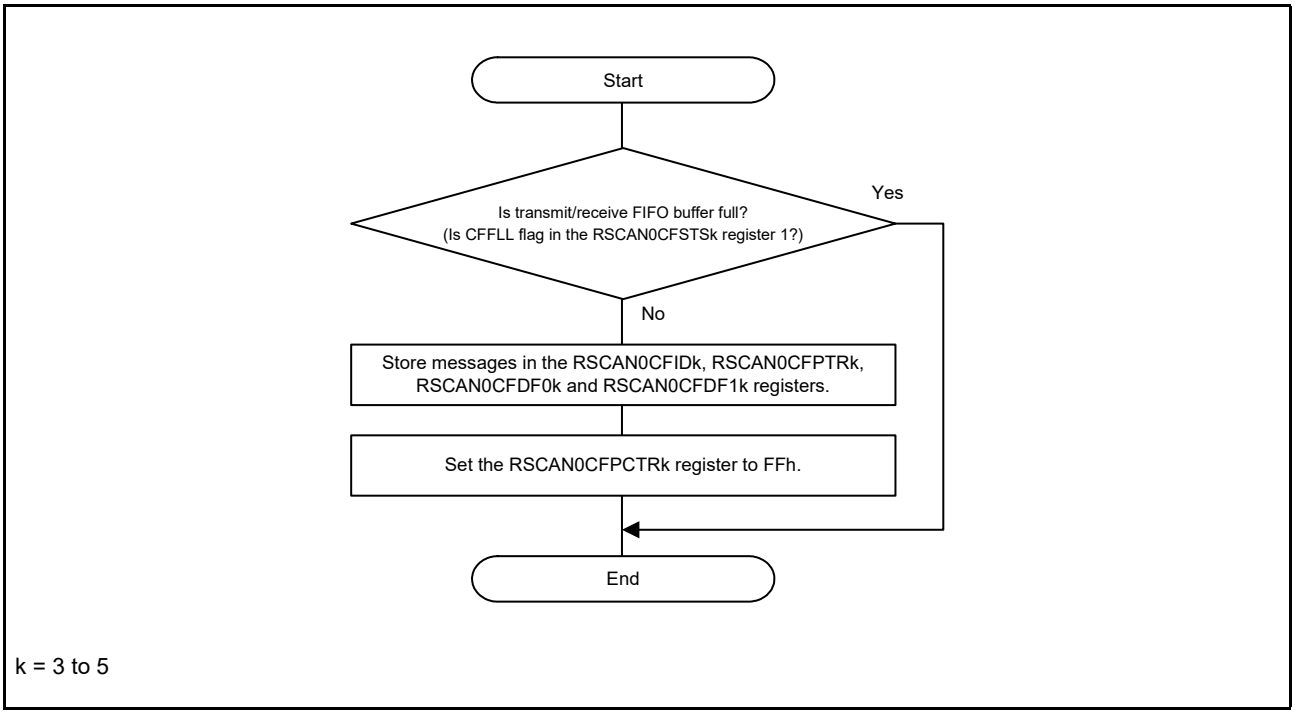
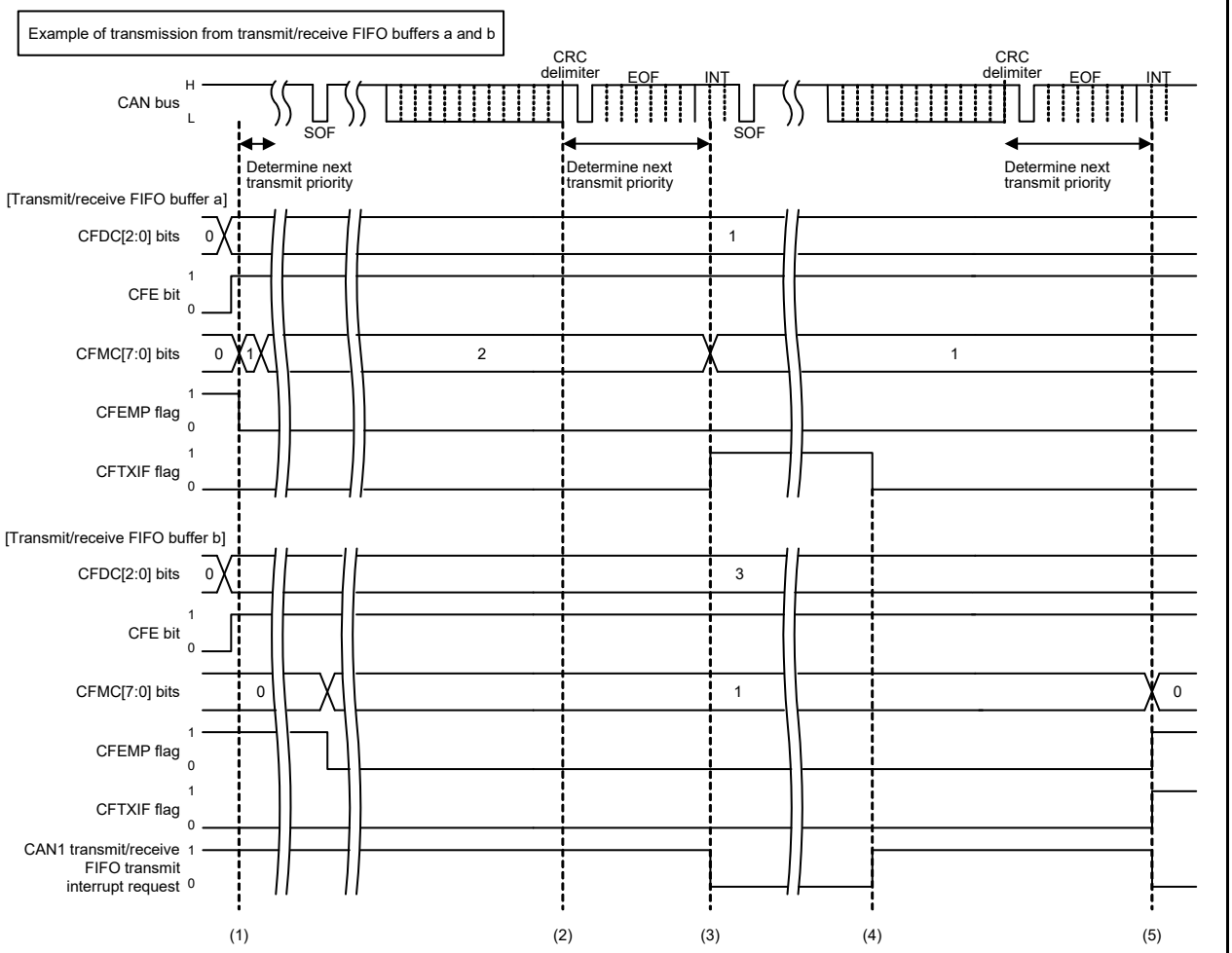


Figure 27.28 Procedure for Transmission from Transmit/Receive FIFO Buffers



a = 3 to 5, b = 3 to 5
CFDC[2:0],CFE: Bits in the RSCAN0CFCCk register
CFMC[7:0], CFEMP, CFTXIF: Flags in the RSCAN0CFSTSk register

Figure 27.29 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) While the CAN bus is idle, when the CFE bit in the RSCA0CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCCa register is 001b (4 messages) or more and the CFMC[7:0] value in the RSCAN0CFSTSa register is 01h 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 1.

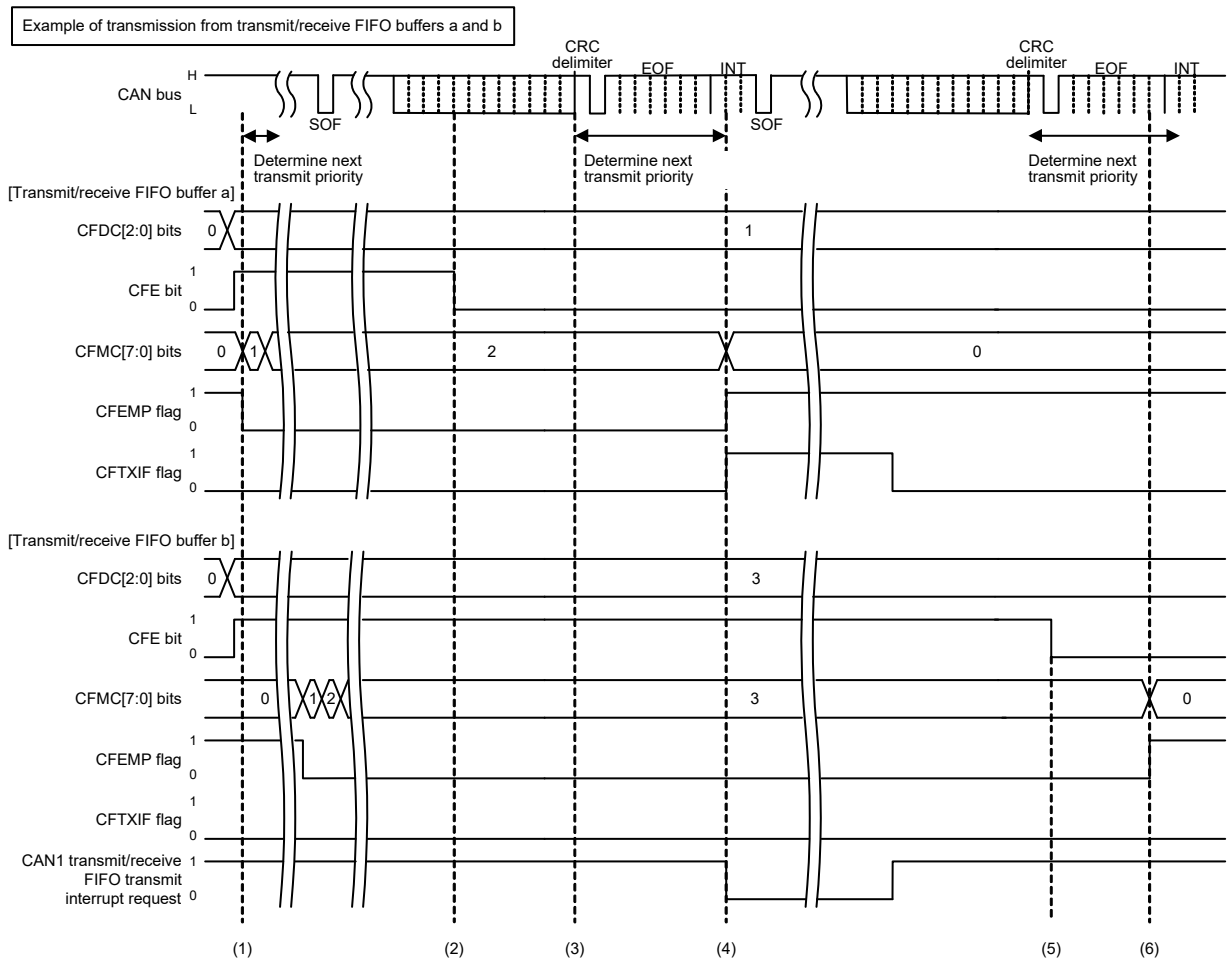
(2) When a transmission request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission.

(3) When transmission completes successfully, the CFMC[7:0] value in the RSCAN0CFSTSa register is decremented. Setting the CFIM bit in the RSCAN0CFCCa register to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCAN0CFSTSk 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 1 has been completed and the CFMC[7:0] value in the RSCAN0CFSTSb register is decremented. The CFMC[7:0] bits are cleared to 00h and therefore the CFEMP flag in the RSCAN0CFSTSk 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 RSCAN0CFSTSa and RSCAN0CFSTSb register is set to 1 (the transmit/receive FIFO buffer is full).



a = 3 to 5, b = 3 to 5
CFDC[2:0], CFE: Bits in the RSCAN0CFCCk register
CFMC[7:0], CFEMP, CFTXIF: Flags in the RSCAN0CFSTSk register

Figure 27.30 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Abort Completed)

(1) While the CAN bus is idle, when the CFE bit in the RSCAN0CFCCa register (a = 3 to 5) is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCCa register is 001b (4 messages) or more and the CFMC[7:0] value in the RSCAN0CFSTSa register is 01h 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 1.

(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 transmission request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. In this figure, transmit/receive FIFO buffer b is not selected as a buffer for the next transmission.

(4) When transmission completes successfully, the CFMC[7:0] value is cleared to 00h. 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 RSCAN0CFSTSa 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 buffers cannot be disabled immediately even if the CFE bit in the RSCAN0CFCCb register is cleared to 0 (no transmit/receive FIFO buffer is used) during transmit priority determination. (The CFEMP flag in the RSCAN0CFSTSb register is not set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) immediately.)

(6) After the internal processing time has passed, transmit/receive FIFO buffers are disabled and the CFMC[7:0] bits in the RSCAN0CFSTSb register are cleared to 00h and the CFEMP flag is set to 1. When the transmit/receive FIFO buffer 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 is immediately disabled. (The CFMC[7:0] bits are cleared to 00h and the CFEMP flag is set to 1.)

27.9.3.3 Procedure for Transmission from the Transmit Queue

Figure 27.31 shows the procedure for transmission from the transmit queue.

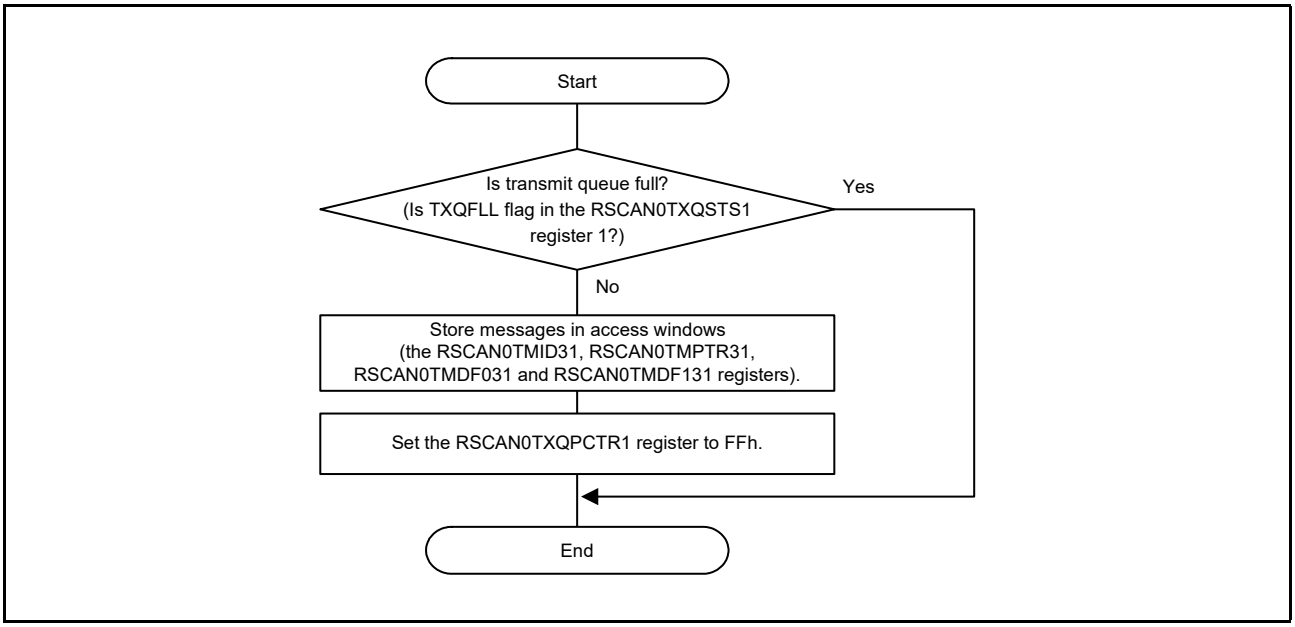


Figure 27.31 Procedure for Transmission from the Transmit Queue

27.9.3.4 Transmission History Buffer Reading Procedure

Transmission history data can be read from the RSCAN0THLACC1 register. The next data can be accessed by writing FFh to the corresponding RSCAN0THLPCTR1 register after reading a set of data. Figure 27.32 shows the transmission history buffer reading procedure.

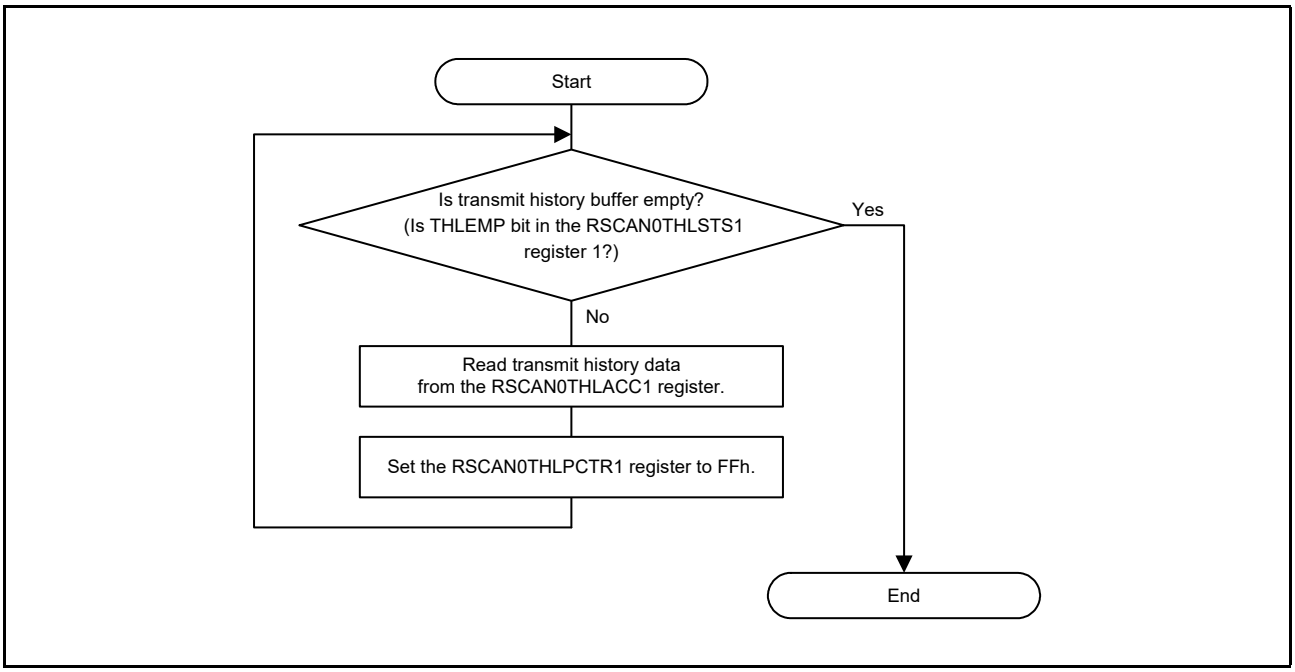


Figure 27.32 Transmission History Buffer Reading Procedure

27.9.4 Test Settings

27.9.4.1 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 27.33 shows the self-test mode setting procedure.

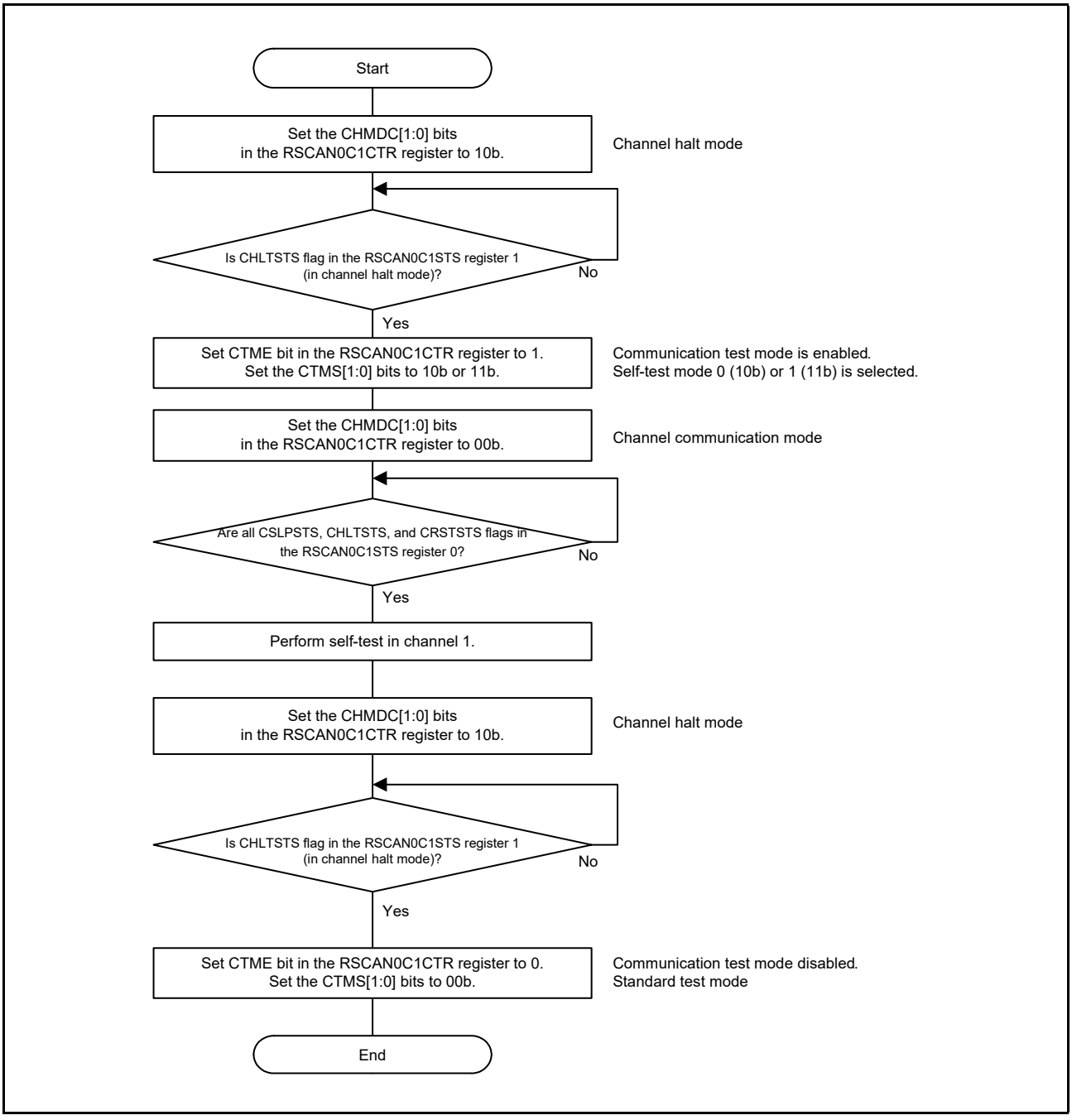


Figure 27.33 Self-Test Mode Setting Procedure

27.9.4.2 Procedure for Unlocking the Protection

Since the global test function in Table 27.24 is protected, write the protection unlock data 1 and unlock data 2 in succession to the LOCK[15:0] bits in the RSCAN0GLOCKK register, then set the target test bit to 1.

Table 27.24 Protection Unlock Data for Test Function

Test Function	Protection Unlock Data 1	Protection Unlock Data 2	Target Bit
RAM test	7575h	8A8Ah	RTME bit in the RSCAN0GTSTCTR register

If an incorrect value is written to the LOCK[15:0] bits, restart from writing the protection unlock data 1. Figure 27.34 shows the procedure for unlocking the protection.

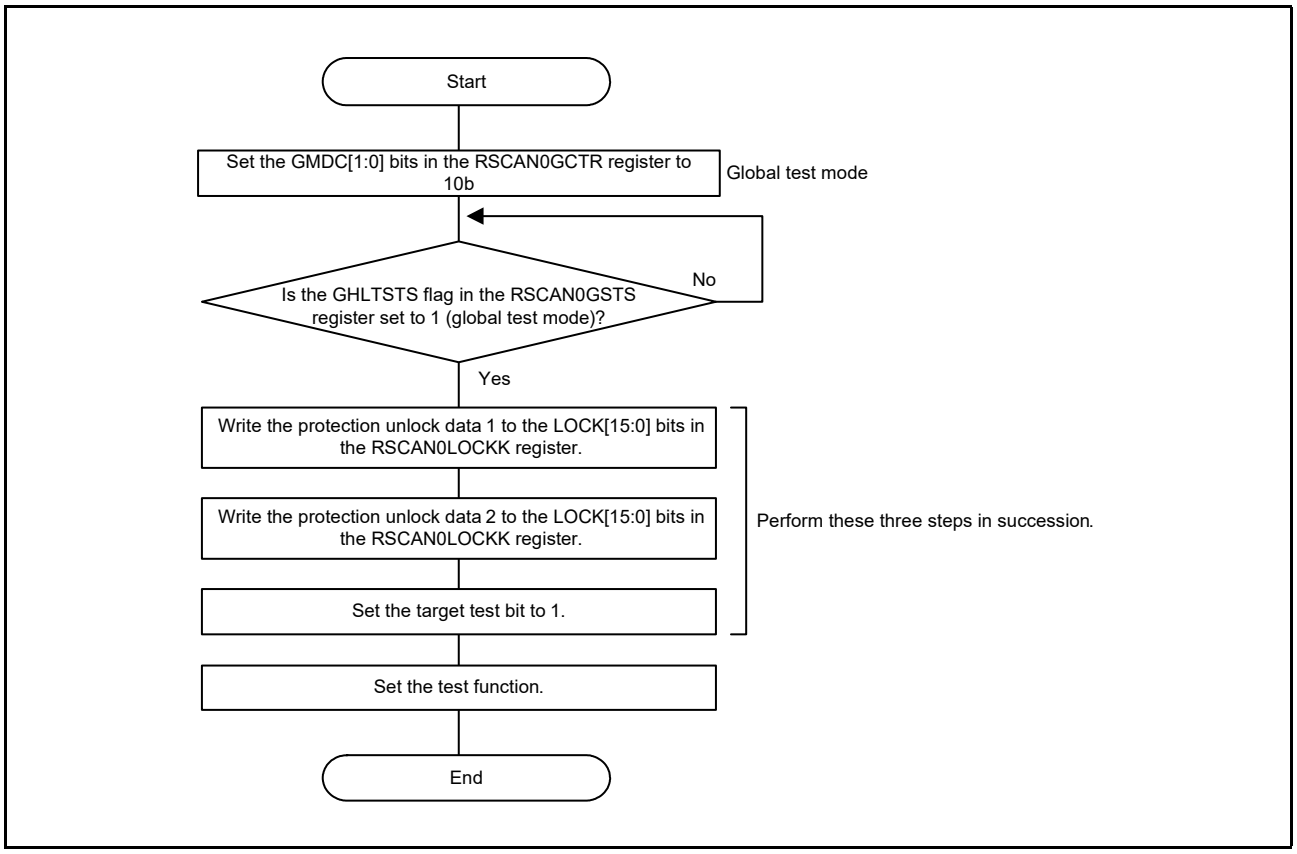


Figure 27.34 Protection Unlock Procedure

27.9.4.3 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 0000h to all pages of the CAN RAM.

Figure 27.35 shows the RAM test setting procedure.

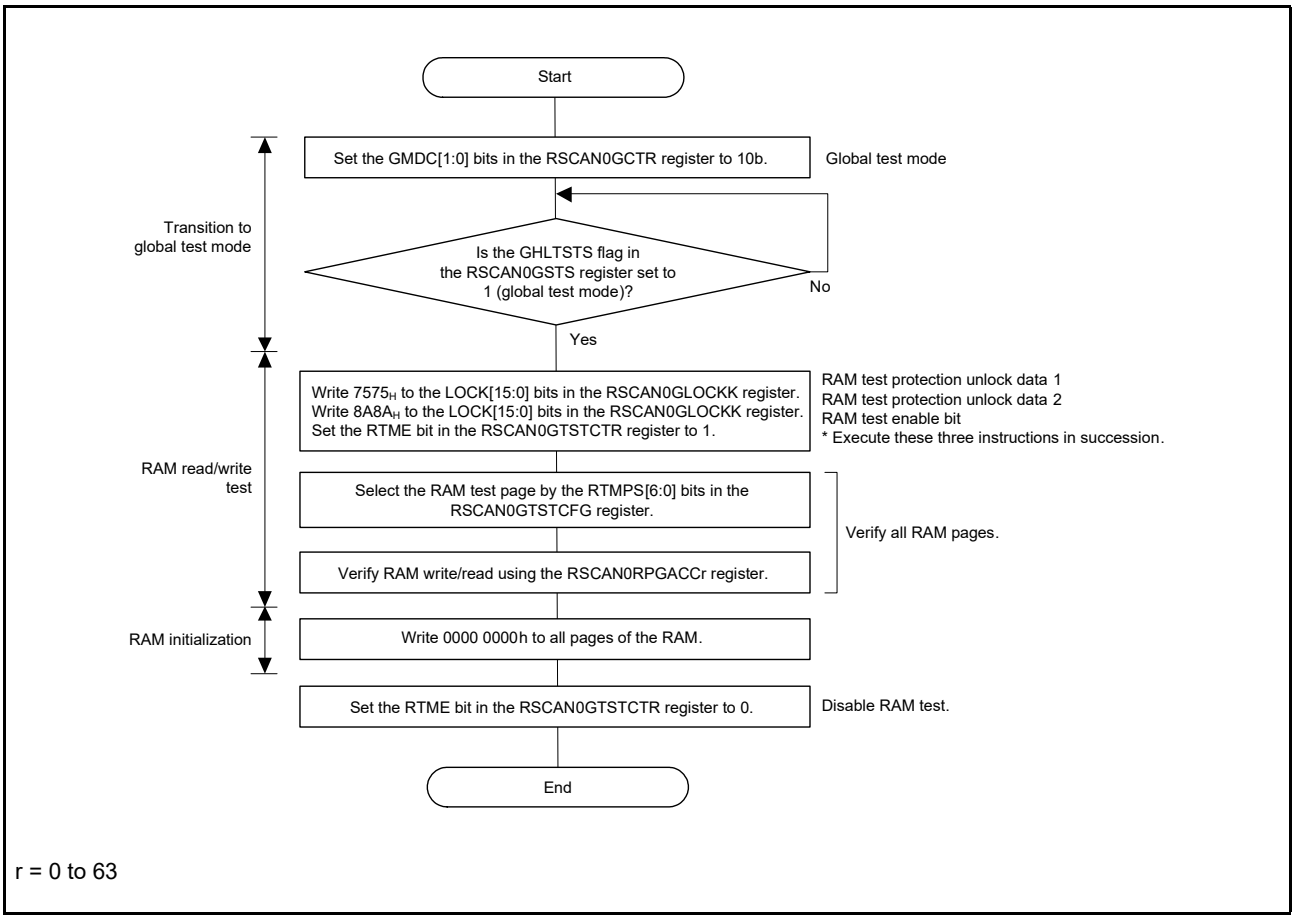


Figure 27.35 RAM Test Setting Procedure

27.10 Detection and Correction of Errors in RSCAN RAM

27.10.1 ECC for the RSCAN RAM

Each buffer RAM of the RSCAN has the following ECC functions.

- ECC error detection/correction
  The RAM is checked for ECC errors. The following options are selectable.
  - 2-bit error detection and 1-bit error detection/correction
  - 2-bit error detection and 1-bit error detection
  The ECC error detection/correction can be disabled by using through mode.
  With the initial settings, error detection/correction is enabled.
- Error notification
  When any of the following errors occurs, it is notified to the error control module (ECM).
  - 1-bit ECC error in the RSCAN RAM
  - 2-bit ECC error in the RSCAN RAM
  - RSCAN overflow error
  In the initial setting, 2-bit error notification is enabled. However, if an interrupt is masked by the FEINTFMSK register, interrupt processing does not proceed.
- Error status
  Detection of 2- and 1-bit ECC errors can be monitored.
  A bit for clearing the error status is provided.

Note: When ECC error detection/correction is performed, initialize the RSCAN RAM by the RSCAN module before it is used.

27.10.2 Output of Errors

The RSCAN outputs the following errors to the error control module (ECM) on detection of a 1-bit or 2-bit ECC error and on overflows of the ECCRCANEADz registers (z = 0 to 7), which hold the addresses where ECC errors were detected. For details, see section 32, Error Control Module (ECM).

- 1-bit ECC error in the RSCAN RAM
- 2-bit ECC error in the RSCAN RAM
- RSCAN overflow error

27.11 Notes on the RSCAN Module

- When changing a global mode, check the GSLPSTS, GHLTSTS, and GRSTSTS flags in the RSCAN0GSTS register for transitions. When changing a channel mode, check the CSLPSTS, CHLTSTS, and CRSTSTS flags in the RSCAN0C1STS register for transitions.
- The acceptance filter processing checks reception rules sequentially in ascending order from the minimum rule number. If the same ID, IDE bit, or RTR bit value is set for multiple reception rules, the minimum number of reception 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 (RSCAN0TMCp) of the corresponding transmit buffer to 00h. The status register (RSCAN0TMSTSp) of the corresponding transmit buffer should not be used. Flags in other status registers (registers RSCAN0TMTRSTS0 to RSCAN0TMTRSTS2, RSCAN0TMTARSTS0 to RSCAN0TMTARSTS2, RSCAN0TMTCSTS0 to RSCAN0TMTCSTS2, and RSCAN0TMTASTS0 to RSCAN0TMTASTS2), 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 RSCAN0TMIEC0 to RSCAN0TMIEC2) to 0 (transmit buffer interrupt is disabled).
- Transmit buffers that are linked to transmit/receive FIFO buffers must not bet allocated to transmit queues.
- 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 CAN1 bit time clock is selected as a timestamp counter clock source, the timestamp counter stops when channel 1 (CAN1) 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.
- The values of unused receive buffers (RSCAN0RMIDq, RSCAN0RMPTRq, RSCAN0RMDF0q, and RSCAN0RMDF1q registers), receive FIFO buffer access registers (RSCAN0RFIDx, RSCAN0RFPTRx, RSCAN0RFDF0x, and RSCAN0RFDF1x registers), and transmit/receive FIFO buffer access registers (RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers) are undefined when the RSCAN module transitions to global operation mode or global test mode after exiting from global reset mode.

28. Serial Peripheral Interface (RSPIa)

28.1 Overview

This LSI includes two channels of serial peripheral interface (RSPI).
The RSPI channels are capable of high-speed, full-duplex synchronous serial communications with multiple processors and peripheral devices.
Table 28.1 lists the specifications of the RSPI, and Figure 28.1 shows a block diagram of the RSPI.
In this section, a lower-case letter m in RSPI command register m (SPCMDm) indicates a value from 0 to 7.

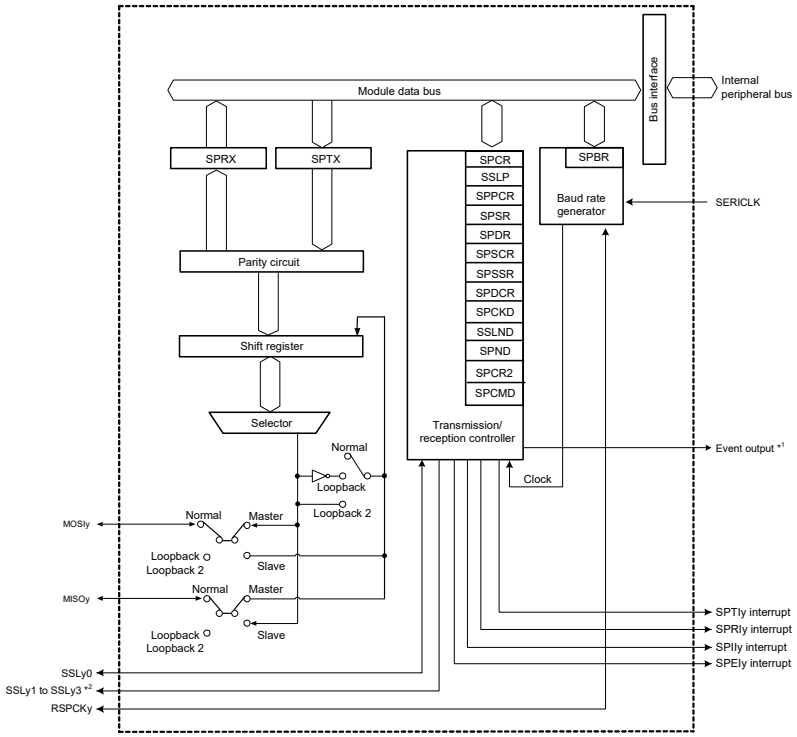
Table 28.1 RSPI Specifications (1 / 2)

Item	Description
Number of channels	Two channels
RSPI transfer functions	• Use of MOSI (master out/slave in), MISO (master in/slave out), SSL (slave select), and RSPCK (RSPI clock) signals allows serial communications through SPI operation (four-wire method) or clock synchronous operation (three-wire method). • Transmit-only operation is available. • Communication mode: Full-duplex or transmit-only can be selected. • Switching of the polarity of RSPCK • Switching of the phase of RSPCK
Data format	• MSB first/LSB first selectable • Transfer bit length is selectable as 8, 9, 10, 11, 12, 13, 14, 15, 16, 20, 24, or 32 bits. • 128-bit transmission/reception buffers • Up to four frames can be transferred in one round of transmission/reception (each frame consisting of up to 32 bits).
Bit rate	• In master mode, the on-chip baud rate generator generates RSPCK by frequency-dividing SERICLK (the division ratio ranges from divided by 4 to divided by 4096). • In slave mode, the minimum SERICLK clock divided by 8 can be input as RSPCK (the maximum frequency of RSPCK is that of SERICLK divided by 8). Width at high level: 4 cycles of SERICLK; width at low level: 4 cycles of SERICLK
Buffer configuration	• Double buffer configuration for the transmission/reception buffers • 128 bits for the transmission/reception buffers
Error detection	• Mode fault error detection • Overrun error detection*1 • Parity error detection
SSL control function	• Four SSL pins for channel 0 (SSL00 to SSL03) and two SSL pins for channel 1 (SSL10, SSL11) • In single-master mode, SSL00 to SSL03, SSL10, and SSL11 pins are output. • In multi-master mode: SSL00 and SSL10 pins for input, and SSL01 to SSL03 and SSL11 pins for either output or unused. • In slave mode: SSL00 and SSL10 pins for input, and SSL01 to SSL03 and SSL11 pins for unused. • Controllable delay from SSL output assertion to RSPCK operation (RSPCK delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Controllable delay from RSPCK stop to SSL output negation (SSL negation delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Controllable wait for next-access SSL output assertion (next-access delay) Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units) • Function for changing SSL polarity
Control in master transfer	• A transfer of up to eight commands can be executed sequentially in looped execution. • For each command, the following can be set: SSL signal value, bit rate, RSPCK polarity/phase, transfer data length, MSB/LSB first, burst, RSPCK delay, SSL negation delay, and next-access delay • A transfer can be initiated by writing to the transmission buffer. • MOSI signal value specifiable in SSL negation • RSPCK auto-stop function
Interrupt sources	• Interrupt sources Reception buffer full interrupt Transmission buffer empty interrupt RSPI error interrupt (mode fault, overrun, parity error) RSPI idle interrupt (RSPI idle)

Table 28.1 RSPI Specifications (2 / 2)

Item	Description
Event link function*2 (output)	• The following events can be output to the event link controller. Reception buffer full signal Transmission buffer empty signal Mode fault, overrun, or parity error signal RSPI idle signal Transmission-completed signal
Others	• Function for switching between CMOS output and open-drain output • Function for initializing the RSPI • Loopback mode
Low-power consumption function	Module-stop state can be set.

Note 1. In master reception and when the RSPCK auto-stop function is enabled, an overrun error does not occur because the transfer clock is stopped at the timing of overrun error detection.
Note 2. Only for channel 0 (RSPI0)



SPCR:	RSPI control register	SPDCR:	RSPI data control register
SPCR2:	RSPI control register 2	SPCKD:	RSPI clock delay register
SSLP:	RSPI slave select polarity register	SSLND:	RSPI slave select negation delay register
SPPCR:	RSPI pin control register	SPND:	RSPI next-access delay register
SPSR:	RSPI status register	SPCMD:	RSPI command registers 0 to 7 (eight registers)
SPDR:	RSPI data register		
SPSCR:	RSPI sequence control register	SPBR:	RSPI bit rate register
SPSSR:	RSPI sequence status register	SPTX:	RSPI transmission buffer
		SPRX:	RSPI reception buffer

Note 1. Only for channel 0.
Note 2. The number of pins differs for each channel (SSL01 to SSL03 for channel 0; SSL11 for channel 1).
y = 0, 1 (for all channels)

Figure 28.1 RSPI Block Diagram

Table 28.2 lists the I/O pins used in the RSPI.
The RSPI automatically switches the I/O direction of the SSLy0 pin. SSLy0 is set as an output when the RSPI is a single master and as an input when the RSPI is a multi-master or a slave. Pins RSPCKy, MOSIy, and MISOy are automatically set as inputs or outputs according to the setting of master or slave and the level input on the SSLy0 pin.
Refer to section 28.3.2, Controlling RSPI Pins.

Table 28.2 RSPI Pin Configuration

Channel	Pin Name	I/O	Function
RSPI0	RSPCK0	I/O	Clock I/O
	MOSI0	I/O	Master transmit data I/O
	MISO0	I/O	Slave transmit data I/O
	SSL00	I/O	Slave selection signal I/O
	SSL01	Output	Slave selection signal output
	SSL02	Output	Slave selection signal output
	SSL03	Output	Slave selection signal output
RSPI1	RSPCK1	I/O	Clock I/O
	MOSI1	I/O	Master transmit data I/O
	MISO1	I/O	Slave transmit data I/O
	SSL10	I/O	Slave selection signal I/O
	SSL11	Output	Slave selection signal output

Note: The number of SSL pins differs from channel to channel. The indices to identify the channels are from 0 and 1.

28.2 Register Descriptions

28.2.1 RSPI Control Register (SPCR)

The SPCR register controls the settings for operation of the RSPI.
The SPCR.MSTR, SPCR.MODFEN, and SPCR.TXMD bits should be set while the setting of the SPCR.SPE bit is 0. Writing to these bits is prohibited while the setting of the SPCR.SPE bit is 1.

Address(es): RSPI0.SPCR A006 8000h, RSPI1.SPCR A006 8400h

	b7	b6	b5	b4	b3	b2	b1	b0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	TXMD	SPMS
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SPMS	RSPI Mode Select	0: SPI operation (four-wire method) 1: Clock synchronous operation (three-wire method)	R/W
b1	TXMD	Communications Operating Mode Select	0: Full-duplex synchronous serial communications 1: Serial communications consisting of only transmit operations	R/W
b2	MODFEN	Mode Fault Error Detection Enable	0: Disables the detection of mode fault error 1: Enables the detection of mode fault error	R/W
b3	MSTR	Master/Slave Mode Select	0: Slave mode 1: Master mode	R/W
b4	SPEIE	Error Interrupt Enable	0: Disables the generation of error interrupt requests 1: Enables the generation of error interrupt requests	R/W
b5	SPTIE	Transmission Buffer Empty Interrupt Enable	0: Disables the generation of transmission buffer empty interrupt requests 1: Enables the generation of transmission buffer empty interrupt requests	R/W
b6	SPE	RSPI Function Enable	0: Disables the RSPI function 1: Enables the RSPI function	R/W
b7	SPRIE	Reception Buffer Full Interrupt Enable	0: Disables the generation of reception buffer full interrupt requests 1: Enables the generation of reception buffer full interrupt requests	R/W

SPMS Bit (RSPI Mode Select)

The SPMS bit selects SPI operation (four-wire method) or clock synchronous operation (three-wire method).
The SSLy0 to SSLy3 pins are not used in clock synchronous operation. The three pins RSPCKy, MOSIy, and MISOy handle communications. If clock synchronous operation is to proceed in master mode (SPCR.MSTR = 1), the SPCMDm.CPHA bit can be set to either 0 or 1. Set the CPHA bit to 1 if clock synchronous operation is to proceed in slave mode (SPCR.MSTR = 0). Do not set the CPHA bit to 0 when clock synchronous operation is to proceed in slave mode (SPCR.MSTR = 0). (y = 0, 1)

TXMD Bit (Communications Operating Mode Select)

The TXMD bit selects the operation for full-duplex synchronous serial communications or the operation for transmission only.
When performing communications with the TXMD bit set to 1, the RSPI performs only transmit operations and not receive operations (refer to section 28.3.6, Communications Operating Mode).
When the TXMD bit is set to 1, reception buffer full interrupt requests cannot be used.

MODFEN Bit (Mode Fault Error Detection Enable)

The MODFEN bit enables or disables the detection of mode fault error (refer to section 28.3.8, Error Detection). In addition, the RSPI determines the I/O direction of the SSLy0 to SSLy3 (y = 0, 1) pins based on combinations of the MODFEN and MSTR bits (refer to section 28.3.2, Controlling RSPI Pins).

MSTR Bit (RSPI Master/Slave Mode Select)

The MSTR bit selects master/slave mode of the RSPI. According to MSTR bit settings, the RSPI determines the direction of pins RSPCKy, MOSIy, and MISOy, and SSLy0 to SSLy3 (y = 0, 1).

SPEIE Bit (RSPI Error Interrupt Enable)

The SPEIE bit enables or disables the generation of RSPI error interrupt requests when the RSPI detects a mode fault error and sets the SPSR.MODF flag to 1, when the RSPI detects an overrun error and sets the SPSR.OVRF flag to 1, or when the RSPI detects a parity error and sets the SPSR.PERF flag to 1 (refer to section 28.3.8, Error Detection).

SPTIE Bit (Transmission Buffer Empty Interrupt Enable)

The SPTIE bit enables or disables a transmission buffer empty interrupt request generated when the RSPI detects that the transmission buffer is empty.
If the RSPI function is disabled (i.e. the SPE bit is set to 0), the transmission buffer being empty is detected. If the SPTIE bit is set to 1 at this time, a transmission buffer empty interrupt is generated.
Note that a transfer buffer empty interrupt request is also generated when the SPTIE and SPE bits are set to 1 at the same time at the start of transmission.

SPE Bit (RSPI Function Enable)

The SPE bit enables or disables the RSPI function.
When the SPSR.MODF bit is 1, the SPE bit cannot be set to 1. For details, refer to section 28.3.8, Error Detection.
Setting the SPE bit to 0 disables the RSPI function, and initializes a part of the module function. For details, refer to section 28.3.9, Initializing RSPI. Furthermore, when the setting of the SPTIE bit is 1 (enabling the generation of transmission buffer empty interrupts), a transmission buffer empty interrupt request is generated by the state of the SPE bit changing from 0 to 1 or from 1 to 0.

SPRIE Bit (RSPI Reception Buffer Full Interrupt Enable)

The SPRIE bit enables or disables RSPI reception buffer full interrupt requests when the RSPI has detected the reception buffer being full after the completion of a serial transfer.

28.2.2 RSPI Slave Select Polarity Register (SSLP)

The SSLP register controls the active sense of the RSPIy (y = 0, 1) slave select signals.
The SSLP register should be set while the setting of the SPCR.SPE bit is 0. Writing to this register is prohibited while the setting of the SPCR.SPE bit is 1.

Address(es): RSPI0.SSLP A006 8001h, RSPI1.SSLP A006 8401h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	SSL3P	SSL2P	SSL1P	SSL0P
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SSL0P	SSL0 Signal Polarity Setting	0: SSLy0 signal is active low 1: SSLy0 signal is active high	R/W
b1	SSL1P	SSL1 Signal Polarity Setting	0: SSLy1 signal is active low 1: SSLy1 signal is active high	R/W
b2	SSL2P	SSL2 Signal Polarity Setting*1	0: SSLy2 signal is active low 1: SSLy2 signal is active high	R/W
b3	SSL3P	SSL3 Signal Polarity Setting*1	0: SSLy3 signal is active low 1: SSLy3 signal is active high	R/W
b7 to b4	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

Note 1. Bits for channel 1 are reserved. These bits are read as 0. The write value should be 0.

28.2.3 RSPI Pin Control Register (SPPCR)

The SPPCR register controls the output setting for RSPI output pins.

The SPPCR register should be set while the setting of the SPCR.SPE bit is 0. Writing to this register is prohibited while the setting of the SPCR.SPE bit is 1.

Address(es): RSPI0.SPPCR A006 8002h, RSPI1.SPPCR A006 8402h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	MOIFE	MOIFV	—	SPOM	SPLP2	SPLP
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SPLP	RSPI Loopback	0: Normal mode 1: Loopback mode (data is inverted for transmission)	R/W
b1	SPLP2	RSPI Loopback 2	0: Normal mode 1: Loopback mode (data is not inverted for transmission)	R/W
b2	SPOM	Output Pin Mode	0: CMOS output 1: Open-drain output	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	MOIFV	MOSI Idle Fixed Value	0: The level output on the MOSIy pin during MOSI idling corresponds to low. 1: The level output on the MOSIy pin during MOSI idling corresponds to high.	R/W
b5	MOIFE	MOSI Idle Value Fixing Enable	0: MOSI output value equals final data from previous transfer 1: MOSI output value equals the value set in the MOIFV bit	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPLP Bit (RSPI Loopback)

The SPLP bit selects the mode of the RSPI pins.

When the SPLP bit is set to 1, the RSPI shuts off the path between the MISOy pin and the shift register if the SPCR.MSTR bit is 1, and between the MOSIy pin and the shift register if the SPCR.MSTR bit is 0, and connects (inverts) the input path and output path for the shift register (loopback mode).

(y = 0, 1)

SPLP2 Bit (RSPI Loopback 2)

The SPLP2 bit selects the mode of the RSPI pins.

When the SPLP2 bit is set to 1, the RSPI shuts off the path between the MISOy pin and the shift register if the SPCR.MSTR bit is 1, and between the MOSIy pin and the shift register if the SPCR.MSTR bit is 0, and connects the input path and output path for the shift register (loopback mode).

(y = 0, 1)

SPOM Bit (Output Pin Mode)

The SPOM bit is used to set the RSPI output pins to CMOS output or open-drain output. For details, see section 28.3.2, Controlling RSPI Pins.

MOIFV Bit (MOSI Idle Fixed Value)

If the MOIFE bit is 1 in master mode, the MOIFV bit determines the MOSIy pin output value as low or high during the SSL negation period (including the SSL retention period during a burst transfer).

MOIFE Bit (MOSI Idle Value Fixing Enable)

The MOIFE bit selects whether the MOSIy output level is to be fixed over the SSL negation period (including the SSL retention period during a burst transfer) when the RSPI is in master mode. When the MOIFE bit is 0, the RSPI continues to output the value of the last bit from the previous serial transfer during the SSL negation period to the MOSIy pin. When the MOIFE bit is 1, the RSPI outputs a fixed low or high level on the MOSIy pin according to the setting of the MOIFV bit.
(y = 0, 1)

28.2.4 RSPI Status Register (SPSR)

The SPSR register indicates the state of RSPI transfer.

Address(es): RSPI0.SPSR A006 8003h, RSPI1.SPSR A006 8403h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	PERF	MODF	IDLNF	OVRF
Value after reset:	x	0	x	0	0	0	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	OVRF	Overrun Error Flag	0: No overrun error occurs 1: An overrun error occurs	R/(W) *1
b1	IDLNF	RSPI Idle Flag	0: RSPI is in the idle state 1: RSPI is in the transfer state	R
b2	MODF	Mode Fault Error Flag	0: No mode fault error occurs 1: A mode fault error occurs	R/(W) *1
b3	PERF	Parity Error Flag	0: No parity error occurs 1: A parity error occurs	R/(W) *1
b4	—	Reserved	This bit is read as 0. The write value should be 0.	R/(W)
b5	—	Reserved	The read value is undefined. The write value should be 1.	R/(W)
b6	—	Reserved	This bit is read as 0. The write value should be 0.	R/(W)
b7	—	Reserved	The read value is undefined. The write value should be 1.	R/(W)

Note 1. Only 0 can be written to clear the flag after reading 1.

OVRF Flag (Overrun Error Flag)

The OVRF flag indicates the occurrence of an overrun error. In master mode (when the SPCR.MSTR bit is 1) and when the RSPCK clock auto-stop function is enabled (the SPCR1.SCKASE bit is 1), an overrun error does not occur; accordingly this flag does not become 1. For details, see section 28.3.8.1, Overrun Error.

[Setting condition]

- When the next serial transfer ends while the SPCR.TXMD bit is 0 and the reception buffer is full.

[Clearing condition]

- When SPSR is read while the OVRF flag is 1, and then writes the value 0 to the OVRF flag.

IDLNF Flag (RSPI Idle Flag)

The IDLNF flag indicates the transfer status of the RSPI.
[Setting condition]
Master mode

- Condition 1 and condition 2 are not satisfied in master mode under the [Clearing condition] below.

Slave mode

- The SPCR.SPE bit is 1 (RSPI function is enabled)

[Clearing condition]
Master mode

- The following 1 is satisfied (condition 1) or all of the following 2 to 4 are satisfied (condition 2).

1. The SPCR.SPE bit is 0 (RSPI is initialized)
2. The transmission buffer (SPTX) is empty (data for the next transfer is not set)
3. The SPSSR.SPCP[2:0] bits are 000b (beginning of sequence control)
4. The RSPI internal sequencer has entered the idle state (status in which operations up to the next-access delay have finished)

Slave mode

- The SPCR.SPE bit is 0 (RSPI is initialized)

MODF Flag (Mode Fault Error Flag)

Indicates the occurrence of a mode fault error.
[Setting condition]
Multi-master mode

- When the input level of the SSLyi pin changes to the active level while the SPCR.MSTR bit is 1 (master mode) and the SPCR.MODFEN bit is 1 (mode fault error detection is enabled), the RSPI detects a mode fault error

Slave mode

- When the SSLyi pin is negated before the RSPCK cycle necessary for data transfer ends while the SPCR.MSTR bit is 0 (slave mode) and the SPCR.MODFEN bit is 1 (mode fault error detection is enabled), the RSPI detects a mode fault error

The active level of the SSLyi signal is determined by the SSLP.SSLiP bit (SSLi signal polarity setting bit).
(y = 0, 1 (for all channels); i = 0 to 3)
[Clearing condition]

- When SPSR is read while the MODF flag is 1, and then writes the value 0 to the MODF flag

PERF Flag (Parity Error Flag)

Indicates the occurrence of a parity error.
[Setting condition]

- When a serial transfer ends while the SPCR.TXMD bit is 0 and the SPCR2.SPPE bit is 1, the RSPI detects a parity error

[Clearing condition]

- When SPSR is read while the PERF flag is 1, and then writes the value 0 to the PERF flag

28.2.5 RSPI Data Register (SPDR)

SPDR is the interface with the buffers that hold data for transmission and reception by the RSPI.
When accessing in longwords (the SPLW bit is 1), access SPDR.
When accessing in words (the SPLW bit is 0), access the higher-order 16 bits (H) of SPDR.

Address(es): RSPI0.SPDR A006 8004h, RSPI1.SPDR A006 8404h

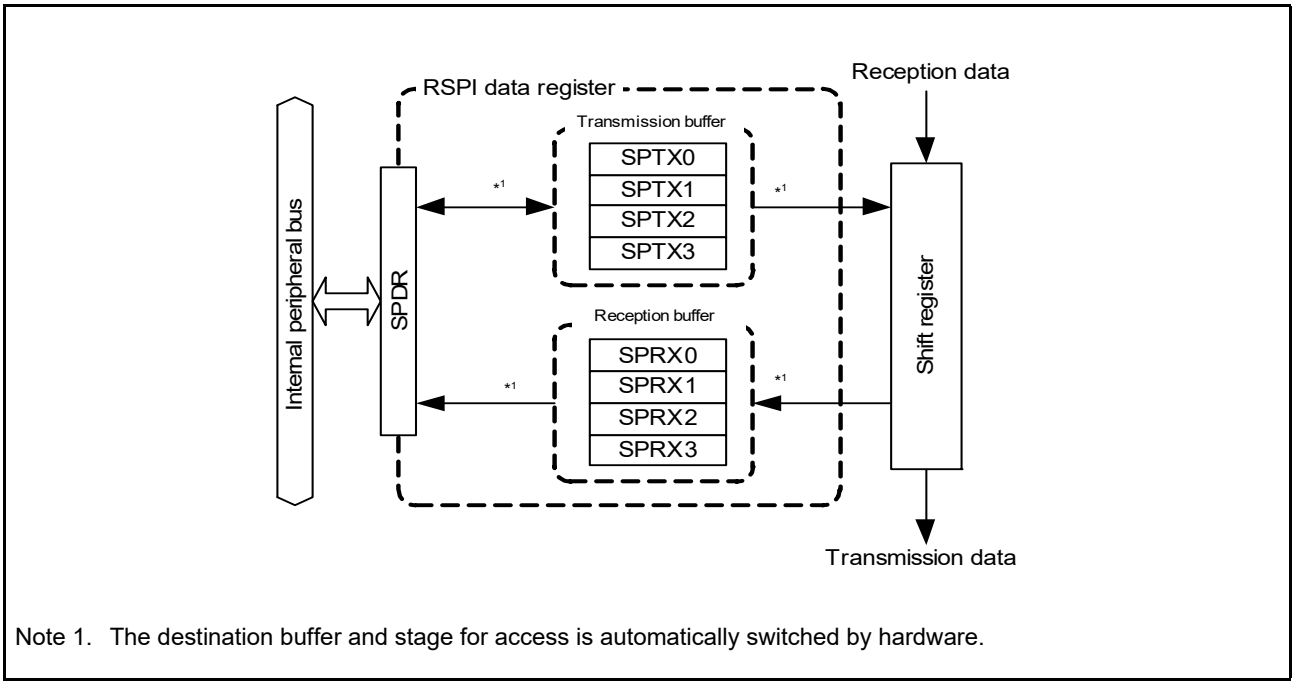
	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Address(es): RSPI0.SPDR A006 8004h, RSPI1.SPDR A006 8404h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	H															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

The transmission buffer (SPTX) and reception buffer (SPRX) are independent but are both mapped to SPDR. Figure 28.2 shows the configuration of SPDR.



Note 1. The destination buffer and stage for access is automatically switched by hardware.

Figure 28.2 Configuration of SPDR

The transmit and reception buffers each have four stages. The number of stages to be used is selectable by the number of frames specification bits in the RSPI data control register (SPDCR.SPFC[1:0]). The eight stages of the buffer are all mapped to the single address of SPDR.
Data written to SPDR are written to a transmission-buffer stage (SPTXn) (n = 0 to 3) and then transmitted from the buffer. The reception buffer holds received data on completion of reception. The reception buffer is not updated if an overrun is generated.
Furthermore, if the data length is other than 32 bits, bits not referred to in SPTXn (n = 0 to 3) are stored in the corresponding bits in SPRXn (n = 0 to 3). For example, if the data length is 9 bits, received data are stored in the SPRXn[8:0] bits and the SPTXn[31:9] bits are stored in the SPRXn[31:9] bits.

(1) Bus Interface

SPDR is the interface with 32-bit wide transmit and reception buffers, each of which has four stages, for a total of 32 bytes. In other words, the 32 bytes are mapped to the 4-byte address space for SPDR. Furthermore, the unit of access for SPDR is selected by the RSPI longword access/word access specification bit in the RSPI data control register (SPDCR.SPLW).
Data for transmission should be flush with the LSB end of the register. Received data are stored flush with the LSB end.
Operations involved in writing to and reading from SPDR are described below.

(a) Writing

Data written to SPDR are written to a transmission buffer (SPTXn). This is not influenced by the value of the SPDCR.SPRDTD bit unlike when reading from SPDR.
The transmission buffer includes a transmission buffer write pointer which is automatically updated to indicate the next stage each time data are written to SPDR.
Figure 28.3 shows the configuration of the bus interface with the transmission buffer in the case of writing to SPDR.

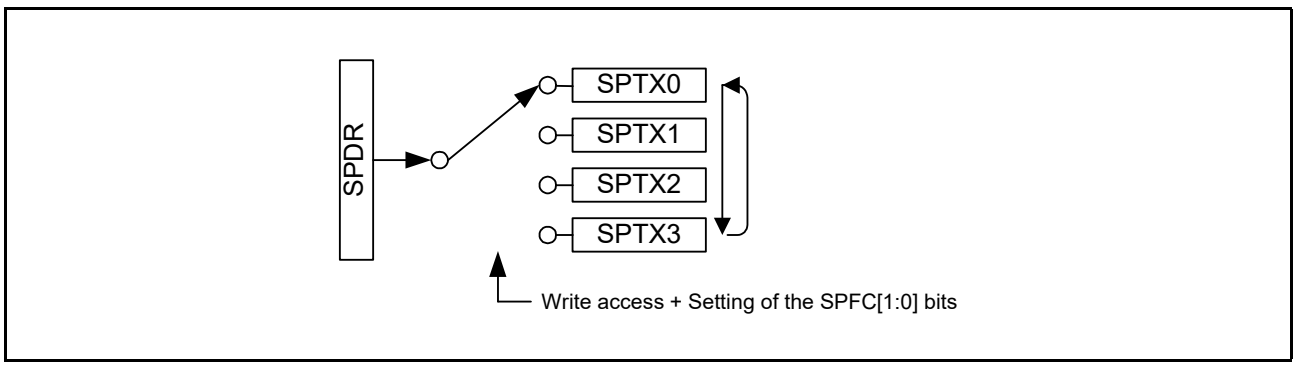


Figure 28.3 Configuration of SPDR (Writing)

The sequence for switching the transmission buffer write pointer differs with the setting of the number of frames specification bits in the RSPI data control register (SPDCR.SPFC[1:0]).

- Settings of the SPFC[1:0] bits and sequence of switching the pointer among SPTX0 to SPTX3.
  When the SPFC[1:0] bits are 00b: SPTX0 → SPTX0 → SPTX0 → …
  When the SPFC[1:0] bits are 01b: SPTX0 → SPTX1 → SPTX0 → SPTX1 → …
  When the SPFC[1:0] bits are 10b: SPTX0 → SPTX1 → SPTX2 → SPTX0 → SPTX1 → …
  When the SPFC[1:0] bits are 11b: SPTX0 → SPTX1 → SPTX2 → SPTX3 → SPTX0 → SPTX1 → …

When 1 is written to the RSPI function enable bit in the RSPI control register (SPCR.SPE) while the bit's current value is 0, SPTX0 will be the destination the next time writing proceeds.
When writing to the transmission buffer (SPTXn) after generation of the transmission buffer empty interrupt, write the number of frames set by the number of frames specification bits (SPFC[1:0]) in the RSPI data control register (SPDCR). Even if the number of frames is written to the transmission buffer (SPTXn), the value of the buffer is not updated after completion of the writing and before generation of the next transmission buffer empty interrupt.

(b) Reading

SPDR can be read to read the value of a reception buffer (SPRXn) or a transmission buffer (SPTXn). The setting of the RSPI receive/transmit data selection bit in the RSPI data control register (SPDCR.SPRDTD) selects whether reading is of the receive or transmission buffer.
The sequence of reading the SPDR register is controlled by independent pointers, reception buffer read pointer and transmission buffer read pointer.
Figure 28.4 shows the configuration of the bus interface with the receive and transmission buffers in the case of reading from SPDR.

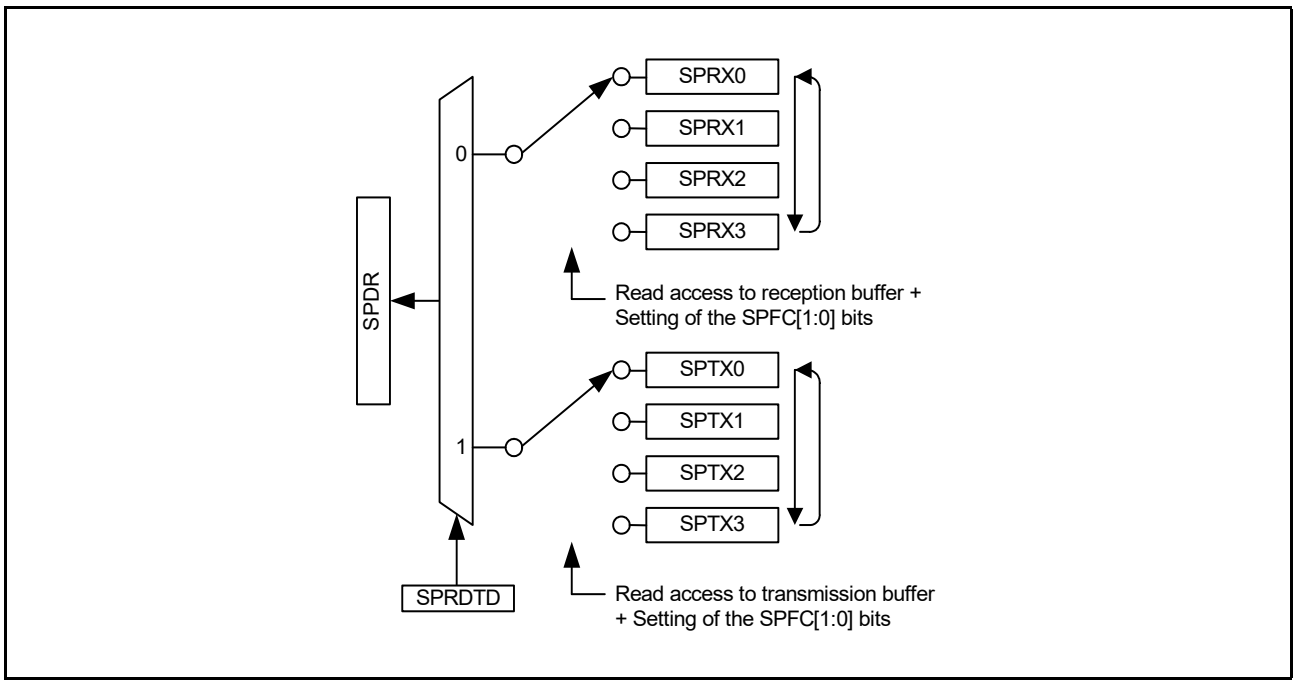


Figure 28.4 Configuration of SPDR (Reading)

Reading the reception buffer switches the reception buffer read pointer to the next buffer automatically.
The sequence of switching the reception buffer read pointer is the same as that for the transmission buffer write pointer. However, when 1 is written to the RSRI function enable bit in the RSPI control register (SPCR.SPE) while the bit's current value is 1, SPRX0 will be indicated by the buffer read pointer the next time reading proceeds.
The transmission buffer read pointer is updated when writing to SPDR, and not updated when reading from the transmission buffer. When reading from the transmission buffer, the value most recently written to SPDR is read. However, after generation of the transmission buffer empty interrupt, the values read from the transmission buffer are all 0 in the interval after completion of writing the number of frames of data specified in the number of frames specification bits (SPDCR.SPFC[1:0]) and before generation of the next buffer empty interrupt.

28.2.6 RSPI Sequence Control Register (SPSCR)

SPSCR sets the sequence length when the RSPI operates in master mode. When changing the SPSCR.SPSLN[2:0] bits while both the SPCR.MSTR and SPCR.SPE bits are 1, the bits should be changed while the SPSR.IDLNF flag is 0.

Address(es): RSPI0.SPSCR A006 8008h, RSPI1.SPSCR A006 8408h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SPSLN[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SPSLN[2:0]	RSPI Sequence Length Specification	b2 b1 b0: Sequence Length — Referenced SPCMD0 to SPCMD7 (No.) 0 0 0: 1 — 0→0→… 0 0 1: 2 — 0→1→0→… 0 1 0: 3 — 0→1→2→0→… 0 1 1: 4 — 0→1→2→3→0→… 1 0 0: 5 — 0→1→2→3→4→0→… 1 0 1: 6 — 0→1→2→3→4→5→0→… 1 1 0: 7 — 0→1→2→3→4→5→6→0→… 1 1 1: 8 — 0→1→2→3→4→5→6→7→0→… The order in which the SPCMD0 to SPCMD07 registers are to be referenced is changed in accordance with the sequence length that is set in these bits. The relationship among the setting of these bits, sequence length, and SPCMD0 to SPCMD7 registers referenced by the RSPI is shown above. However, the RSPI in slave mode always references SPCMD0.	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPSLN[2:0] Bits (RSPI Sequence Length Specification)

The SPSLN[2:0] bits specify a sequence length when the RSPI in master mode performs sequential operations. The RSPI in master mode changes SPCMD0 to SPCMD7 registers to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN[2:0] bits.

In slave mode, SPCMD0 is always referred to.

28.2.7 RSPI Sequence Status Register (SPSSR)

SPSSR indicates the sequence control status when the RSPI operates in master mode.
Any writing to SPSSR is ignored.

Address(es): RSPI0.SPSSR A006 8009h, RSPI1.SPSSR A006 8409h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	SPECM[2:0]			—	SPCP[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SPCP[2:0]	RSPI Command Pointer	b2 b0 0 0 0: SPCMD0 0 0 1: SPCMD1 0 1 0: SPCMD2 0 1 1: SPCMD3 1 0 0: SPCMD4 1 0 1: SPCMD5 1 1 0: SPCMD6 1 1 1: SPCMD7	R
b3	—	Reserved	This bit is read as 0.	R
b6 to b4	SPECM[2:0]	RSPI Error Command	b6 b4 0 0 0: SPCMD0 0 0 1: SPCMD1 0 1 0: SPCMD2 0 1 1: SPCMD3 1 0 0: SPCMD4 1 0 1: SPCMD5 1 1 0: SPCMD6 1 1 1: SPCMD7	R
b7	—	Reserved	This bit is read as 0.	R

SPCP[2:0] Bits (RSPI Command Pointer)

The SPCP[2:0] bits indicate SPCMDm that is currently pointed to by the pointer during sequence control by the RSPI. For the RSPI's sequence control, refer to section 28.3.10.1, Master Mode Operation.

SPECM[2:0] Bits (RSPI Error Command)

The SPECM[2:0] bits indicate SPCMDm that is specified by the SPCP[2:0] bits when an error is detected during sequence control by the RSPI. The RSPI updates the SPECM[2:0] bits only when an error is detected. If both the SPSR.OVRF and SPSR.MODF bits are 0 and there is no error, the values of the SPECM[2:0] bits have no meaning. For the RSPI's error detection function, refer to section 28.3.8, Error Detection. For the RSPI's sequence control, refer to section 28.3.10.1, Master Mode Operation.

28.2.8 RSPI Bit Rate Register (SPBR)

SPBR controls the bit rate settings in master mode. Writing to SPBR is prohibited while both the SPCR.MSTR and SPCR.SPE bits are 1.

Address(es): RSPI0.SPBR A006 800Ah, RSPI1.SPBR A006 840Ah

When the RSPI is used in slave mode, the bit rate depends on the bit rate of the input clock (bit rate satisfying the electrical characteristics should be used) regardless of the settings of SPBR and the SPCMDm.BRDV[1:0] bits (bit rate division setting bits).
The bit rate is determined by combinations of the SPBR setting and the SPCMDm.BRDV[1:0] bit setting. The equation for calculating the bit rate is given below. In the equation, n denotes an SPBR setting (0, 1, 2, …, 255), and N denotes a BRDV[1:0] bit setting (0, 1, 2, 3) (m = 0 to 7). However, setting n = 0 (SPR[7:0] = 0) and N = 0 (BRDV[1:0] = 0) is prohibited.

$$\text{Bit rate} = \frac{f(\text{SERICLK})}{2 \times (n + 1)\, 2^{N}}$$

Table 28.3 lists examples of the relationship among the SPBR settings, the BRDV[1:0] settings, and bit rates.

Table 28.3 Relationship among SPBR Settings, BRDV[1:0] Settings, and Bit Rates

			Bit Rate	
SPBR (n)	**BRDV[1:0] Bits (N)**	**Division Ratio**	**SERICLK = 120 MHz**	**SERICLK = 150 MHz**
0	0	2	Setting prohibited	Setting prohibited
1	0	4	30.0 Mbps	37.5 Mbps
2	0	6	20.0 Mbps	25.0 Mbps
3	0	8	15.0 Mbps	18.8 Mbps
4	0	10	12.0 Mbps	15.0 Mbps
5	0	12	10.0 Mbps	12.5 Mbps
5	1	24	5.00 Mbps	6.25 Mbps
5	2	48	2.50 Mbps	3.13 Mbps
5	3	96	1.25 Mbps	1.56 Mbps
255	3	4096	29.3 kbps	36.6 kbps

Note: The settings must not exceed the range of electrical characteristics.

28.2.9 RSPI Data Control Register (SPDCR)

SPDCR controls data in the SPDR register.

Up to four frames can be transmitted or received in one round of transmission or reception activation (m = 0 to 7). The amount of data in each transfer is controlled by the combination of the SPCMDm.SPB[3:0] bits, the SPSCR.SPSLN[2:0] bits, and the SPDCR.SPFC[1:0] bits.

When changing the SPDCR.SPFC[1:0] bits while the SPCR.SPE bit is 1, the bits should be changed while the SPSR.IDLNF flag is 0.

Address(es): RSPI0.SPDCR A006 800Bh, RSPI1.SPDCR A006 840Bh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	SPLW	SPRDTD	—	—	SPFC[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	SPFC[1:0]	Number of Frames Specification	b1 b0 0 0: 1 frame 0 1: 2 frames 1 0: 3 frames 1 1: 4 frames	R/W
b3, b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b4	SPRDTD	RSPI Receive/Transmit Data Selection	0: SPDR values are read from the reception buffer 1: SPDR values are read from the transmission buffer (but only if the transmission buffer is empty)	R/W
b5	SPLW	RSPI Longword Access/ Word Access Specification	0: SPDR is accessed in words 1: SPDR is accessed in longwords	R/W
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPFC[1:0] Bits (Number of Frames Specification)

The SPFC[1:0] bits specify the number of frames that can be stored in SPDR (per transfer activation). Up to four frames can be transmitted or received in one round of transmission or reception, and the amount of data is determined by the combination of the SPSCR.SPSLN[2:0] bits, and the SPDCR.SPFC[1:0] bits. Furthermore, the setting of the SPFC[1:0] bits adjusts the number of frames for generation of RSPI reception buffer full interrupt, and start of transmission or generation of transmission buffer empty interrupts. Table 28.4 lists the frame configurations that can be stored in SPDR and examples of combinations of settings for transmission and reception. Combinations of settings other than those shown in the examples should not be made.

Table 28.4 Settable Combinations of SPSLN[2:0] Bits and SPFC[1:0] Bits

Setting	SPSLN[2:0]	SPFC[1:0]	Number of Frames in a Single Sequence	Number of Frames at which Reception Buffer Full Interrupt Occurs or Transmission Buffer Holding Data is Recognized
1-1	000b	00b	1	1
1-2	000b	01b	2	2
1-3	000b	10b	3	3
1-4	000b	11b	4	4
2-1	001b	01b	2	2
2-2	001b	11b	4	4
3	010b	10b	3	3
4	011b	11b	4	4
5	100b	00b	5	1
6	101b	00b	6	1
7	110b	00b	7	1
8	111b	00b	8	1

SPRDTD Bit (RSPI Receive/Transmit Data Selection)

The SPRDTD bit selects whether the SPDR reads values from the reception buffer or from the transmission buffer.
If reading is from the transmission buffer, the value written to SPDR register immediately beforehand is read.
When reading the transmission buffer, do so before writing of the number of frames set in the SPFC[1:0] bits is finished and after generation of the transmission buffer empty interrupt.
For details, refer to section 28.2.5, RSPI Data Register (SPDR).

SPLW Bit (RSPI Longword Access/Word Access Specification)

The SPLW bit specifies the access width for SPDR. Access to SPDR is in words when the SPLW bit is 0 and in longwords when the SPLW bit is 1.
Also, when the SPLW bit is 0, set the SPCMDm.SPB[3:0] bits (RSPI data length setting bits) to 8 to 16 bits. Setting these bits to 20, 24, or 32 bits is prohibited.

28.2.10 RSPI Clock Delay Register (SPCKD)

SPCKD sets a period from the beginning of SSLyi signal assertion to RSPCK oscillation (RSPCK delay) when the SPCMDm.SCKDEN bit is 1. While both the SPCR.MSTR and SPCR.SPE bits are 1, do not change the value of the SPCKD register (m = 0 to 7; y = 0, 1 (for all channels); i = 0 to 3).

Address(es): RSPI0.SPCKD A006 800Ch, RSPI1.SPCKD A006 840Ch

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SCKDL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SCKDL[2:0]	RSPCK Delay Setting	b2 b0 0 0 0: 1 RSPCK 0 0 1: 2 RSPCK 0 1 0: 3 RSPCK 0 1 1: 4 RSPCK 1 0 0: 5 RSPCK 1 0 1: 6 RSPCK 1 1 0: 7 RSPCK 1 1 1: 8 RSPCK	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SCKDL[2:0] Bits (RSPCK Delay Setting)

The SCKDL[2:0] bits set an RSPCK delay value when the SPCMDm.SCKDEN bit is 1.
When using the RSPI in slave mode, set the SCKDL[2:0] bits to 000b.

28.2.11 RSPI Slave Select Negation Delay Register (SSLND)

SSLND sets a period (SSL negation delay) from the transmission of a final RSPCK edge to the negation of the SSLyi signal during a serial transfer by the RSPI in master mode. Writing to SSLND is prohibited while both the SPCR.MSTR and SPCR.SPE bits are 1 (y = 0, 1 (for all channels); i = 0 to 3).

Address(es): RSPI0.SSLND A006 800Dh, RSPI1.SSLND A006 840Dh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SLNDL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SLNDL[2:0]	SSL Negation Delay Setting	b2 b0 0 0 0: 1 RSPCK 0 0 1: 2 RSPCK 0 1 0: 3 RSPCK 0 1 1: 4 RSPCK 1 0 0: 5 RSPCK 1 0 1: 6 RSPCK 1 1 0: 7 RSPCK 1 1 1: 8 RSPCK	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SLNDL[2:0] Bits (SSL Negation Delay Setting)

The SLNDL[2:0] bits set an SSL negation delay value when the RSPI is in master mode.
When using the RSPI in slave mode, set the SLNDL[2:0] bits to 000b.

28.2.12 RSPI Next-Access Delay Register (SPND)

SPND sets a non-active period (next-access delay) of the SSLyi signal after termination of a serial transfer when the SPCMDm.SPNDEN bit is 1. Writing to SPND is prohibited while both the SPCR.MSTR and SPCR.SPE bits are 1 (m = 0 to 7, y = 0, 1 (for all channels); i = 0 to 3).

Address(es): RSPI0.SPND A006 800Eh, RSPI1.SPND A006 840Eh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SPNDL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SPNDL[2:0]	RSPI Next-Access Delay Setting	b2 b0 0 0 0: 1 RSPCK + 2 SERICLK 0 0 1: 2 RSPCK + 2 SERICLK 0 1 0: 3 RSPCK + 2 SERICLK 0 1 1: 4 RSPCK + 2 SERICLK 1 0 0: 5 RSPCK + 2 SERICLK 1 0 1: 6 RSPCK + 2 SERICLK 1 1 0: 7 RSPCK + 2 SERICLK 1 1 1: 8 RSPCK + 2 SERICLK	R/W
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPNDL[2:0] Bits (RSPI Next-Access Delay Setting)

The SPNDL[2:0] bits set a next-access delay when the SPCMDm.SPNDEN bit is 1.

When using the RSPI in slave mode, set the SPNDL[2:0] bits to 000b.

28.2.13 RSPI Control Register 2 (SPCR2)

The SPCR2 register controls the settings for operation of the RSPI.
Changing the value of SPPE, SPOE, or SCKASE bit in the SPCR2 register is prohibited while the setting of the SPCR.SPE bit is 1.

Address(es): RSPI0.SPCR2 A006 800Fh, RSPI1.SPCR2 A006 840Fh

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	SCKASE	PTE	SPIIE	SPOE	SPPE
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SPPE	Parity Enable	0: Does not add the parity bit to transmit data and does not check the parity bit of receive data 1: Adds the parity bit to transmit data and checks the parity bit of receive data (when SPCR.TXMD = 0) Adds the parity bit to transmit data but does not check the parity bit of receive data (when SPCR.TXMD = 1)	R/W
b1	SPOE	Parity Mode	0: Selects even parity for use in transmission and reception 1: Selects odd parity for use in transmission and reception	R/W
b2	SPIIE	RSPI Idle Interrupt Enable	0: Disables the generation of idle interrupt requests 1: Enables the generation of idle interrupt requests	R/W
b3	PTE	Parity Self-Testing	0: Disables the self-diagnosis function of the parity circuit 1: Enables the self-diagnosis function of the parity circuit	R/W
b4	SCKASE	RSPCK Auto-Stop Function Enable	0: Disables the RSPCK auto-stop function 1: Enables the RSPCK auto-stop function	R/W
b7 to b5	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

SPPE Bit (Parity Enable)

The SPPE bit enables or disables the parity function.
The parity bit is added to transmit data and parity checking is performed for receive data when the SPCR.TXMD bit is 0 and the SPCR2.SPPE bit is 1.
The parity bit is added to transmit data but parity checking is not performed for receive data when the SPCR.TXMD bit is 1 and the SPCR2.SPPE bit is 1.

SPOE Bit (Parity Mode)

The SPOE bit specifies odd or even parity.
When even parity is set, parity bit addition is performed so that the total number of 1-bits in the transmit/receive character plus the parity bit is even. Similarly, when odd parity is set, parity bit addition is performed so that the total number of 1-bits in the transmit/receive character plus the parity bit is odd.
The SPOE bit is valid only when the SPPE bit is 1.

SPIIE Bit (RSPI Idle Interrupt Enable)

The SPIIE bit enables or disables the generation of RSPI idle interrupt requests when the RSPI being in the idle state is detected and the SPSR.IDLNF flag is set to 0.

PTE Bit (Parity Self-Testing)

The PTE bit enables the self-diagnosis function of the parity circuit in order to check whether the parity function is operating correctly.

SCKASE Bit (RSPCK Auto-Stop Function Enable)

The SCKASE bit enables or disables the RSPCK auto-stop function. When this function is enabled, the RSPCK clock is stopped before an overrun error occurs when data is received in master mode. For details, see section 28.3.8.1, Overrun Error.

28.2.14 RSPI Command Registers 0 to 7 (SPCMD0 to SPCMD7)

SPCMDm registers control a transfer format for the RSPI in master mode. Each channel has eight RSPI command registers (m = 0 to 7).

Some of the bits in the SPCMD0 register is used to set a transfer format for the RSPI in slave mode. The RSPI in master mode refers to SPCMDm registers in sequence according to the settings in the SPSCR.SPSLN[2:0] bits, and executes the serial transfer that is set in respective SPCMDm registers.

SPCMDm registers should be set while the transmission buffers are empty (data for the next transfer is not set) and before setting of the data that are to be transmitted as a result of reference to the SPCMDm registers.

SPCMDm currently being referred to by the RSPI in master mode can be checked by means of the SPSSR.SPCP[2:0] bits. Writing to SPCMDm is prohibited while the SPCR.MSTR bit is 0 and the SPCR.SPE bit is 1.

Address(es): RSPI0.SPCMD0 A006 8010h, RSPI0.SPCMD1 A006 8012h, RSPI0.SPCMD2 A006 8014h, RSPI0.SPCMD3 A006 8016h, RSPI0.SPCMD4 A006 8018h, RSPI0.SPCMD5 A006 801Ah, RSPI0.SPCMD6 A006 801Ch, RSPI0.SPCMD7 A006 801Eh, RSPI1.SPCMD0 A006 8410h, RSPI1.SPCMD1 A006 8412h, RSPI1.SPCMD2 A006 8414h, RSPI1.SPCMD3 A006 8416h, RSPI1.SPCMD4 A006 8418h, RSPI1.SPCMD5 A006 841Ah, RSPI1.SPCMD6 A006 841Ch, RSPI1.SPCMD7 A006 841Eh

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB[3:0]				SSLKP	SSLy[2:0]			BRDV[1:0]		CPOL	CPHA
Value after reset:	0	0	0	0	0	1	1	1	0	0	0	0	1	1	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	CPHA	RSPCK Phase Setting	0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge	R/W
b1	CPOL	RSPCK Polarity Setting	0: RSPCK is low when idle 1: RSPCK is high when idle	R/W
b3, b2	BRDV[1:0]	Bit Rate Division Setting	b3 b2 0 0: These bits select the base bit rate 0 1: These bits select the base bit rate divided by 2 1 0: These bits select the base bit rate divided by 4 1 1: These bits select the base bit rate divided by 8	R/W
b6 to b4	SSLy[2:0]	SSL Signal Assertion Setting	b6 b4 0 0 0: SSLy0 (y = 0, 1) 0 0 1: SSLy1 (y = 0, 1) 0 1 0: SSL02 0 1 1: SSL03 Settings other than above are prohibited.	R/W
b7	SSLKP	SSL Signal Level Keeping	0: Negates all SSL signals upon completion of transfer 1: Keeps the SSL signal level from the end of transfer until the beginning of the next access	R/W
b11 to b8	SPB[3:0]	RSPI Data Length Setting	b11 b8 0100 to 0111: 8 bits 1 0 0 0: 9 bits 1 0 0 1: 10 bits 1 0 1 0: 11 bits 1 0 1 1: 12 bits 1 1 0 0: 13 bits 1 1 0 1: 14 bits 1 1 1 0: 15 bits 1 1 1 1: 16 bits 0 0 0 0: 20 bits 0 0 0 1: 24 bits 0010, 0011: 32 bits	R/W
b12	LSBF	RSPI LSB First	0: MSB first 1: LSB first	R/W
b13	SPNDEN	RSPI Next-Access Delay Enable	0: A next-access delay of 1 RSPCK + 2 SERICLK 1: A next-access delay is equal to the setting of the RSPI next-access delay register (SPND)	R/W

Bit	Symbol	Bit Name	Description	R/W
b14	SLNDEN	SSL Negation Delay Setting Enable	0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay is equal to the setting of the RSPI slave select negation delay register (SSLND)	R/W
b15	SCKDEN	RSPCK Delay Setting Enable	0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay is equal to the setting of the RSPI clock delay register (SPCKD)	R/W

CPHA Bit (RSPCK Phase Setting)

The CPHA bit sets an RSPCK phase of the RSPI in master mode or slave mode. Data communications between RSPI modules require the same RSPCK phase setting between the modules.

CPOL Bit (RSPCK Polarity Setting)

The CPOL bit sets an RSPCK polarity of the RSPI in master mode or slave mode. Data communications between RSPI modules require the same RSPCK polarity setting between the modules.

BRDV[1:0] Bits (Bit Rate Division Setting)

The BRDV[1:0] bits are used to determine the bit rate. A bit rate is determined by combinations of the settings in the BRDV[1:0] bits and SPBR (refer to section 28.2.8, RSPI Bit Rate Register (SPBR)). The settings in SPBR determine the base bit rate. The settings in the BRDV[1:0] bits are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In SPCMDm register, different BRDV[1:0] bit settings can be specified. This permits the execution of serial transfers at a different bit rate for each command (m = 0 to 7).

SSLy[2:0] Bits (SSL Signal Assertion Setting)

The SSLy[2:0] bits control the SSLyi signal assertion when the RSPI performs serial transfers in master mode.
Setting the SSLy[2:0] bits controls the assertion for the SSLyi signal. When an SSLyi signal is asserted, its polarity is determined by the set value in the corresponding SSLP. When the SSLy[2:0] bits are set to 000b in multi-master mode, serial transfers are performed with all the SSL signals in the negated state (as the SSLy0 pin acts as input).
When using the RSPI in slave mode, set the SSLy[2:0] bits to 000b.
(y = 0, 1 (for all channels); i = 0 to 3)

SSLKP Bit (SSL Signal Level Keeping)

When the RSPI in master mode performs a serial transfer, the SSLKP bit specifies whether the SSLyi signal level for the current command is to be kept or negated between the SSL negation timing associated with the current command and the SSL assertion timing associated with the next command.
Setting the SSLKP bit to 1 enables a burst transfer. For details, refer to section 28.3.10.1, (4) Burst Transfer.
When using the RSPI in slave mode, the SSLKP bit should be set to 0.
(y = 0, 1 (for all channels); i = 0 to 3)

SPB[3:0] Bits (RSPI Data Length Setting)

The SPB[3:0] bits set a transfer data length for the RSPI in master mode or slave mode.

LSBF Bit (RSPI LSB First)

The LSBF bit sets the data format of the RSPI in master mode or slave mode to MSB first or LSB first.

SPNDEN Bit (RSPI Next-Access Delay Enable)

The SPNDEN bit sets the period from the time the RSPI in master mode terminates a serial transfer and sets the SSLyi signal inactive until the RSPI enables the SSLyi signal assertion for the next access (next-access delay) (y = 0, 1 (for all channels); i = 0 to 3). If the SPNDEN bit is 0, the RSPI sets the next-access delay to 1 RSPCK + 2 SERICLK. If the SPNDEN bit is 1, the RSPI inserts a next-access delay in compliance with the SPND setting.
When using the RSPI in slave mode, the SPNDEN bit should be set to 0.

SLNDEN Bit (SSL Negation Delay Setting Enable)

The SLNDEN bit sets the period from the time the RSPI in master mode stops RSPCK oscillation until the RSPI sets the SSLyi signal inactive (SSL negation delay). If the SLNDEN bit is 0, the RSPI sets the SSL negation delay to 1 RSPCK. If the SLNDEN bit is 1, the RSPI negates the SSL signal at an SSL negation delay in compliance with the SSLND setting.
When using the RSPI in slave mode, the SLNDEN bit should be set to 0.

SCKDEN Bit (RSPCK Delay Setting Enable)

The SCKDEN bit sets the period from the point when the RSPI in master mode activates the SSLyi signal until the RSPCK starts oscillation (RSPI clock delay). If the SCKDEN bit is 0, the RSPI sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is 1, the RSPI starts the oscillation of RSPCK at an RSPCK delay in compliance with the SPCKD setting.
When using the RSPI in slave mode, the SCKDEN bit should be set to 0.

28.3 Operation

In this section, the serial transfer period means a period from the beginning of driving valid data to the fetching of the final valid data.

28.3.1 Overview of RSPI Operations

The RSPI is capable of synchronous serial transfers in slave mode (SPI operation), single-master mode (SPI operation), multi-master mode (SPI operation), slave mode (clock synchronous operation), and master mode (clock synchronous operation). A particular mode of the RSPI can be selected by using the MSTR, MODFEN, and SPMS bits in SPCR. Table 28.5 lists the relationship between RSPI modes and SPCR settings, and a description of each mode.

Table 28.5 Relationship between RSPI Modes and SPCR Settings and Description of Each Mode

Mode	Slave (SPI Operation)	Single-Master (SPI Operation)	Multi-Master (SPI Operation)	Slave (Clock Synchronous Operation)	Master (Clock Synchronous Operation)
MSTR bit setting	0	1	1	0	1
MODFEN bit setting	0 or 1	0	1	0	0
SPMS bit setting	0	0	0	1	1
RSPCKy signal	Input	Output	Output/Hi-Z	Input	Output
MOSIy signal	Input	Output	Output/Hi-Z	Input	Output
MISOy signal	Output/Hi-Z	Input	Input	Output	Input
SSLy0 signal	Input	Output	Input	Hi-Z*1	Hi-Z*1
SSLy1 to SSLy3 signals	Hi-Z*1	Output	Output/Hi-Z	Hi-Z*1	Hi-Z*1
SSL polarity modification function	Supported	Supported	Supported	—	—
Transfer rate	Up to SERICLK/8	Up to SERICLK/4	Up to SERICLK/4	Up to SERICLK/8	Up to SERICLK/4
Clock source	RSPCK input	On-chip baud rate generator	On-chip baud rate generator	RSPCK input	On-chip baud rate generator
Clock polarity	Two				
Clock phase	Two	Two	Two	One (CPHA = 1)	Two
First transfer bit	MSB/LSB				
Transfer data length	8 to 16, 20, 24, 32 bits				
Burst transfer	Possible (CPHA = 1)	Possible (CPHA = 0,1)	Possible (CPHA = 0,1)	—	—
RSPCK delay control	Not supported	Supported	Supported	Not supported	Supported
SSL negation delay control	Not supported	Supported	Supported	Not supported	Supported
Next-access delay control	Not supported	Supported	Supported	Not supported	Supported
Transfer activation method	SSL input active or RSPCK oscillation	Transmission buffer is written to at generation of a transmission buffer empty interrupt request	Transmission buffer is written to at generation of a transmission buffer empty interrupt request	RSPCK oscillation	Transmission buffer is written to at generation of a transmission buffer empty interrupt request
Sequence control	Not supported	Supported	Supported	Not supported	Supported
Transmission buffer empty detection	Supported				
Reception buffer full detection	Supported*2				
Overrun error detection	Supported*2	Supported*2, *4	Supported*2, *4	Supported*2	Supported*2
Parity error detection	Supported*2,*3				
Mode fault error detection	Supported (MODFEN = 1)	Not supported	Supported	Not supported	Not supported

Note 1. This function is not supported in this mode.
Note 2. When the SPCR.TXMD bit is 1, receiver buffer full detection, overrun error detection, and parity error detection are not performed.
Note 3. When the SPCR2.SPPE bit is 0, parity error detection is not performed.
Note 4. When the SPCR2.SCKASE bit is 1, overrun error detection does not proceed.
y = 0, 1 (for all channels)

28.3.2 Controlling RSPI Pins

According to the SPCR.MSTR, MODFEN, SPMS, and SPPCR.SPOM bits, the RSPI can switch pin states. Table 28.6 lists the relationship between pin states and bit settings. Setting the SPPCR.SPOM bit to 0 selects CMOS output; setting it to 1 selects open-drain output. The I/O port settings should follow this relationship.

Table 28.6 Relationship between Pin States and Bit Settings

		Pin State*2	
Mode	**Pin**	**SPOM = 0**	**SPOM = 1**
Single-master mode (SPI operation) (MSTR = 1, MODFEN = 0, SPMS = 0)	RSPCKy	CMOS output	Open-drain output
	SSLy0 to SSLy3	CMOS output	Open-drain output
	MOSIy	CMOS output	Open-drain output
	MISOy	Input	Input
Multi-master mode (SPI operation) (MSTR = 1, MODFEN = 1, SPMS = 0)	RSPCKy*3	CMOS output/Hi-Z	Open-drain output/Hi-Z
	SSLy0	Input	Input
	SSLy1 to SSLy3*3	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MOSIy*3	CMOS output/Hi-Z	Open-drain output/Hi-Z
	MISOy	Input	Input
Slave mode (SPI operation) (MSTR = 0, SPMS = 0)	RSPCKy	Input	Input
	SSLy0	Input	Input
	SSLy1 to SSLy3*5	Hi-Z*1	Hi-Z*1
	MOSIy	Input	Input
	MISOy*4	CMOS output/Hi-Z	Open-drain output/Hi-Z
Master mode (Clock synchronous operation) (MSTR = 1, MODFEN = 0, SPMS = 1)	RSPCKy	CMOS output	Open-drain output
	SSLy0 to SSLy3*5	Hi-Z*1	Hi-Z*1
	MOSIy	CMOS output	Open-drain output
	MISOy	Input	Input
Slave mode (Clock synchronous operation) (MSTR = 0, SPMS = 1)	RSPCKy	Input	Input
	SSLy0 to SSLy3*5	Hi-Z*1	Hi-Z*1
	MOSIy	Input	Input
	MISOy	CMOS output	Open-drain output

Note 1. This function is not supported in this mode.
Note 2. RSPI settings are not reflected in the multiplex pins for which the RSPI function is not selected.
Note 3. When SSLy0 is at the active level, the pin state is Hi-Z.
Note 4. When SSLy0 is at the non-active level or the SPCR.SPE bit is 0, the pin state is Hi-Z.
Note 5. These pins are available for use as I/O port pins.
y = 0, 1 (for all channels)

The RSPI in single-master mode (SPI operation) or multi-master mode (SPI operation) determines MOSI signal values during the SSL negation period (including the SSL retention period during a burst transfer) according to MOIFE and MOIFV bit settings in SPPCR, as listed in Table 28.7.

Table 28.7 MOSI Signal Value Determination during SSL Negation Period

MOIFE Bit	MOIFV Bit	MOSIy Signal Value during SSL Negation Period
0	0, 1	Final data from previous transfer
1	0	Always low
1	1	Always high

Note: During the period of SSL negation, the RSPI function must be enabled (i.e. SPCR.SPE bit = 1) for control over the levels of the MOSIy signals.

28.3.3 RSPI System Configuration Examples

28.3.3.1 Single Master/Single Slave (with This LSI Acting as Master)

Figure 28.5 shows a single-master/single-slave RSPI system configuration example when this LSI is used as a master. In the single-master/single-slave configuration, the SSLy0 to SSLy3 output of this LSI (master) are not used. The SSL input of the SPI slave is fixed to the low level, and the SPI slave is always maintained in a select state.*1
This LSI (master) always drives RSPCKy and MOSIy. The SPI slave always drives the MISO.
(y = 0, 1 (for all channels))

Note 1. In the transfer format corresponding to the case where the SPCMDm.CPHA bit is 0, there are slave devices for which the SSL signal cannot be fixed to the active level. In situations where the SSL signal cannot be fixed, the SSLyi output of this LSI should be connected to the SSL input of the slave device.

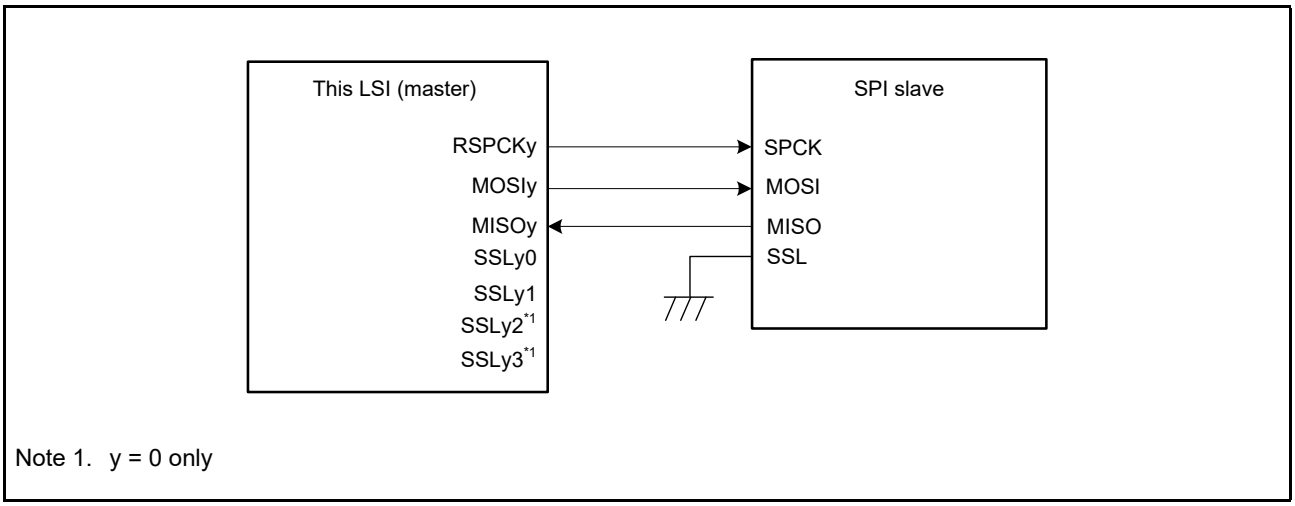


Figure 28.5 Single-Master/Single-Slave Configuration Example (This LSI = Master)

28.3.3.2 Single Master/Single Slave (with This LSI Acting as Slave)

Figure 28.6 shows a single-master/single-slave RSPI system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave, the SSLy0 pin is used as SSL input. The SPI master always drives the RSPCK and MOSI. This LSI (slave) always drives the MISOy.*1

In the single-slave configuration in which the SPCMDm.CPHA bit is set to 1, the SSLy0 input of this LSI (slave) is fixed to the low level, this LSI (slave) is always maintained in a select state, and in this manner it is possible to execute serial transfer (Figure 28.7).

(y = 0, 1 (for all channels))

Note 1. When SSLy0 is at the non-active level, the pin state is Hi-Z.

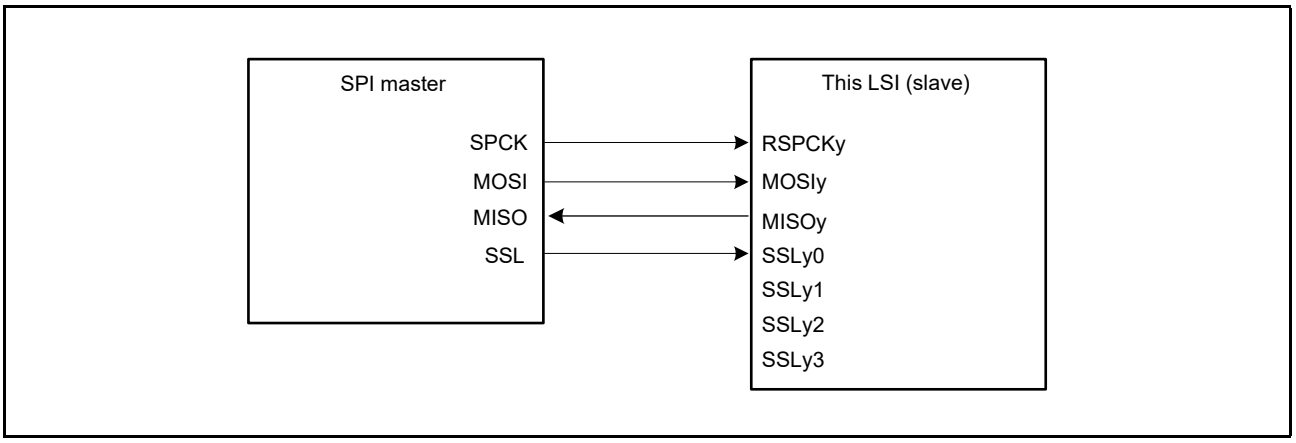


Figure 28.6 Single-Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 0)

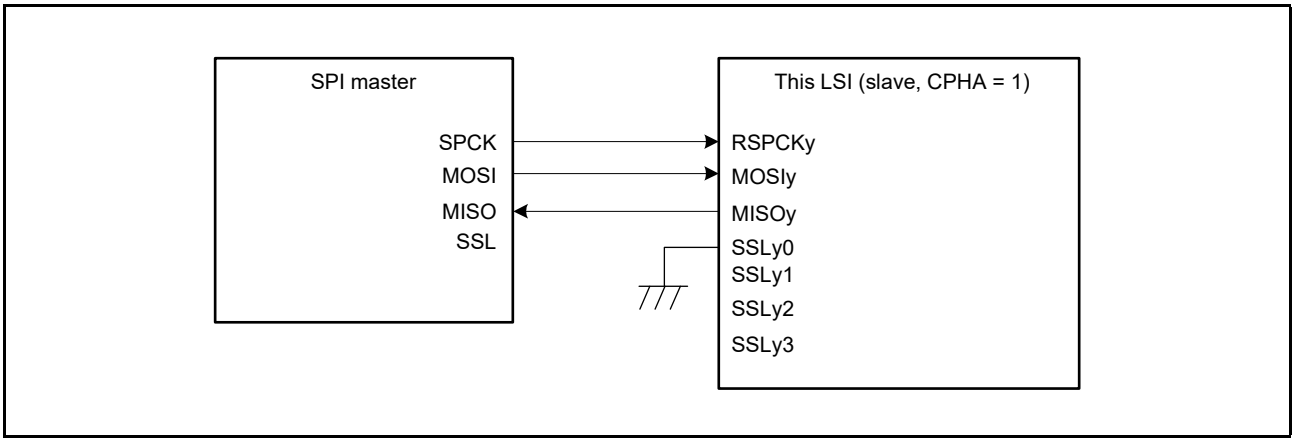


Figure 28.7 Single-Master/Single-Slave Configuration Example (This LSI = Slave, CPHA = 1)

28.3.3.3 Single Master/Multi-Slave (with This LSI Acting as Master)

Figure 28.8 shows a single-master/multi-slave RSPI system configuration example when this LSI is used as a master. In the example of Figure 28.8, the RSPI system is comprised of this LSI (master) and four slaves (SPI slave 0 to SPI slave 3).

The RSPCKy and MOSIy outputs of this LSI (master) are connected to the RSPCK and MOSI inputs of SPI slave 0 to SPI slave 3. The MISO outputs of SPI slave 0 to SPI slave 3 are all connected to the MISOy input of this LSI (master). SSLy0 to SSLy3 outputs of this LSI (master) are connected to the SSL inputs of SPI slave 0 to SPI slave 3, respectively. This LSI (master) always drives RSPCK, MOSI, and SSLy0 to SSLy3. Of the SPI slave 0 to SPI slave 3, the slave that receives low-level input into the SSL input drives MISO.

(y = 0, 1 (for all channels))

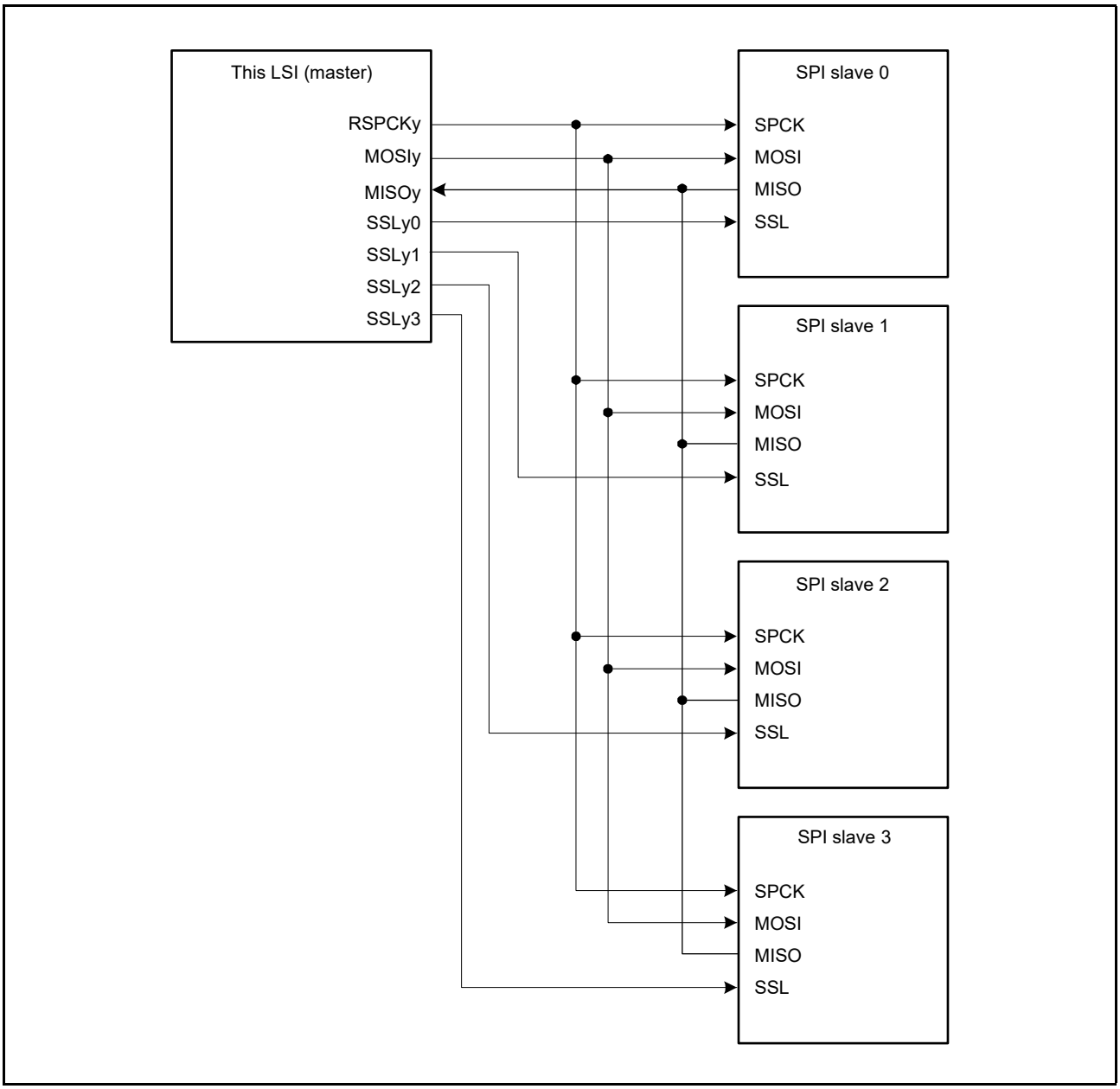


Figure 28.8 Single-Master/Multi-Slave Configuration Example (This LSI = Master)

28.3.3.4 Single Master/Multi-Slave (with This LSI Acting as Slave)

Figure 28.9 shows a single-master/multi-slave RSPI system configuration example when this LSI is used as a slave. In the example of Figure 28.9, the RSPI system is comprised of an SPI master and two LSIs (slave X and slave Y).
The SPCK and MOSI outputs of the SPI master are connected to the RSPCKy and MOSIy inputs of the LSIs (slave X and slave Y). The MISOy outputs of the LSIs (slave X and slave Y) are all connected to the MISO input of the SPI master. SSLX and SSLY outputs of the SPI master are connected to the SSLy0 inputs of the LSIs (slave X and slave Y), respectively.
The SPI master always drives SPCK, MOSI, SSLX, and SSLY. Of the LSIs (slave X and slave Y), the slave that receives low-level input into the SSLy0 input drives MISOy.
(y = 0, 1 (for all channels))

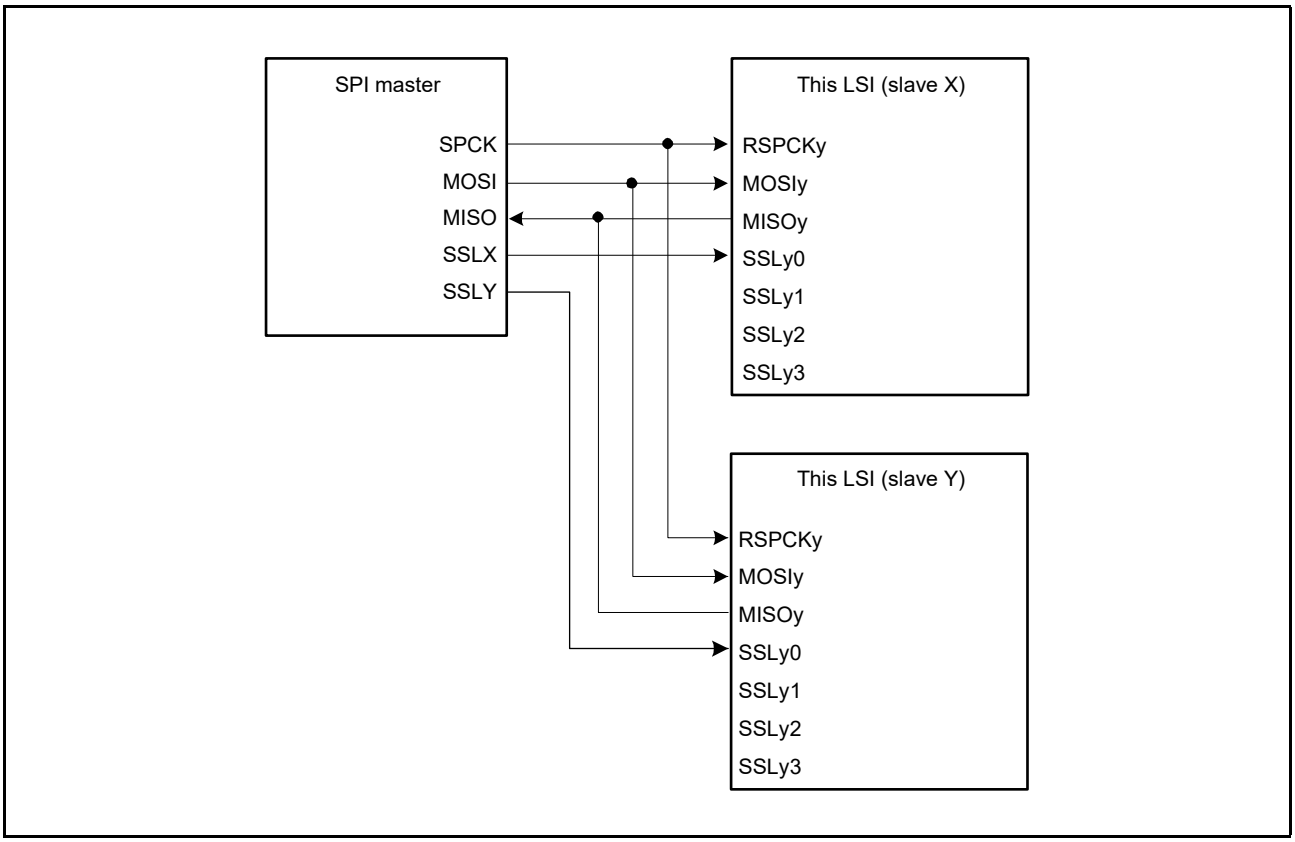


Figure 28.9 Single-Master/Multi-Slave Configuration Example (This LSI = Slave)

28.3.3.5 Multi-Master/Multi-Slave (with This LSI Acting as Master)

Figure 28.10 shows a multi-master/multi-slave RSPI system configuration example when this LSI is used as a master. In the example of Figure 28.10, the RSPI system is comprised of two LSIs (master X and master Y) and two SPI slaves (SPI slave 1 and SPI slave 2).

The RSPCKy and MOSIy outputs of the LSIs (master X and master Y) are connected to the RSPCK and MOSI inputs of SPI slaves 1 and 2. The MISO outputs of SPI slaves 1 and 2 are connected to the MISOy inputs of the LSIs (master X and master Y). Any generic port Y output from this LSI (master X) is connected to the SSLy0 input of this LSI (master Y). Any generic port X output of this LSI (master Y) is connected to the SSLy0 input of this LSI (master X). The SSLy1 and SSLy2 outputs of the LSIs (master X and master Y) are connected to the SSL inputs of the SPI slaves 1 and 2. In this configuration example, since the system can be comprised solely of SSLy0 input, and SSLy1 and SSLy2 outputs for slave connections, the SSLy3 output of this LSI is not required.

This LSI drives RSPCKy, MOSIy, SSLy1, and SSLy2 when the SSLy0 input level is high. When the SSLy0 input level is low, this LSI detects a mode fault error, sets RSPCKy, MOSIy, SSLy1, and SSLy2 to Hi-Z, and releases the RSPI bus right to the other master. Of the SPI slaves 1 and 2, the slave that receives low-level input into the SSL input drives MISO.

(y = 0, 1 (for all channels))

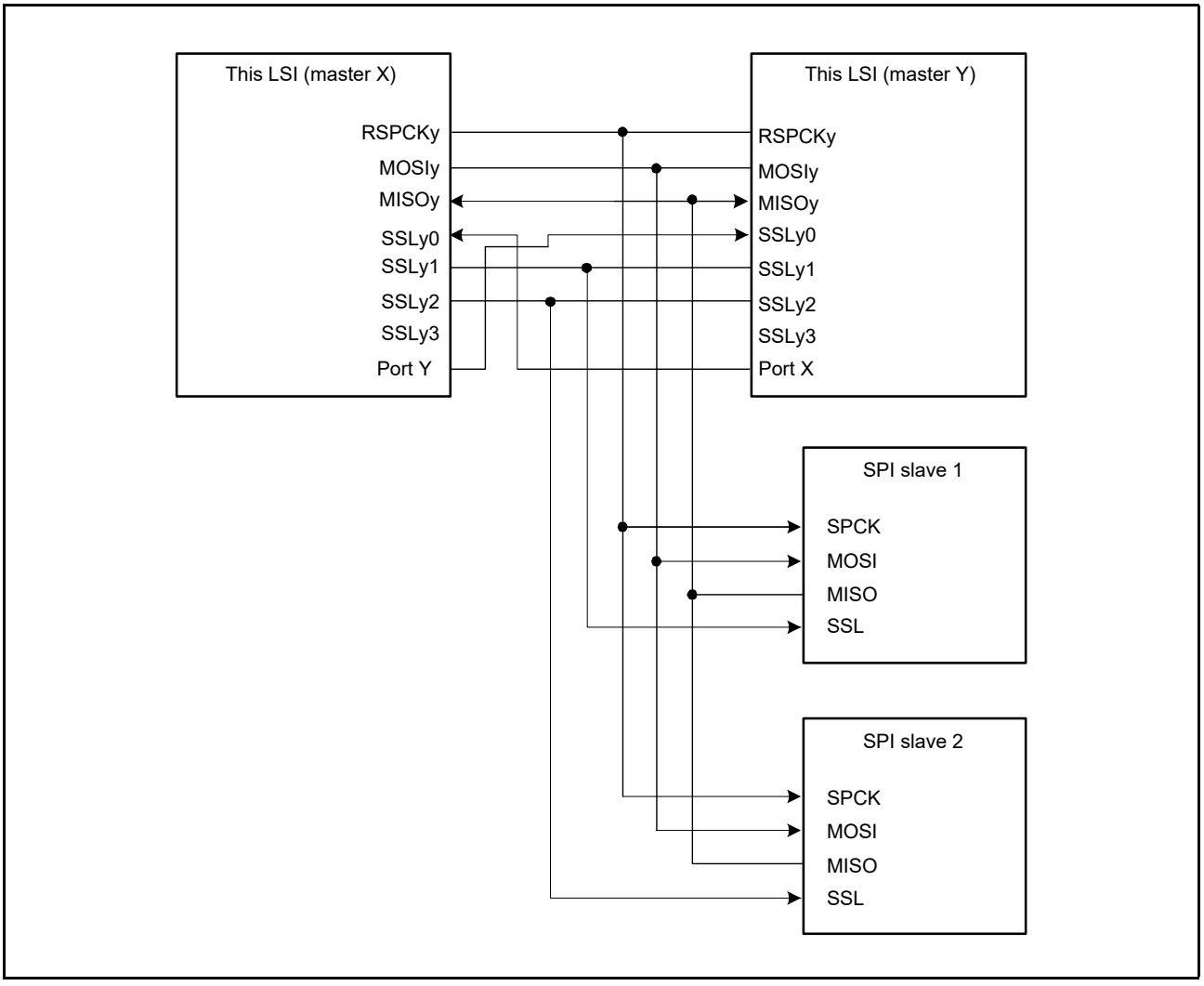


Figure 28.10 Multi-Master/Multi-Slave Configuration Example (This LSI = Master)

28.3.3.6 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) (with This LSI Acting as Master)

Figure 28.11 shows a master (clock synchronous operation)/slave (clock synchronous operation) RSPI system configuration example when this LSI is used as a master. In the master (clock synchronous operation)/slave (clock synchronous operation) configuration, SSLy0 to SSLy3 of this LSI (master) are not used.
This LSI (master) always drives the RSPCKy and MOSIy. The SPI slave always drives the MISO.
(y = 0, 1 (for all channels))

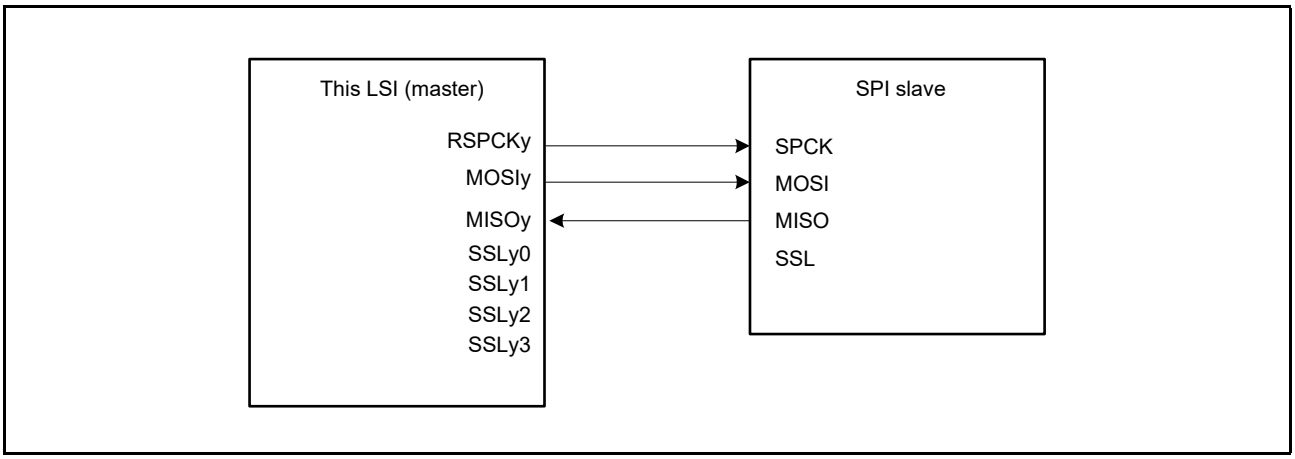


Figure 28.11 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) Configuration Example (This LSI = Master)

28.3.3.7 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) (with This LSI Acting as Slave)

Figure 28.12 shows a master (clock synchronous operation)/slave (clock synchronous operation) RSPI system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave (clock synchronous operation), this LSI (slave) always drives the MISOy and the SPI master always drives the SPCK and MOSI. In addition, SSLy0 to SSLy3 of this LSI (slave) are not used.
Only in the single-slave configuration in which the SPCMDm.CPHA bit is set to 1, this LSI (slave) can execute serial transfer.
(y = 0, 1 (for all channels))

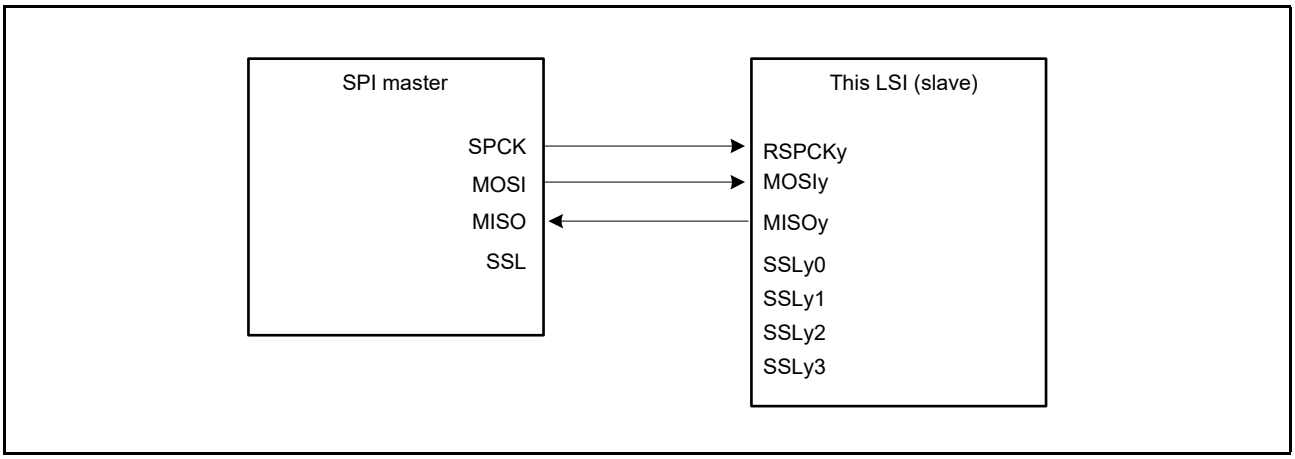


Figure 28.12 Master (Clock Synchronous Operation)/Slave (Clock Synchronous Operation) Configuration Example (This LSI = Slave, CPHA = 1)

28.3.4 Data Format

The RSPI's data format depends on the settings in RSPI command register m (SPCMDm) (m = 0 to 7) and the parity enable bit of RSPI control register 2 (SPCR2.SPPE). Regardless of whether the MSB or LSB is first, the RSPI treats the range from the LSB bit of the RSPI data register (SPDR) to the selected data length as transfer data.
The format of one frame of data before or after transfer is shown below.

(a) With Parity Disabled

When parity is disabled, transmission or reception of data proceeds with the length in bits selected in the RSPI data length setting bits in RSPI command register m (SPCMDm.SPB[3:0]).

(b) With Parity Enabled

When parity is enabled, transmission or reception of data proceeds with the length in bits selected in the RSPI data length setting bits in RSPI command register m (SPCMDm.SPB[3:0]). In this case, however, the last bit is a parity bit.

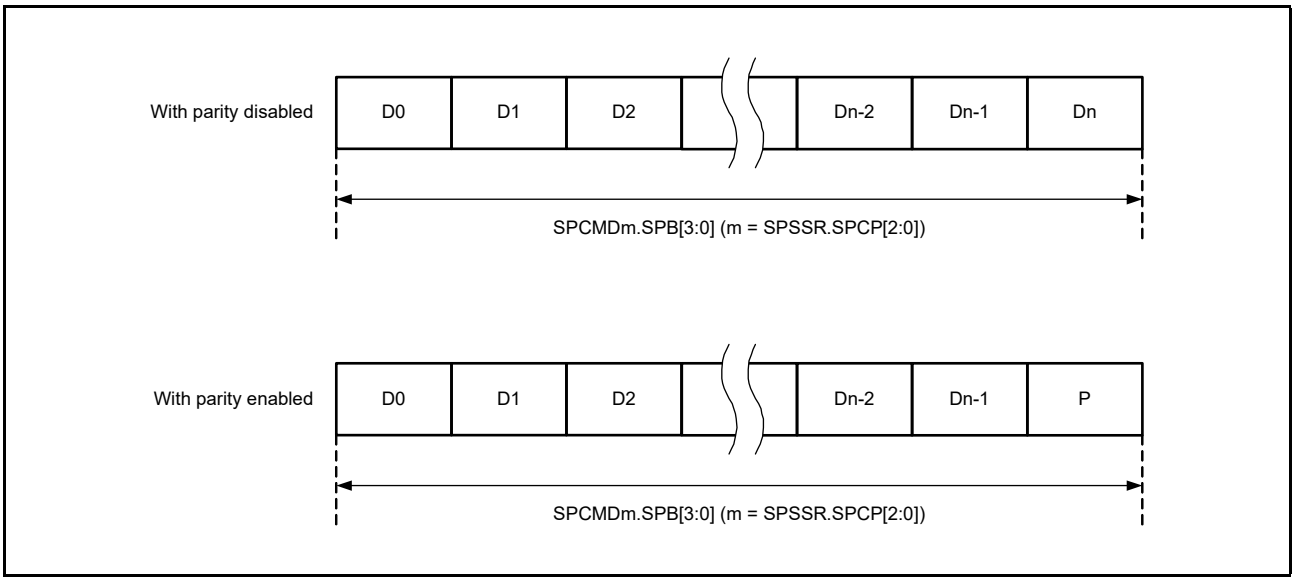


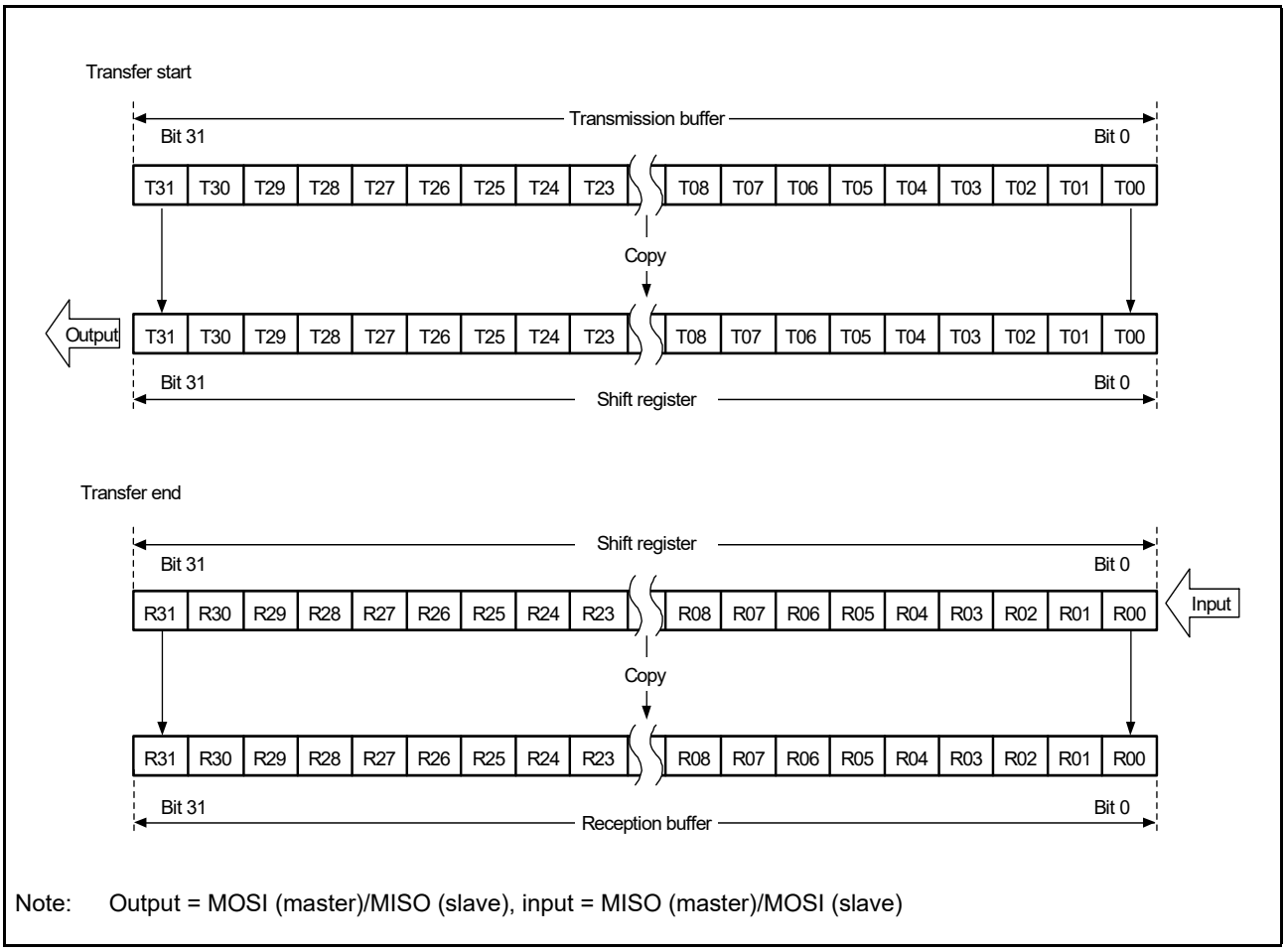
Figure 28.13 Outline of the Data Format (with Parity Disabled/Enabled)

28.3.4.1 When Parity is Disabled (SPCR2.SPPE = 0)

When parity is disabled, data for transmission are copied to the shift register with no prior processing. A description of the connection between the RSPI data register (SPDR) and the shift register in terms of the combination of MSB or LSB first and data length is given below.

(1) MSB First Transfer (32-Bit Data)

Figure 28.14 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity disabled, an RSPI data length of 32 bits, and MSB first selected.
In transmission, bits T31 to T00 from the current stage of the transmission buffer are copied to the shift register. Data for transmission are shifted out from the shift register in order from T31, through T30, and so on to T00.
In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R31 to R00 have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer.

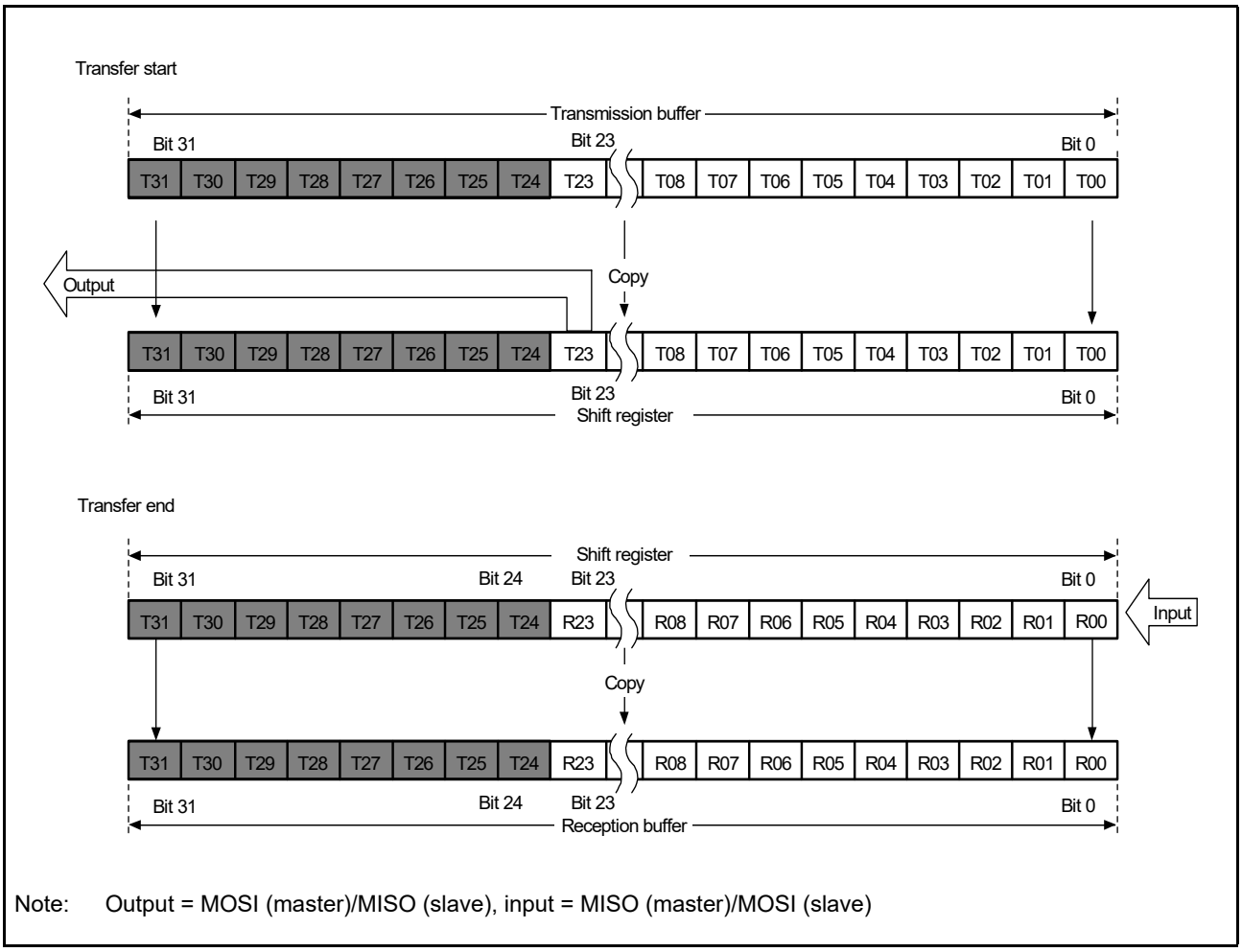


Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 28.14 MSB First Transfer (32-Bit Data, Parity Disabled)

(2) MSB First Transfer (24-Bit Data)

Figure 28.15 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity disabled, 24 bits as the RSPI data length for an example that is not 32 bits, and MSB first selected.
In transmission, the lower-order 24 bits (T23 to T00) from the current stage of the transmission buffer are copied to the shift register. Data for transmission are shifted out from the shift register in order from T23, through T22, and so on to T00.
In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R23 to R00 have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. At this time, the higher-order 8 bits of the transmission buffer are stored in the higher-order 8 bits of the reception buffer. Writing 0 to bits T31 to T24 at the time of transmission leads to 0 being inserted in the higher-order 8 bits of the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

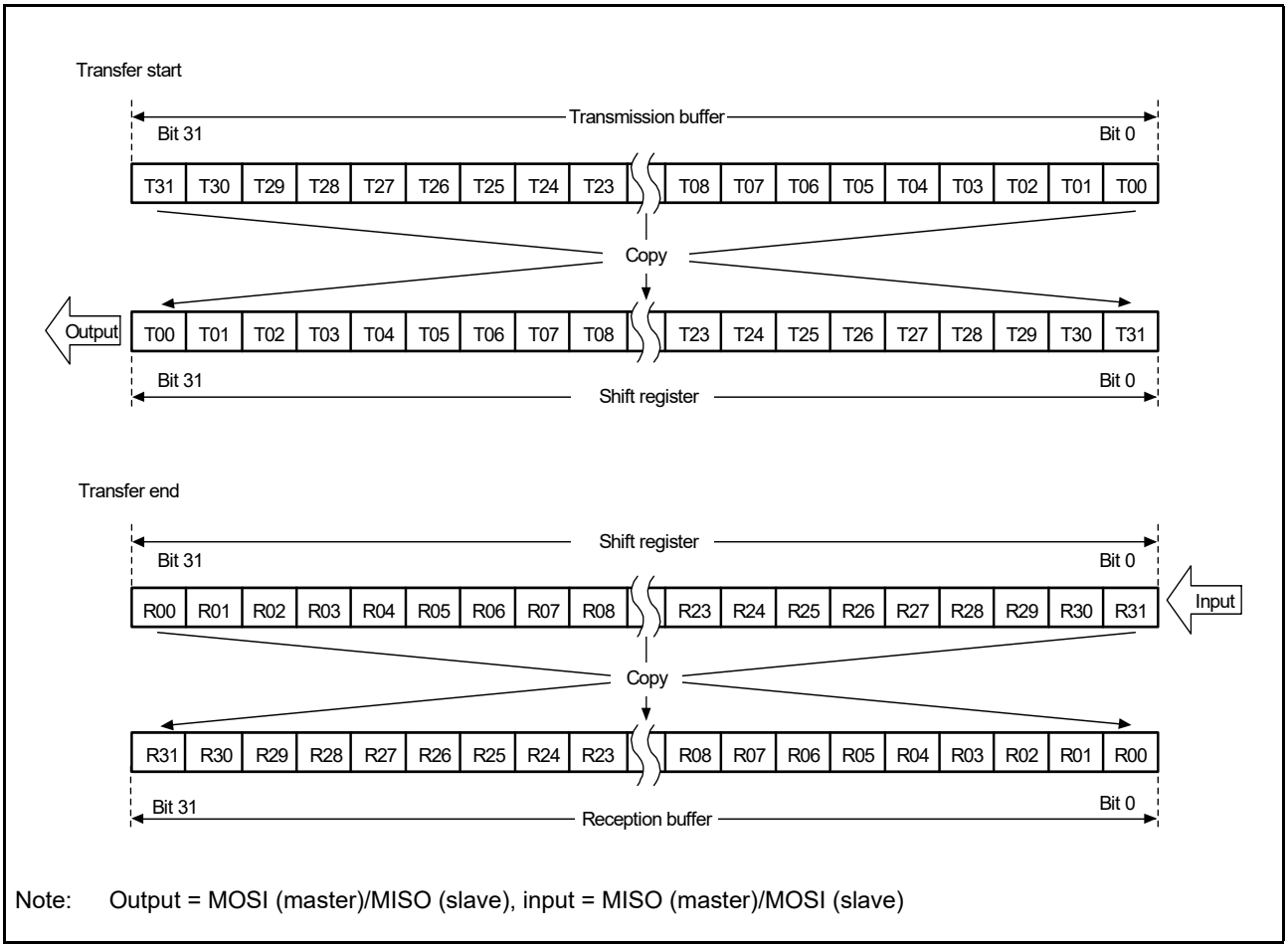
Figure 28.15 MSB First Transfer (24-Bit Data, Parity Disabled)

(3) LSB First Transfer (32-Bit Data)

Figure 28.16 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity disabled, an RSPI data length of 32 bits, and LSB first selected.

In transmission, bits T31 to T00 from the current stage of the transmission buffer are reordered bit by bit to obtain the order T00 to T31 for copying to the shift register. Data for transmission are shifted out from the shift register in order from T00, through T01, and so on to T31.

In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R00 to R31 have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 28.16 LSB First Transfer (32-Bit Data, Parity Disabled)

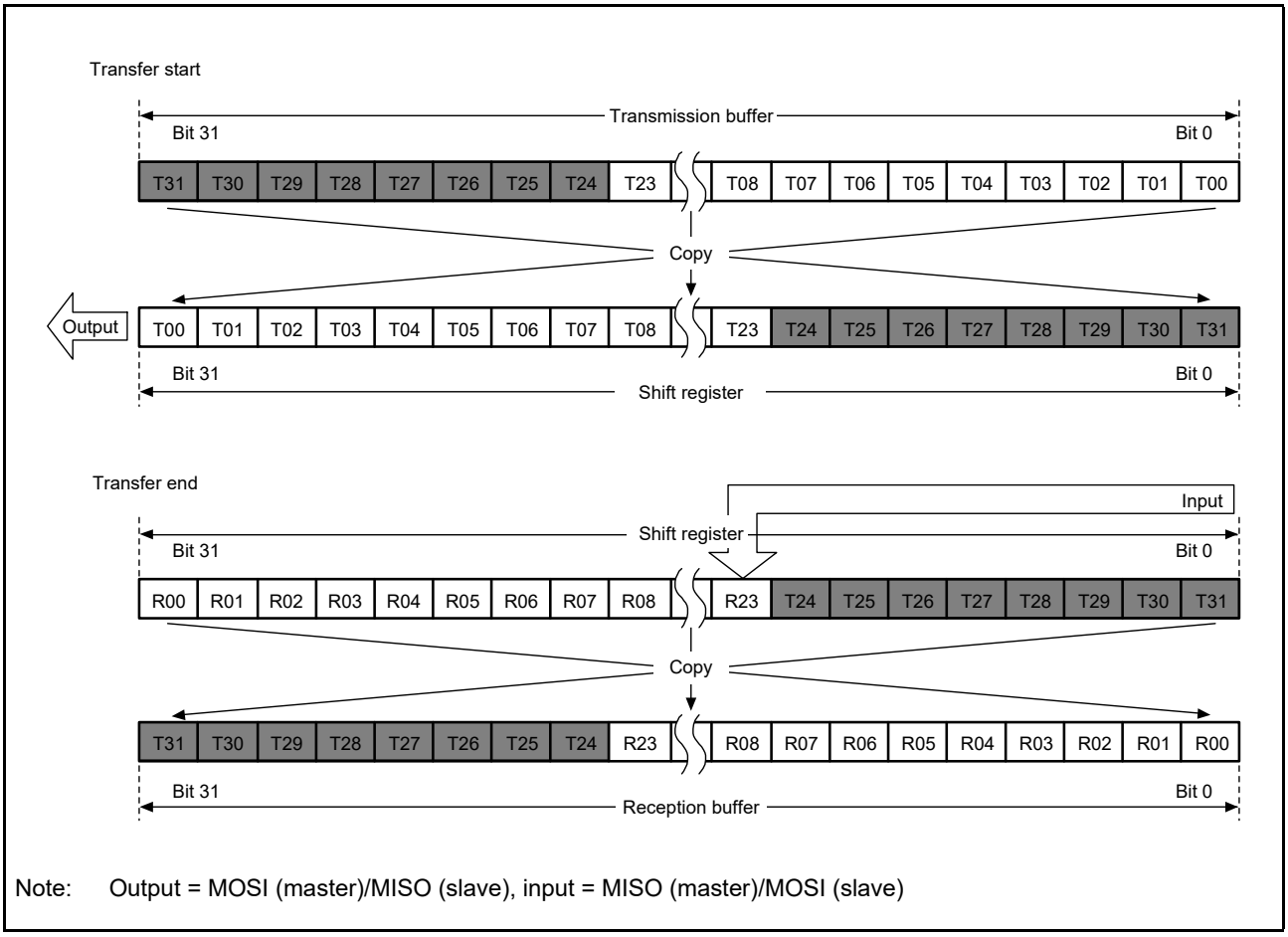
(4) LSB First Transfer (24-Bit Data)

Figure 28.17 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity disabled, 24 bits as the RSPI data length for an example that is not 32 bits, and LSB first selected.

In transmission, the lower-order 24 bits (T23 to T00) from the current stage of the transmission buffer are reordered bit by bit to obtain the order T00 to T23 for copying to the shift register. Data for transmission are shifted out from the shift register in order from T00, through T01, and so on to T23.

In reception, received data are shifted in bit by bit through bit 8 of the shift register. When bits R00 to R23 have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer.

At this time, the higher-order 8 bits of the transmission buffer are stored in the higher-order 8 bits of the reception buffer. Writing 0 to bits T31 to T24 at the time of transmission leads to 0 being inserted in the higher-order 8 bits of the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

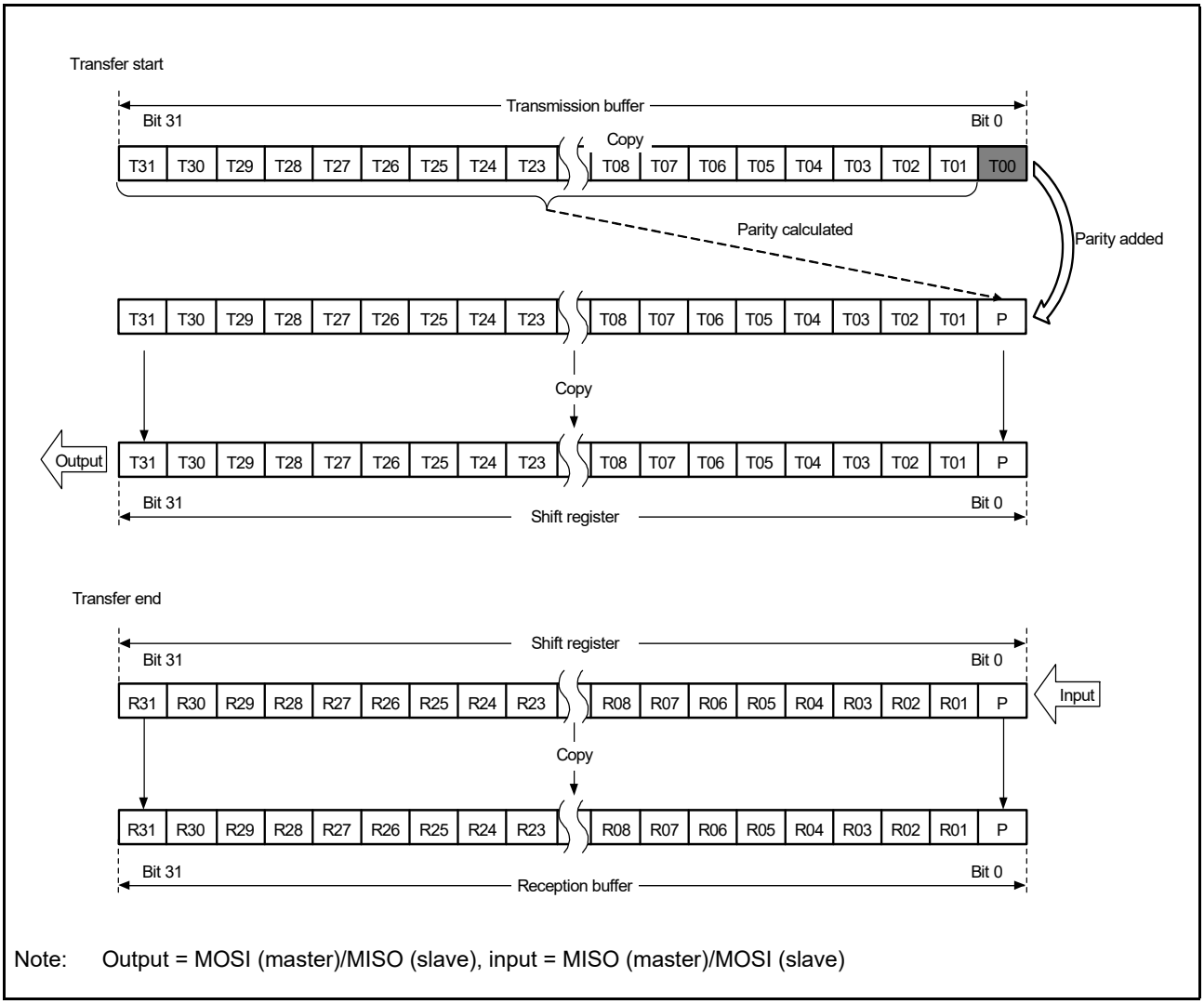
Figure 28.17 LSB First Transfer (24-Bit Data, Parity Disabled)

28.3.4.2 When Parity is Enabled (SPCR2.SPPE = 1)

When parity is enabled, the lowest-order bit of the data for transmission becomes a parity bit. Hardware calculates the value of the parity bit.

(1) MSB First Transfer (32-Bit Data)

Figure 28.18 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity enabled, an RSPI data length of 32 bits, and MSB first selected.
In transmission, the value of the parity bit (P) is calculated from bits T31 to T01. This replaces the final bit, T00, and the whole is copied to the shift register. Data are transmitted in the order T31, T30, …, T01, and P.
In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R31 to P have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. On copying of data to the shift register, the data from R31 to P are checked by judging the parity.

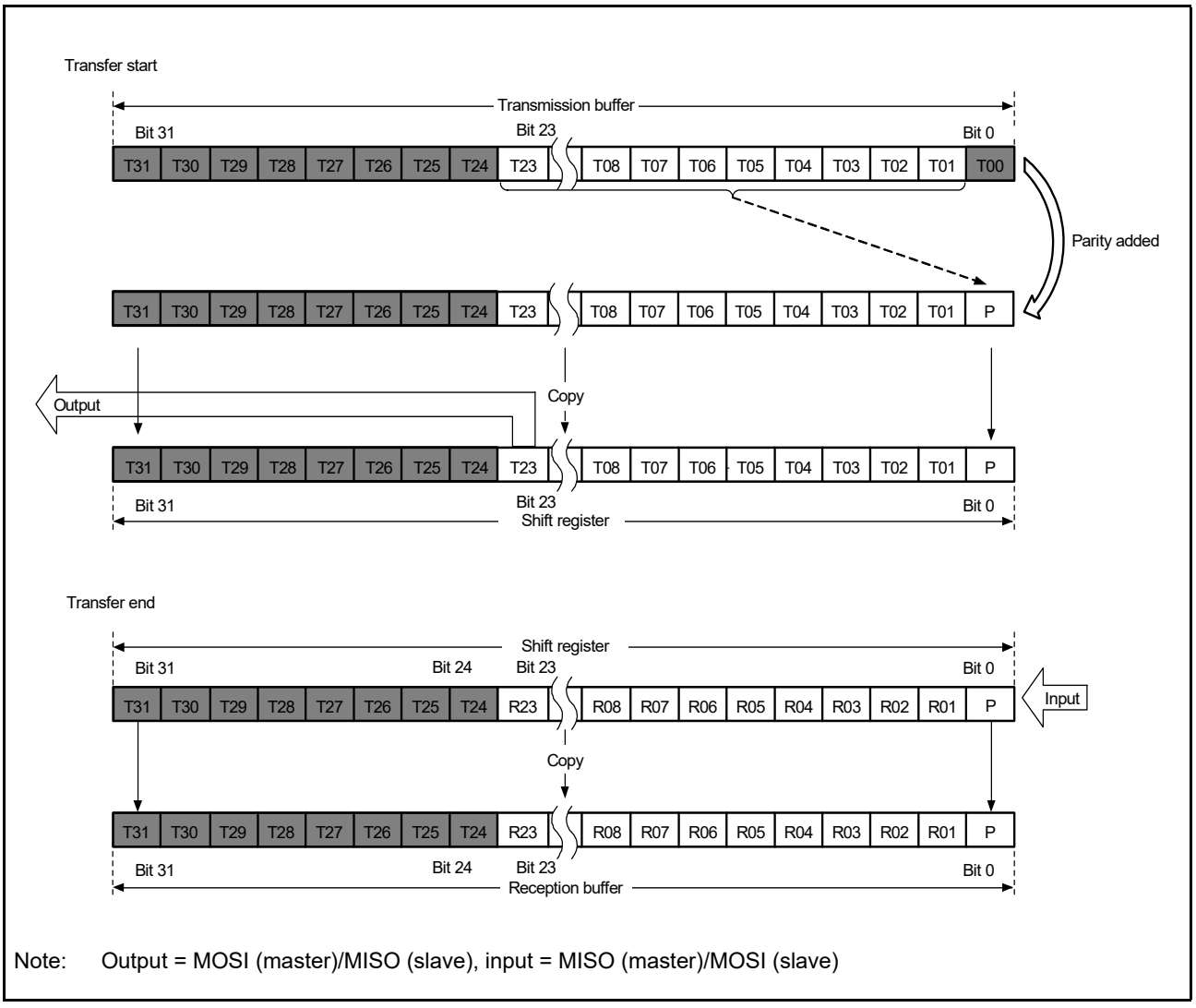


Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 28.18 MSB First Transfer (32-Bit Data, Parity Enabled)

(2) MSB First Transfer (24-Bit Data)

Figure 28.19 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity enabled, 24 bits as the RSPI data length for an example that is not 32 bits, and MSB first selected.
In transmission, the value of the parity bit (P) is calculated from bits T23 to T01. This replaces the final bit, T00, and the whole is copied to the shift register. Data are transmitted in the order T23, T22, …, T01, and P.
In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R23 to P have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. On copying of data to the shift register, the data from R23 to P are checked by judging the parity. At this time, the higher-order 8 bits of the transmission buffer are stored in the higher-order 8 bits of the reception buffer. Writing 0 to bits T31 to T24 at the time of transmission leads to 0 being inserted in the higher-order 8 bits of the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

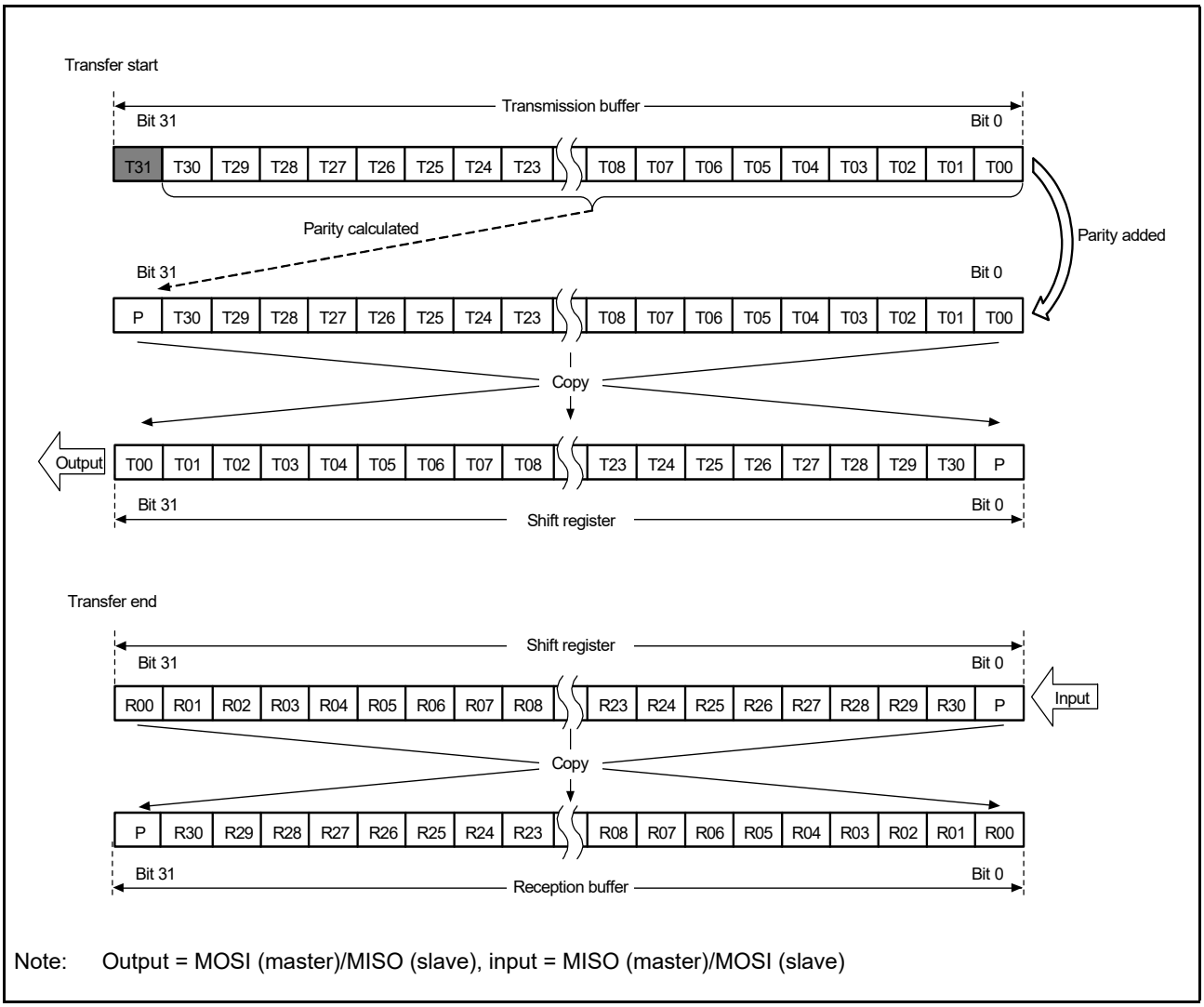
Figure 28.19 MSB First Transfer (24-Bit Data, Parity Enabled)

(3) LSB First Transfer (32-Bit Data)

Figure 28.20 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity enabled, an RSPI data length of 32 bits, and LSB first selected.

In transmission, the value of the parity bit (P) is calculated from bits T30 to T00. This replaces the final bit, T31, and the whole is copied to the shift register. Data are transmitted in the order T00, T01, …, T30, and P.

In reception, received data are shifted in bit by bit through bit 0 of the shift register. When bits R00 to P have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. On copying of data to the shift register, the data from R00 to P are checked by judging the parity.

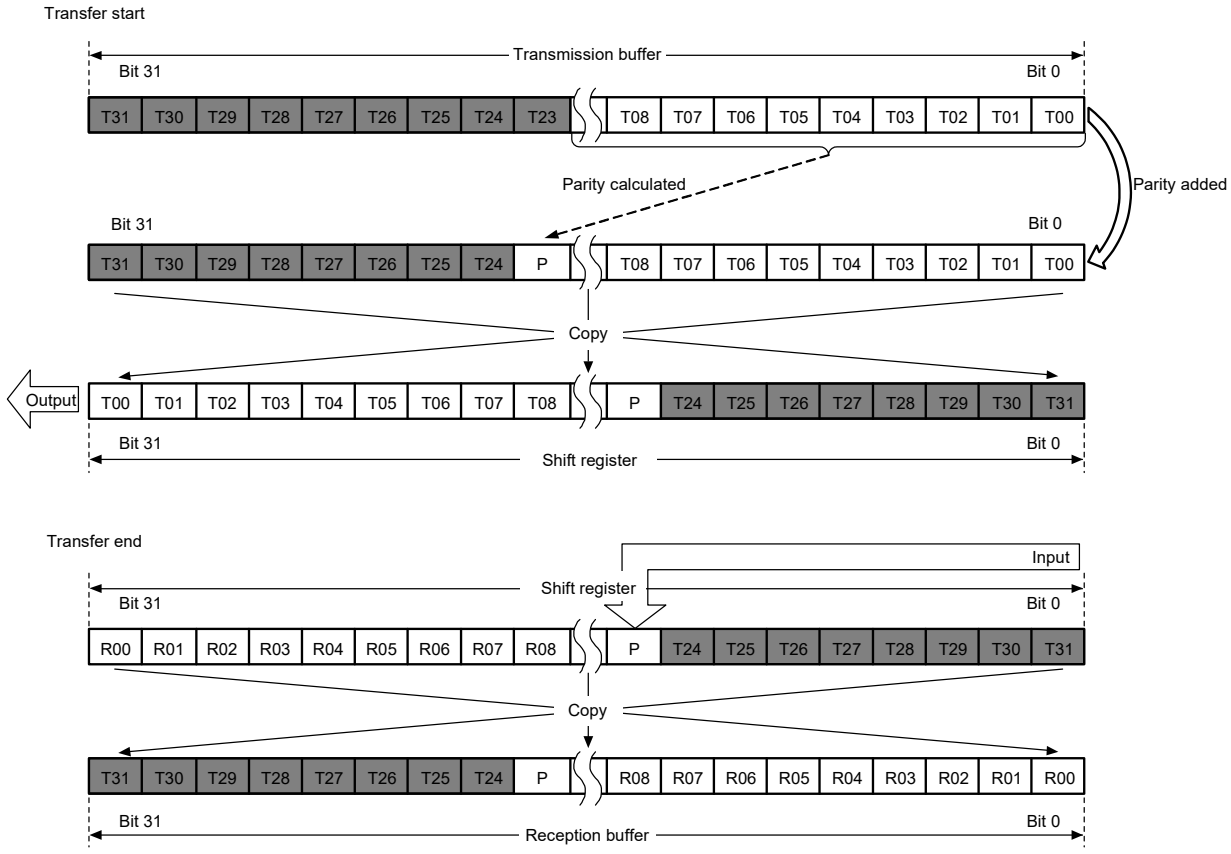


Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 28.20 LSB First Transfer (32-Bit Data, Parity Enabled)

(4) LSB First Transfer (24-Bit Data)

Figure 28.21 shows details of operations by the RSPI data register (SPDR) and the shift register in transfer with parity enabled, 24 bits as the RSPI data length for an example that is not 32 bits, and LSB first selected.
In transmission, the value of the parity bit (P) is calculated from bits T22 to T00. This replaces the final bit, T23, and the whole is copied to the shift register. Data are transmitted in the order T00, T01, …, T22, and P.
In reception, received data are shifted in bit by bit through bit 8 of the shift register. When bits R00 to P have been collected after input of the required number of cycles of RSPCK, the value in the shift register is copied to the reception buffer. On copying of data to the shift register, the data from R00 to P are checked by judging the parity. At this time, the higher-order 8 bits of the transmission buffer are stored in the higher-order 8 bits of the reception buffer. Writing 0 to bits T31 to T24 at the time of transmission leads to 0 being inserted in the higher-order 8 bits of the reception buffer.



Note: Output = MOSI (master)/MISO (slave), input = MISO (master)/MOSI (slave)

Figure 28.21 LSB First Transfer (24-Bit Data, Parity Enabled)

28.3.5 Transfer Format

28.3.5.1 CPHA = 0

Figure 28.22 shows a sample transfer format for the serial transfer of 8-bit data when the SPCMDm.CPHA bit is 0. Note that clock synchronous operation (the SPCR.SPMS bit is 1) should not be set when the RSPI operates in slave mode (SPCR.MSTR = 0) and the CPHA bit is 0. In Figure 28.22, RSPCKy (CPOL = 0) indicates the RSPCKy signal waveform when the SPCMDm.CPOL bit is 0; RSPCKy (CPOL = 1) indicates the RSPCKy signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which the RSPI fetches serial transfer data into the shift register. The I/O directions of the signals depend on the RSPI settings. For details, refer to section 28.3.2, Controlling RSPI Pins.

When the SPCMDm.CPHA bit is 0, the driving of valid data to the MOSIy and MISOy signals commences at an SSLyi signal assertion timing. The first RSPCKy signal change timing that occurs after the SSLyi signal assertion becomes the first transfer data fetch timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for MOSIy and MISOy signals is always 1/2 RSPCK cycles after the transfer data fetch timing. The CPOL bit setting does not affect the RSPCK signal operation timing; it only affects the signal polarity.

t1 denotes a period from an SSLyi signal assertion to RSPCKy oscillation (RSPCK delay). t2 denotes a period from the termination of RSPCKy oscillation to an SSLyi signal negation (SSL negation delay). t3 denotes a period in which SSLyi signal assertion is suppressed for the next transfer after the end of serial transfer (next-access delay). t1, t2, and t3 are controlled by a master device running on the RSPI system. For a description of t1, t2, and t3 when the RSPI of this LSI is in master mode, refer to section 28.3.10.1, Master Mode Operation.

(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

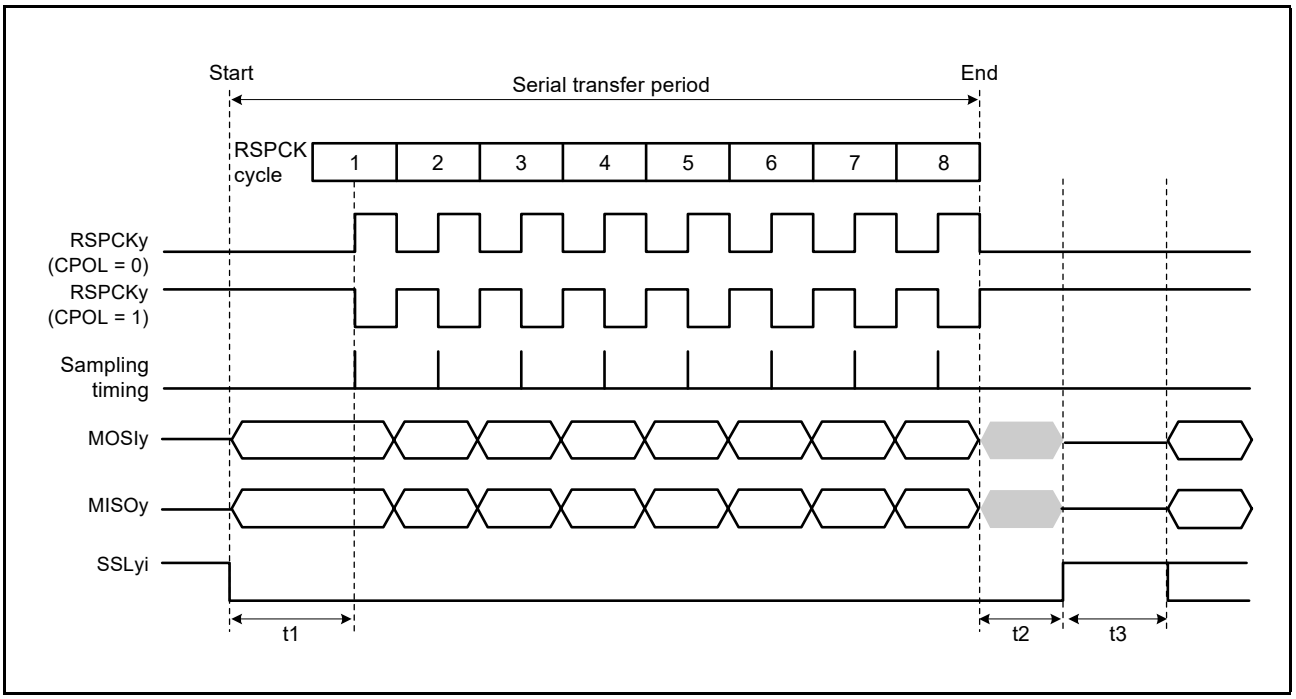


Figure 28.22 RSPI Transfer Format (CPHA = 0)

28.3.5.2 CPHA = 1

Figure 28.23 shows a sample transfer format for the serial transfer of 8-bit data when the SPCMDm.CPHA bit is 1. However, when the SPCR.SPMS bit is 1, the SSLyi signals are not used, and only the three signals RSPCKy, MOSIy, and MISOy handle communications. In Figure 28.23, RSPCK (CPOL = 0) indicates the RSPCKy signal waveform when the SPCMDm.CPOL bit is 0; RSPCK (CPOL = 1) indicates the RSPCKy signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which the RSPI fetches serial transfer data into the shift register. The I/O directions of the signals depend on the RSPI mode (master or slave). For details, refer to section 28.3.2, Controlling RSPI Pins.

When the SPCMDm.CPHA bit is 1, the driving of invalid data to the MISOy signal commences at an SSLyi signal assertion timing. The output of valid data to the MOSIy and MISOy signals commences at the first RSPCKy signal change timing that occurs after the SSLyi signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycles after the data update timing. The SPCMDm.CPOL bit setting does not affect the RSPCKy signal operation timing; it only affects the signal polarity.

t1, t2, and t3 are the same as those in the case of CPHA = 0. For a description of t1, t2, and t3 when the RSPI of this LSI is in master mode, refer to section 28.3.10.1, Master Mode Operation.

(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

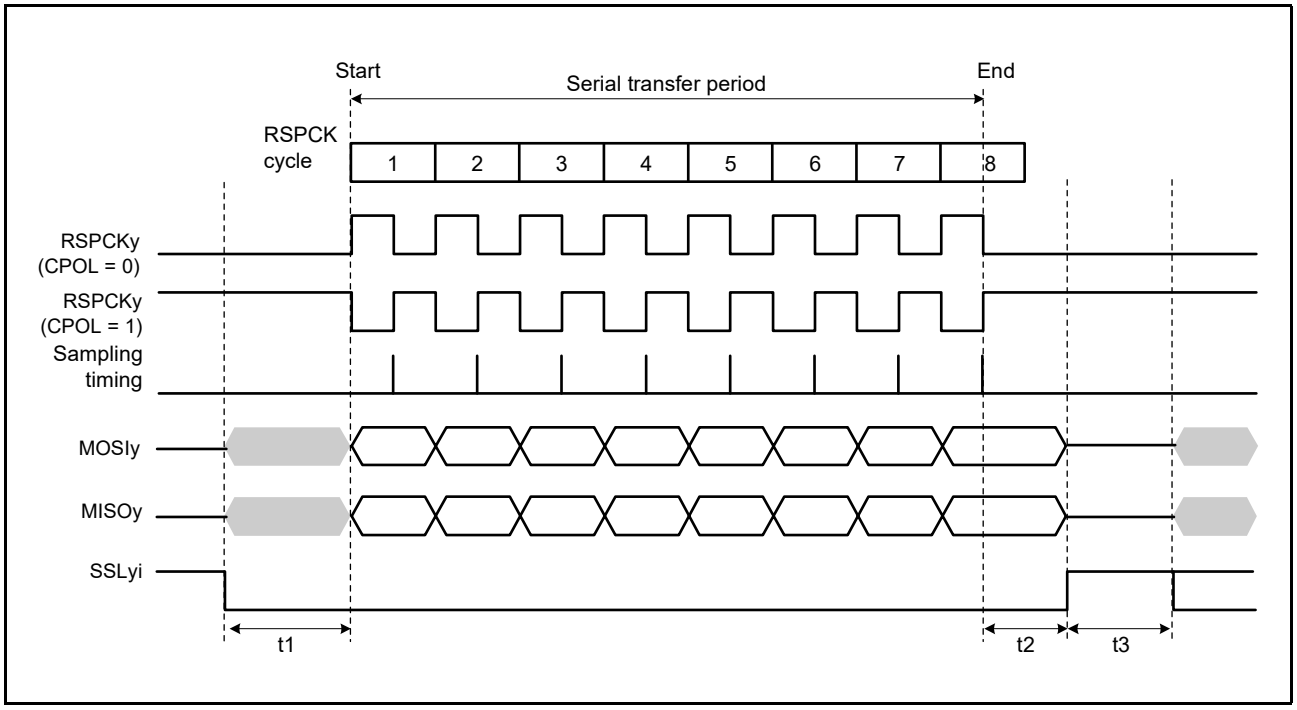


Figure 28.23 RSPI Transfer Format (CPHA = 1)

28.3.6 Communications Operating Mode

Full-duplex synchronous serial communications or transmit operations only can be selected by the communications operating mode select bit (SPCR.TXMD). The SPDR access shown in Figure 28.24 and Figure 28.25 indicates the condition of access to the SPDR register, where W denotes a write cycle.

28.3.6.1 Full-Duplex Synchronous Serial Communications (SPCR.TXMD = 0)

Figure 28.24 shows an example of operation when the communications operating mode select bit (SPCR.TXMD) is set to 0. In the example in Figure 28.24, the RSPI performs an 8-bit serial transfer in which the SPDCR.SPFC[1:0] bits are 00b, the SPCMDm.CPHA bit is 1, and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

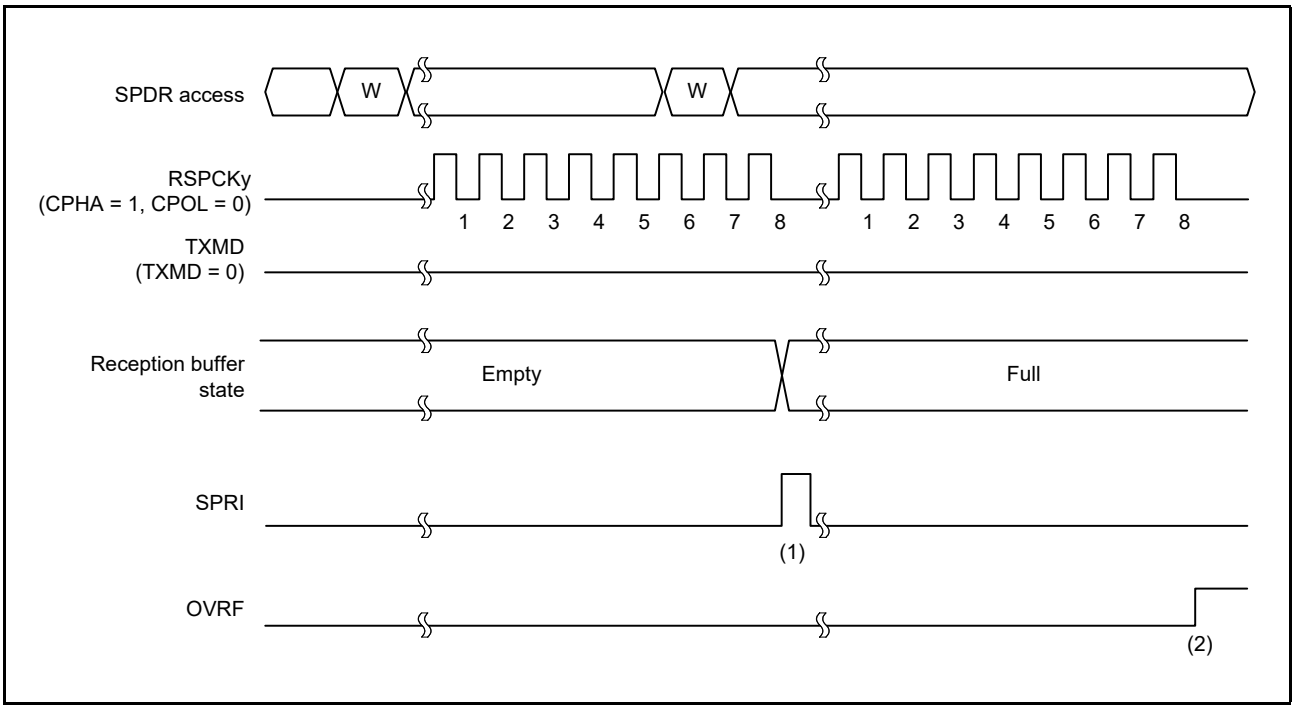


Figure 28.24 Operation Example of SPCR.TXMD = 0

The operation of the flags at timings shown in steps (1) and (2) in the figure is described below.

(1) When a serial transfer ends with the reception buffer of SPDR empty, the RSPI generates a reception buffer full interrupt request (SPRI) and copies the received data in the shift register to the reception buffer.

(2) When a serial transfer ends with the reception buffer of SPDR holding data that was received in the previous serial transfer, the RSPI sets the SPSR.OVRF flag to 1 and discards the received data in the shift register.

28.3.6.2 Transmit Operations Only (SPCR.TXMD = 1)

Figure 28.25 shows an example of operation when the communications operating mode select bit (SPCR.TXMD) is set to 1. In the example in Figure 28.25, the RSPI performs an 8-bit serial transfer in which the SPDCR.SPFC[1:0] bits are 00b, the SPCMDm.CPHA bit is 1, and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

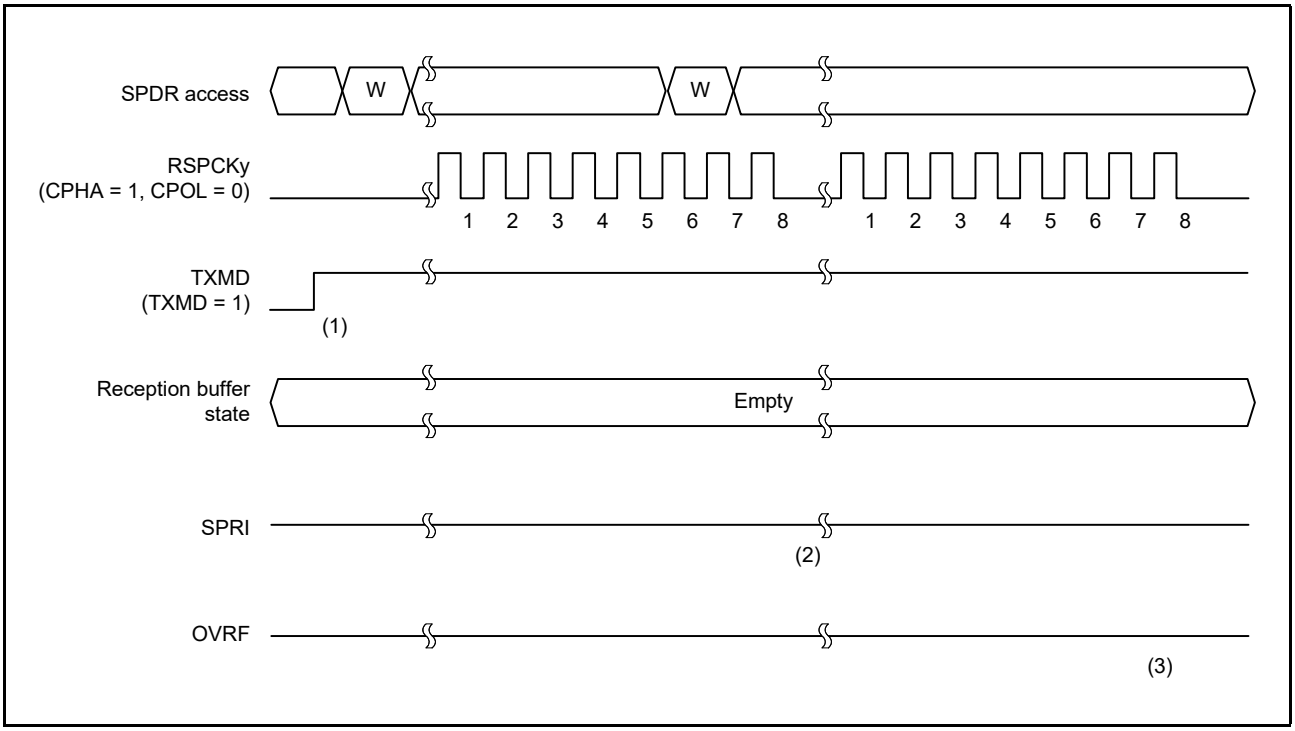


Figure 28.25 Operation Example of SPCR.TXMD = 1

The operation of the flags at timings shown in steps (1) to (3) in the figure is described below.

(1) Make sure there is no data left in the reception buffer and the SPSR.OVRF flag is 0 before entering the mode of transmit operations only (SPCR.TXMD = 1).

(2) When a serial transfer ends with the reception buffer of SPDR empty, if the mode of transmit operations only is selected (SPCR.TXMD = 1), the RSPI does not copy the data in the shift register to the reception buffer.

(3) Since the reception buffer of SPDR does not hold data that was received in the previous serial transfer, even when a serial transfer ends, the SPSR.OVRF flag retains the value of 0, and the data in the shift register is not copied to the reception buffer.

When performing transmit operations only (SPCR.TXMD = 1), the RSPI transmits transmit data but does not receive received data. Therefore, the SPSR.OVRF flag remains 0 at the timings of (1) to (3).

28.3.7 Transmission Buffer Empty/Reception Buffer Full Interrupts

Figure 28.26 shows an example of operation of the transmission buffer empty interrupt (SPTI) and the reception buffer full interrupt (SPRI). The SPDR register access shown in Figure 28.26 indicates the condition of access to the SPDR register, where W denotes a write cycle, and R a read cycle. In the example in Figure 28.26, the RSPI performs an 8-bit serial transfer in which the SPCR.TXMD bit is 0, the SPDCR.SPFC[1:0] bits are 00b, the SPCMDm.CPHA bit is 1, and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

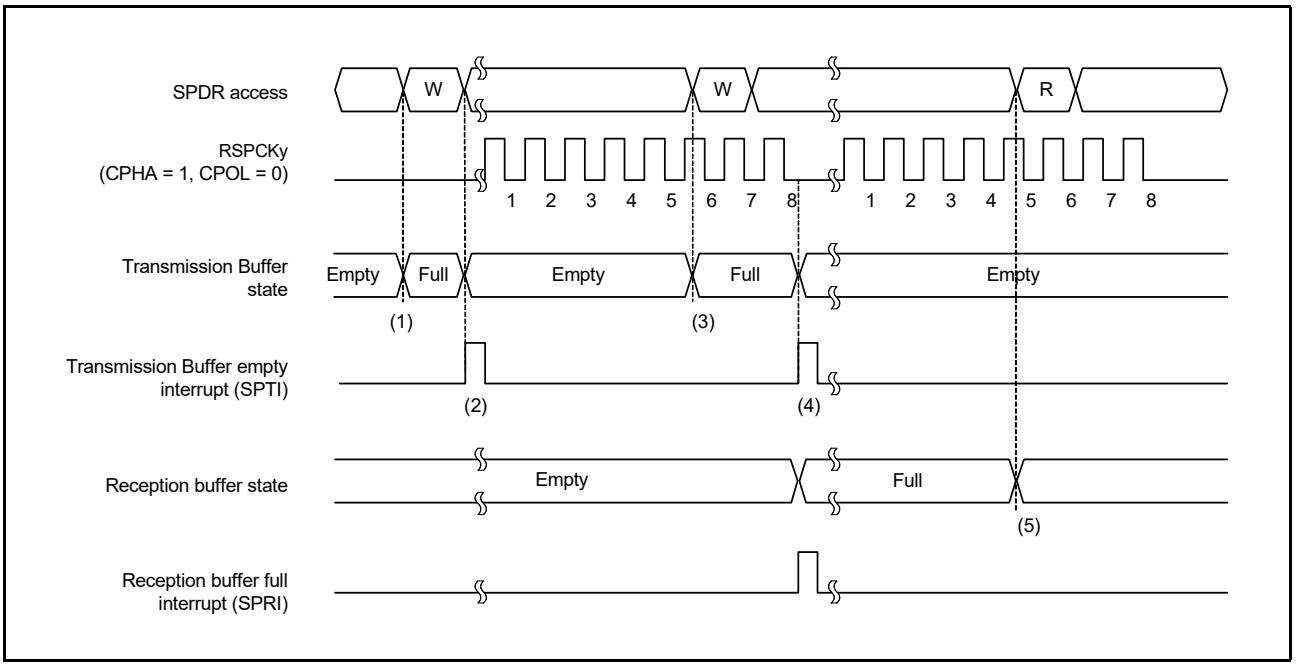


Figure 28.26 Operation Example of Transmission Buffer Empty Interrupt (SPTI) and Reception Buffer Full Interrupt (SPRI)

The operation of the interrupts at timings shown in steps (1) to (5) in the figure is described below.

1. When transmit data is written to SPDR when the transmission buffer of SPDR is empty (data for the next transfer is not set), the RSPI writes data to the transmission buffer.
2. If the shift register is empty, the RSPI copies the data in the transmission buffer to the shift register and generates a transmission buffer empty interrupt request (SPTI). How a serial transfer is started depends on the mode of the RSPI. For details, refer to section 28.3.10, SPI Operation, and section 28.3.11, Clock Synchronous Operation.
3. When transmit data is written to SPDR by the transmission buffer empty interrupt routine, the data is transferred to the transmission buffer. Because the data being transferred serially is stored in the shift register, the RSPI does not copy the data in the transmission buffer to the shift register.
4. When the serial transfer ends with the reception buffer of SPDR being empty, the RSPI copies the receive data in the shift register to the reception buffer and generates a reception buffer full interrupt request (SPRI). Since the shift register becomes empty upon completion of serial transfer, when the transmission buffer had been full before the serial transfer ended, the RSPI copies the data in the transmission buffer to the shift register. Even when received data is not copied from the shift register to the reception buffer in an overrun error status, upon completion of the serial transfer, the RSPI determines that the shift register is empty, thus data transfer from the transmission buffer to the shift register is enabled.
5. When SPDR is read by the reception buffer full interrupt routine, the receive data can be read.

If SPDR is written to when the transmission buffer holds data that has not yet been transmitted, the RSPI does not update the data in the transmission buffer. When writing to SPDR, make sure to use a transmission buffer empty interrupt request. To use a transmission buffer empty interrupt, set the SPTIE bit in SPCR to 1.
If the RSPI function is disabled (the SPCR.SPE bit being 0), set the SPTIE bit to 0.
When serial transfer ends with the reception buffer being full, the RSPI does not copy data from the shift register to the reception buffer, and detects an overrun error (refer to section 28.3.8, Error Detection). To prevent a receive data overrun error, read the received data using a reception buffer full interrupt request before the next serial transfer ends. To use an RSPI reception buffer full interrupt, set the SPCR.SPRIE bit to 1.
For the states of the transmit and reception buffers, transmission buffer empty and reception buffer interrupts or the corresponding IRQ status register (IRQSn) can be used to confirm the generation of interrupt requests. For the IRQ status register (IRQSn), see section 12.4.2.1, IRQ Status Register n (IRQSn) (n = 0 to 9).

28.3.8 Error Detection

In the normal RSPI serial transfer, the data written to the transmission buffer of SPDR is transmitted, and the received data can be read from the reception buffer of SPDR. If access is made to SPDR, depending on the status of the transmission/reception buffer or the status of the RSPI at the beginning or end of serial transfer, in some cases non-normal transfers can be executed.
If a non-normal transfer operation occurs, the RSPI detects the event as an overrun error, parity error, or mode fault error. Table 28.8 lists the relationship between non-normal transfer operations and the RSPI's error detection function.

Table 28.8 Relationship between Non-Normal Transfer Operations and RSPI Error Detection Function

	Occurrence Condition	RSPI Operation	Error Detection
1	SPDR is written when the transmission buffer is full.	• The contents of the transmission buffer are kept. • Missing write data.	None
2	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is transmitted.	None
3	SPDR is read when the reception buffer is empty.	Previously received data is output.	None
4	Serial transfer terminates when the reception buffer is full.	• The contents of the reception buffer are kept. • Missing receive data.	Overrun error
5	An incorrect parity bit is received when performing full-duplex synchronous serial communications with the parity function enabled.	The parity error flag is asserted.	Parity error
6	The SSLy0 input signal is asserted when the serial transfer is idle in multi-master mode.	• Driving of the RSPCKy, MOSIy, SSLy1 to SSLy3 output signals is stopped. • RSPI function is disabled.	Mode fault error
7	The SSLy0 input signal is asserted during serial transfer in multi-master mode.	• Serial transfer is suspended. • Missing transmit/receive data. • Driving of the RSPCKy, MOSIy, SSLy1 to SSLy3 output signals is stopped. • RSPI function is disabled.	Mode fault error
8	The SSLy0 input signal is negated during serial transfer in slave mode.	• Serial transfer is suspended. • Missing transmit/receive data. • Driving of the MISOy output signal is stopped. • RSPI function is disabled.	Mode fault error

y = 0, 1 (for all channels)

On operation 1 described in Table 28.8, the RSPI does not detect an error. To prevent data omission during the writing to SPDR, write operations to SPDR should be executed using a transmission buffer empty interrupt request.
Likewise, the RSPI does not detect an error on operation 2. In a serial transfer that was started before the shift register was updated, the RSPI sends the data that was received in the previous serial transfer, and does not treat the operation indicated in 2 as an error. Note that the received data from the previous serial transfer is retained in the reception buffer of SPDR, thus it can be correctly read (if SPDR is not read before the end of the serial transfer, an overrun error may occur).
Similarly, the RSPI does not detect an error on operation 3. To prevent extraneous data from being read, SPDR read operation should be executed using an RSPI reception buffer full interrupt request.
An overrun error shown in 4 is described in section 28.3.8.1, Overrun Error. A parity error shown in 5 is described in section 28.3.8.2, Parity Error. A mode fault error shown in 6 to 8 is described in section 28.3.8.3, Mode Fault Error. For the transmit and receive interrupts, refer to section 28.3.7, Transmission Buffer Empty/Reception Buffer Full Interrupts.

28.3.8.1 Overrun Error

If a serial transfer ends when the reception buffer of SPDR is full, the RSPI detects an overrun error, and sets the OVRF flag in SPSR to 1. When the OVRF flag is 1, the RSPI does not copy data from the shift register to the reception buffer so that the data prior to the occurrence of the error is retained in the reception buffer. To set the OVRF flag to 0, write 0 to the OVRF flag after the CPU has read SPSR with the OVRF flag set to 1.
Figure 28.27 shows an example of operation of the OVRF flag. The SPSR and SPDR accesses shown in Figure 28.27 indicate the condition of accesses to SPSR and SPDR, respectively, where W denotes a write cycle, and R a read cycle. In the example in Figure 28.27, the RSPI performs an 8-bit serial transfer in which the SPCMDm.CPHA bit is 1 and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

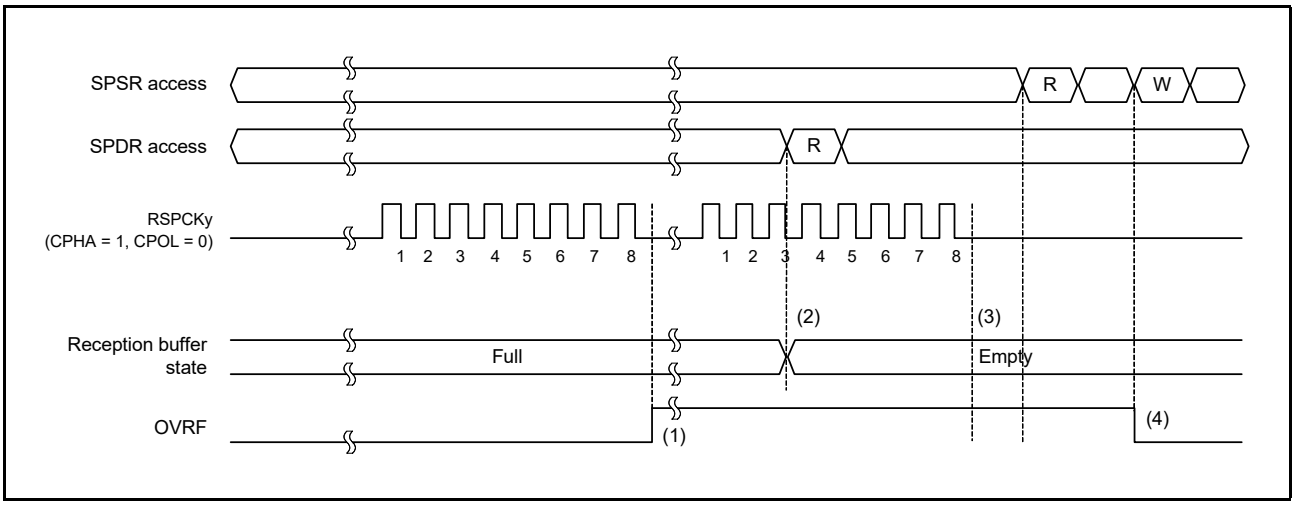


Figure 28.27 Operation Example of OVRF Flag

The operation of the flags at the timing shown in steps (1) to (4) in the figure is described below.

1. If a serial transfer terminates with the reception buffer full, the RSPI detects an overrun error, and sets the OVRF flag to 1. The RSPI does not copy the data in the shift register to the reception buffer. Even if the SPPE bit is 1, parity errors are not detected. In master mode, the RSPI copies the pointer value to SPCMDm register to the SPSSR.SPECM[2:0] bits.
2. When SPDR is read, the RSPI can read the data in the reception buffer. The reception buffer becoming empty does not set the OVRF flag to 0.
3. If the serial transfer ends with the OVRF flag being 1 (an overrun error occurs), the RSPI does not copy the data in the shift register to the reception buffer. A reception-buffer interrupt is not generated. Even if the SPPE bit is 1, parity errors are not detected. When in master mode, the RSPI does not update the SPSSR.SPECM[2:0] bits. When in an overrun error state and the RSPI does not copy the received data from the shift register to the reception buffer, upon termination of the serial transfer, the RSPI determines that the shift register is empty; in this manner, data transfer from the transmission buffer to the shift register is enabled.
4. If the value 0 is written to the OVRF flag after SPSR is read when the OVRF flag is 1, the OVRF flag is set to 0.

The occurrence of an overrun can be checked either by reading SPSR or by using an RSPI error interrupt and reading SPSR. When executing a serial transfer, measures should be taken to ensure the early detection of overrun errors, such as reading SPSR immediately after SPDR is read. When the RSPI is used in master mode, the pointer value to SPCMDm register at the occurrence of the error can be checked by reading the SPSSR.SPECM[2:0] bits.
If an overrun error occurs and the OVRF flag is set to 1, normal reception operations cannot be performed until the OVRF flag is set to 0.
When the RSPCK auto-stop function is enabled in master mode, an overrun error does not occur. Figure 28.28 and Figure 28.29 show the clock stop waveform when a serial transfer continues while the reception buffer is full in master mode.

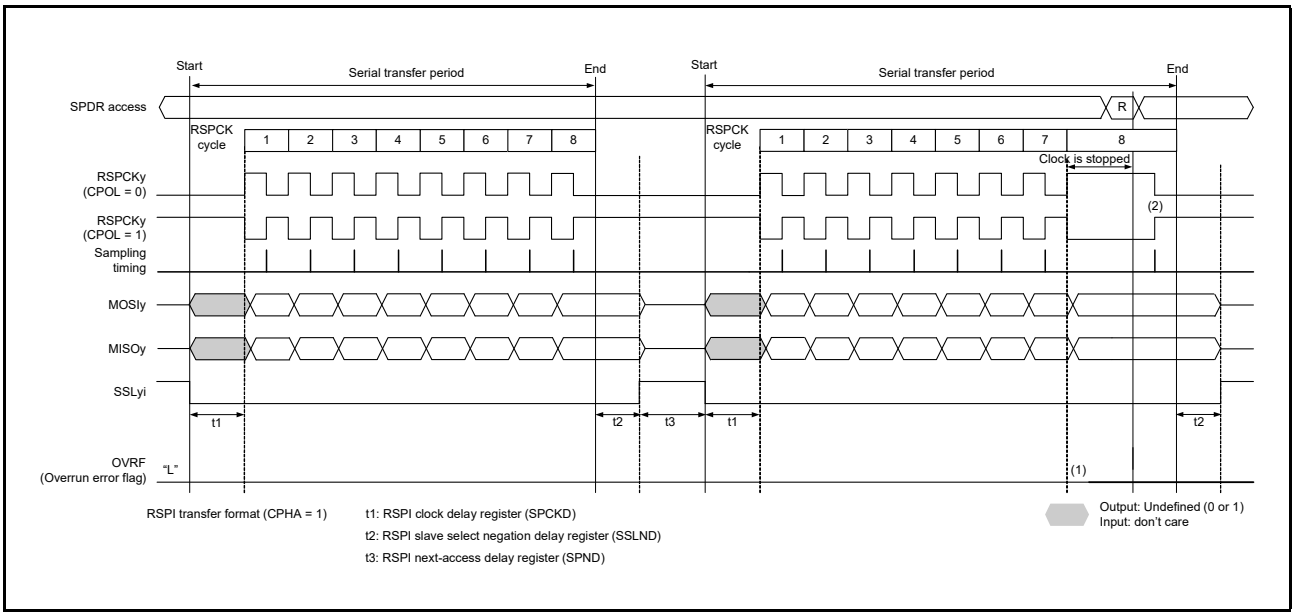


Figure 28.28 Clock Stop Waveform When a Serial Transfer Continues While the Reception Buffer is Full in Master Mode (CPHA = 1)

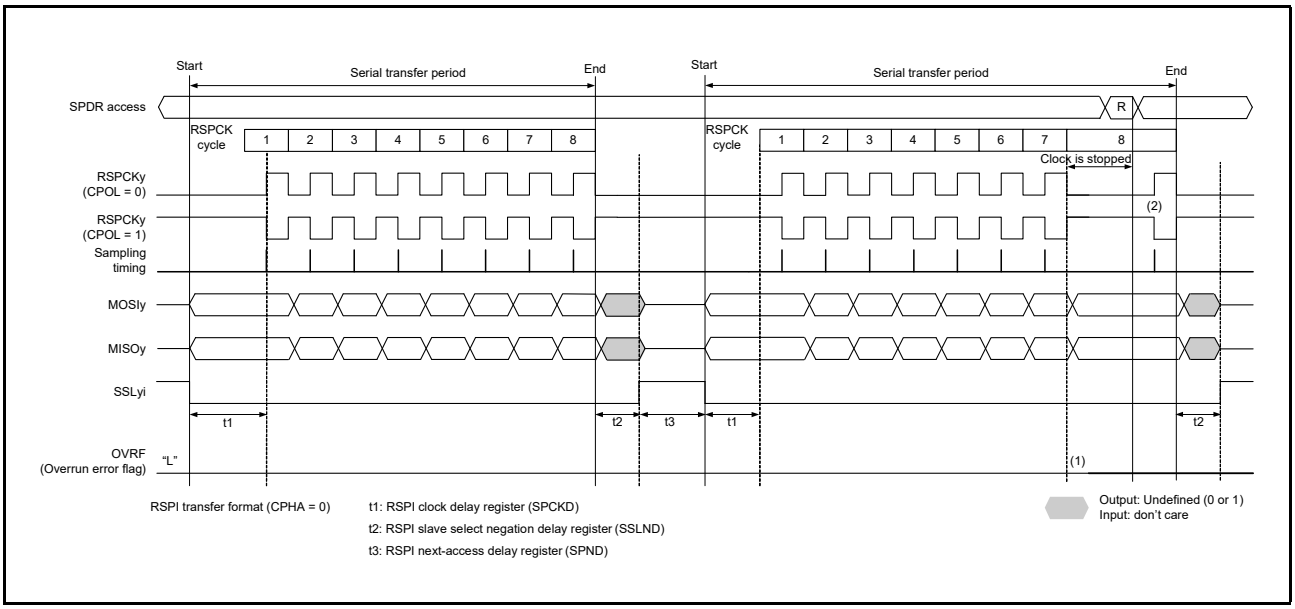


Figure 28.29 Clock Stop Waveform When a Serial Transfer Continues While the Reception Buffer is Full in Master Mode (CPHA = 0)

The operation of the flags at the timings shown in steps (1) and (2) in the figure is described below.

(1) When the reception buffer is full, an overrun error to stop the reception buffer does not occur.

(2) If SPDR is read while the clock is stopped, data in the reception buffer can be read. The RSPCK clock restarts after reading the reception buffer.

28.3.8.2 Parity Error

If full-duplex synchronous serial communications is performed with the SPCR.TXMD bit set to 0 and the SPCR2.SPPE bit set to 1, when serial transfer ends, the RSPI checks whether there are parity errors. Upon detecting a parity error in the received data, the RSPI sets the SPSR.PERF flag to 1. Since the RSPI does not copy the data in the shift register to the reception buffer when the SPSR.OVRF flag is set to 1, parity error detection is not performed for the received data. To set the PERF flag to 0, write 0 to the PERF flag after SPSR register is read with the PERF flag set to 1.
Figure 28.30 shows an example of operation of the OVRF and PERF flags. The SPSR access shown in Figure 28.30 indicates the condition of access to SPSR register, where W denotes a write cycle, and R a read cycle. In the example of Figure 28.30, full-duplex synchronous serial communications is performed while the SPCR.TXMD bit is 0 and the SPCR2.SPPE bit is 1. The RSPI performs an 8-bit serial transfer in which the SPCMDm.CPHA bit is 1 and the SPCMDm.CPOL bit is 0. The numbers given under the RSPCKy waveform represent the number of RSPCK cycles (i.e., the number of transferred bits) (m = 0 to 7; y = 0, 1 (for all channels)).

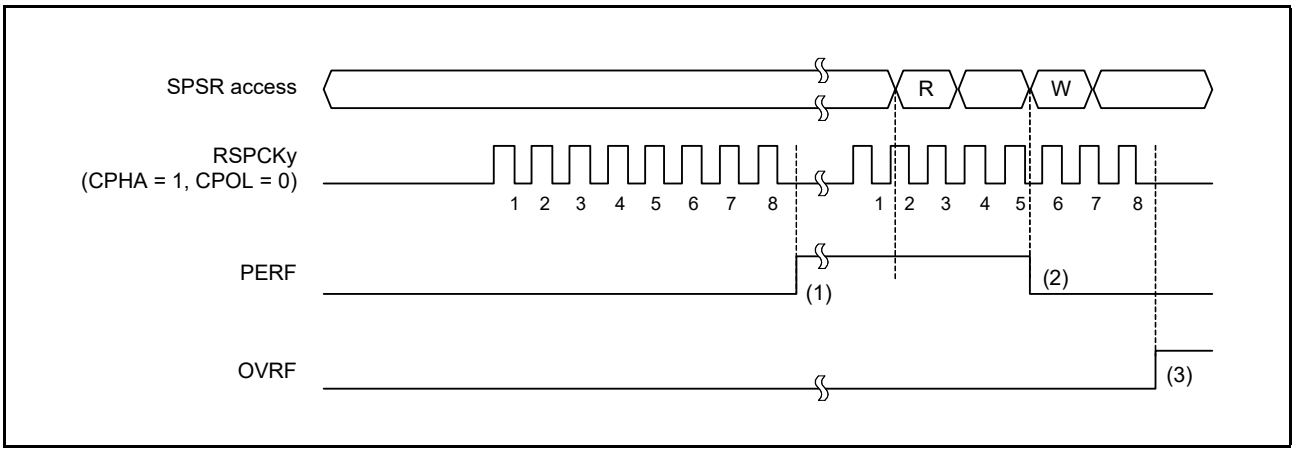


Figure 28.30 Operation Example of PERF Flag

The operation of the flags at the timing shown in steps (1) to (3) in the figure is described below.

1. If a serial transfer terminates with the RSPI not detecting an overrun error, the RSPI copies the data in the shift register to the reception buffer. The RSPI judges the received data at this timing, and sets the PERF flag to 1 if a parity error is detected. In master mode, the RSPI copies the pointer value to SPCMDm register to the SPSSR.SPECM[2:0] bits.
2. If the value 0 is written to the PERF flag after SPSR register is read when the PERF flag is 1, the PERF flag is set to 0.
3. When the RSPI detects an overrun error and serial transfer is terminated, the data in the shift register is not copied to the reception buffer. The RSPI does not perform parity error detection at this timing.

The occurrence of a parity error can be checked either by reading SPSR register or by using an RSPI error interrupt and reading SPSR register. When executing a serial transfer, measures should be taken to ensure the early detection of parity errors, such as reading SPSR. When the RSPI is used in master mode, the pointer value to SPCMDm register at the occurrence of the error can be checked by reading the SPSSR.SPECM[2:0] bits.

28.3.8.3 Mode Fault Error

The RSPI operates in multi-master mode when the SPCR.MSTR bit is 1, the SPCR.SPMS bit is 0, and the SPCR.MODFEN bit is 1. If the active level is input with respect to the SSLy0 input signal of the RSPI in multi-master mode, the RSPI detects a mode fault error irrespective of the status of the serial transfer, and sets the MODF bit in the RSPI status register (SPSR) to 1. Upon detecting the mode fault error, the RSPI copies the value of the pointer to SPCMDm to the SPSSR.SPECM[2:0] bits. The active level of the SSLy0 signal is determined by the SSL0P bit in the RSPI slave select polarity register (SSLP).

When the MSTR bit is 0, the RSPI operates in slave mode. The RSPI detects a mode fault error if the MODFEN bit in the RSPI in slave mode is 1, and the SPMS bit is 0, and if the SSLy0 input signal is negated during the serial transfer period (from the time the driving of valid data is started to the time the final valid data is fetched).

Upon detecting a mode fault error, the RSPI stops driving of the output signals and clears the SPCR.SPE bit to 0 (refer to section 28.3.9, Initializing RSPI). In the case of multi-master configuration, detection of a mode fault error is used to stop driving of the output signals and the RSPI function, which allows the master right to be released.

The occurrence of a mode fault error can be checked either by reading SPSR or by using an RSPI error interrupt and reading SPSR. Detecting mode-fault errors without utilizing the RSPI error interrupt requires polling of SPSR. When using the RSPI in master mode, the pointer value to SPCMDm register at the occurrence of the error can be checked by reading the SPSSR.SPECM[2:0] bits.

When the MODF bit is 1, writing of the value 1 to the SPE bit is ignored by the RSPI. To enable the RSPI function after the detection of a mode fault error, the MODF bit must be set to 0 (m = 0 to 7; y = 0, 1 (for all channels)).

28.3.9 Initializing RSPI

If the value 0 is written to the SPCR.SPE bit or the RSPI sets the SPE bit to 0 because of the detection of a mode fault error, the RSPI disables the RSPI function, and initializes some of the module functions. When a system reset is generated, the RSPI initializes all of the module functions. The following describes initialization by the clearing of the SPCR.SPE bit and initialization by a system reset.

28.3.9.1 Initialization by Clearing the SPE Bit

When the SPCR.SPE bit is set to 0, the RSPI performs the following initialization:

- Suspending any serial transfer that is being executed
- Stopping the driving of output signals (Hi-Z) in slave mode
- Initializing the internal state of the RSPI
- Initializing the transmission buffer of the RSPI

Initialization by the clearing of the SPE bit does not initialize the control bits of the RSPI. For this reason, the RSPI can be started in the same transfer mode as prior to the initialization if the SPE bit is set to 1 again.
The SPSR.OVRF and SPSR.MODF flags are not initialized, nor is the value of the RSPI sequence status register (SPSSR) initialized. For this reason, even after the RSPI is initialized, data from the reception buffer can be read in order to check the status of error occurrence during an RSPI transfer.
The transmission buffer is initialized to an empty state. Therefore, if the SPCR.SPTIE bit is set to 1 after RSPI initialization, a transmission buffer empty interrupt is generated. When the RSPI is initialized, in order to disable any transmission buffer empty interrupt, the value 0 should be written to the SPTIE bit simultaneously with the writing of the value 0 to the SPE bit. To disable any transmission buffer empty interrupt after a mode fault error is detected, use an error handling routine to write the value 0 to the SPTIE bit.

28.3.9.2 System Reset

The initialization by a system reset completely initializes the RSPI through the initialization of all bits for controlling the RSPI, initialization of the status bits, and initialization of data registers, in addition to the requirements described in section 28.3.9.1, Initialization by Clearing the SPE Bit. For details, see section 6, Reset.

28.3.10 SPI Operation

28.3.10.1 Master Mode Operation

The only difference between single-master mode operation and multi-master mode operation lies in mode fault error detection (refer to section 28.3.8, Error Detection). When operating in single-master mode, the RSPI does not detect mode fault errors whereas the RSPI running in multi-master mode does detect mode fault errors. This section explains operations that are common to single-master mode and multi-master mode.

(1) Starting a Serial Transfer

The RSPI updates the data in the transmission buffer (SPTX) when data is written to the RSPI data register (SPDR) with the RSPI transmission buffer being empty (data for the next transfer is not set). When the shift register is empty after the number of frames set in the SPDCR.SPFC[1:0] bits are written to the SPDR, the RSPI copies data from the transmission buffer to the shift register and starts serial transfer. Upon copying transmit data to the shift register, the RSPI changes the status of the shift register to “full”, and upon termination of serial transfer, it changes the status of the shift register to “empty”. The status of the shift register cannot be referenced.
For details on the RSPI transfer format, refer to section 28.3.5, Transfer Format. The polarity of the SSLyi output pins depends on the SSLP register settings (i = 0 to 3; y = 0, 1 (for all channels)).

(2) Terminating a Serial Transfer

Irrespective of the SPCMDm.CPHA bit the RSPI terminates the serial transfer after transmitting an RSPCKy edge corresponding to the final sampling timing. If free space is available in the reception buffer (SPRX), upon termination of serial transfer, the RSPI copies data from the shift register to the reception buffer of the RSPI data register (SPDR).
It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the RSPI data length depends on the SPCMDm.SPB[3:0] bit setting. The polarity of the SSLyi output pin depends on the SSLP register settings. For details on the RSPI transfer format, refer to section 28.3.5, Transfer Format.
(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

(3) Sequence Control

The transfer format that is employed in master mode is determined by SPSCR, SPCMDm, SPBR, SPCKD, SSLND, and SPND registers.

SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by the RSPI in master mode. The following items are set in SPCMDm register: SSLyi pin output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether SPCKD is to be referenced, whether SSLND is to be referenced, and whether SPND is to be referenced. SPBR holds some of the bit rate settings; SPCKD, an RSPI clock delay value; SSLND, an SSL negation delay; and SPND, a next-access delay value.

According to the sequence length that is assigned to SPSCR, the RSPI makes up a sequence comprised of a part or all of SPCMDm register. The RSPI contains a pointer to the SPCMDm register that makes up the sequence. The value of this pointer can be checked by reading the SPSSR.SPCP[2:0] bits. When the SPCR.SPE bit is set to 1 and the RSPI function is enabled, the RSPI loads the pointer to the commands in SPCMD0, and incorporates the SPCMD0 settings into the transfer format at the beginning of serial transfer. The RSPI increments the pointer each time the next-access delay period for a data transfer ends. Upon completion of the serial transfer that corresponds to the final command comprising the sequence, the RSPI sets the pointer in SPCMD0, and in this manner the sequence is executed repeatedly.

(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

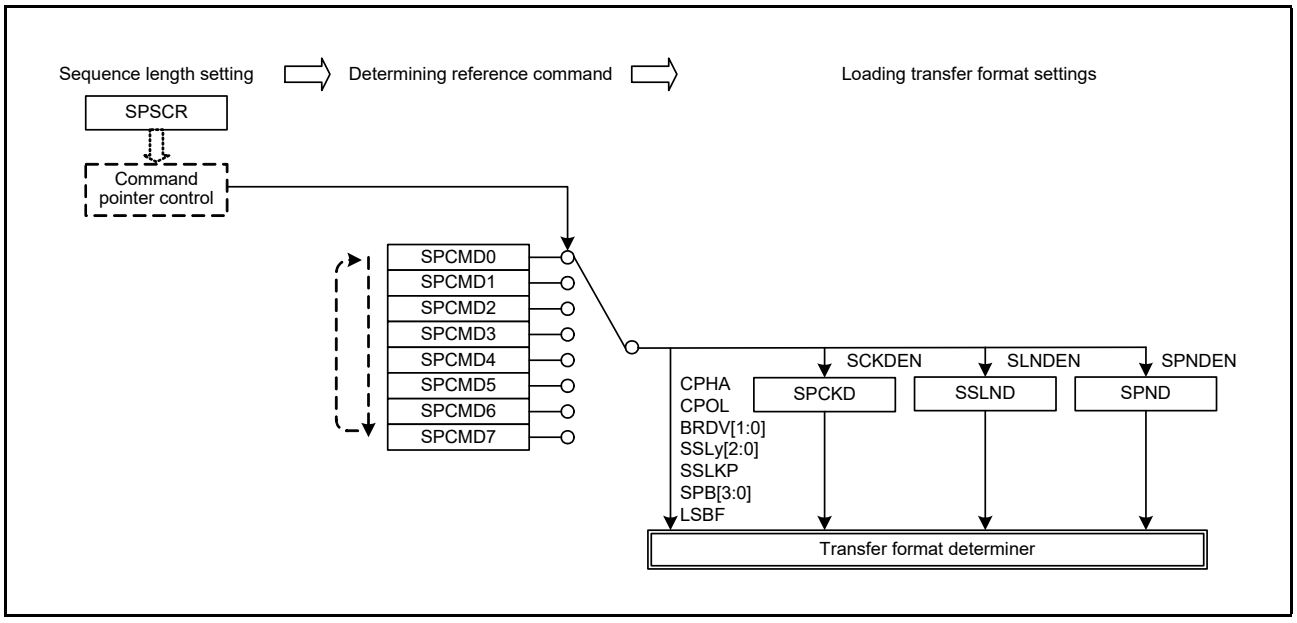


Figure 28.31 Procedure for Determining the Form of Serial Transfer in Master Mode

In this section, a frame is the combination of the data (SPDR) and the settings (SPCMDm).

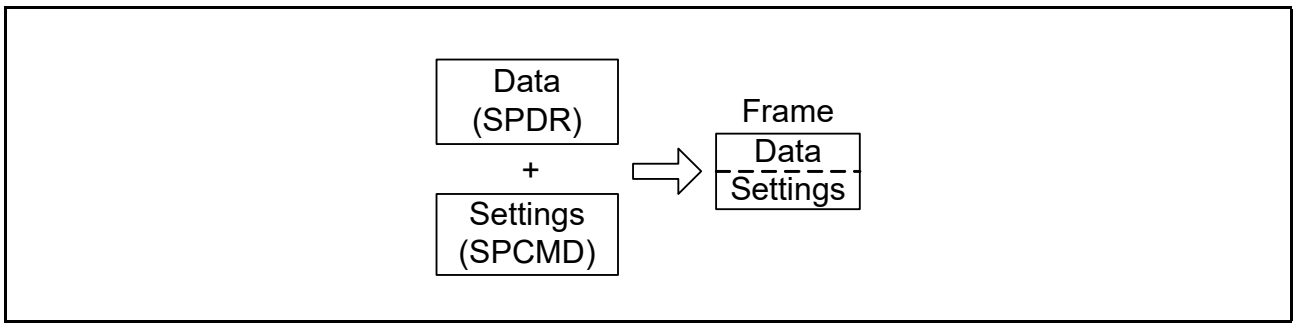


Figure 28.32 Concept of a Frame

Figure 28.33 shows the relationship between the command and the transmit and reception buffers in the sequence of operations specified by the settings in Table 28.4.

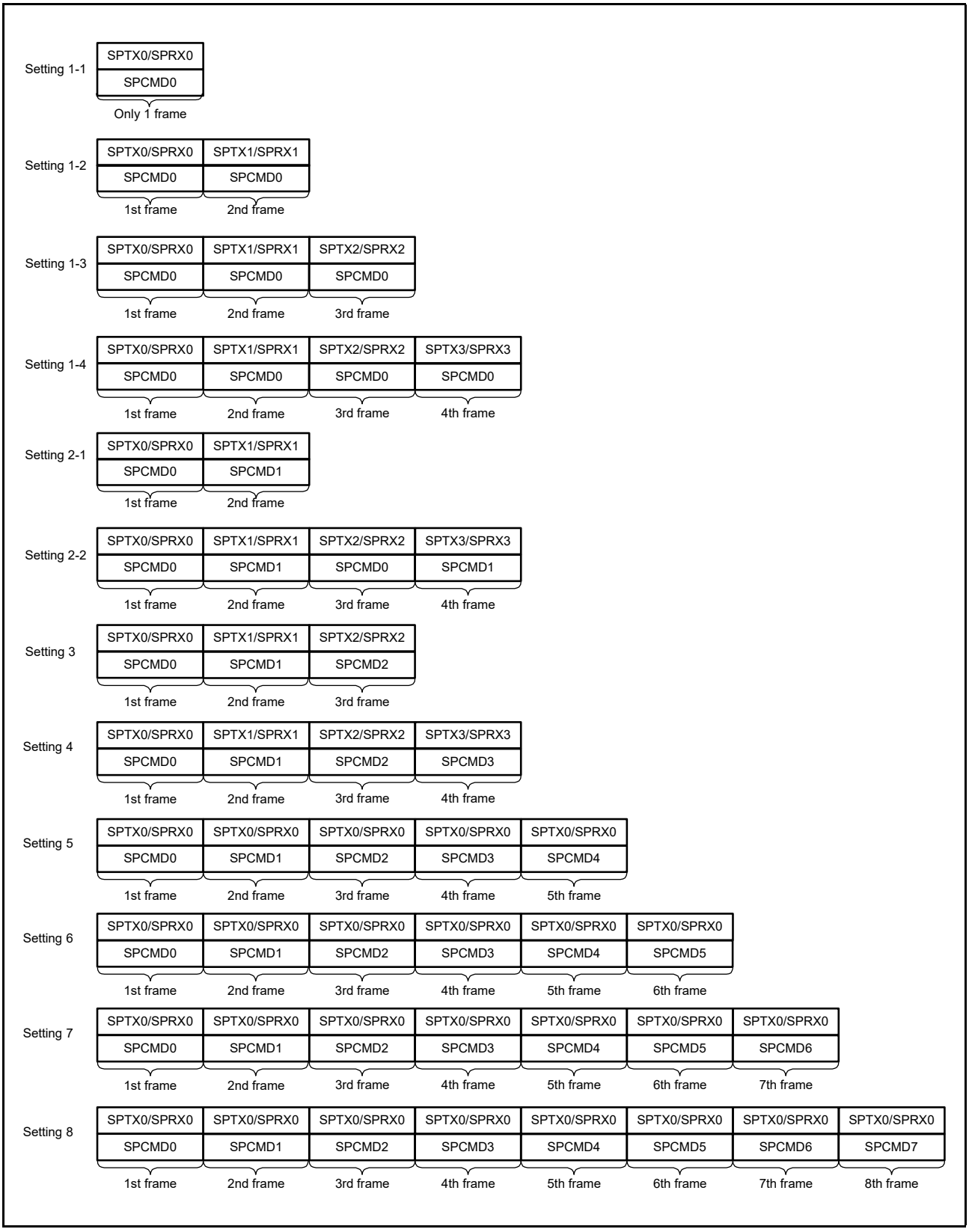


Figure 28.33 Correspondence between the RSPI Command Register and Transmission/Reception Buffers in Sequence Operations

(4) Burst Transfer

If the SPCMDm.SSLKP bit that the RSPI references during the current serial transfer is 1, the RSPI keeps the SSLyi signal level during the serial transfer until the beginning of the SSLyi signal assertion for the next serial transfer. If the SSLyi signal level for the next serial transfer is the same as the SSLyi signal level for the current serial transfer, the RSPI can execute continuous serial transfers while keeping the SSLyi signal assertion status (burst transfer).
Figure 28.34 shows an example of an SSLyi signal operation for the case where a burst transfer is implemented using SPCMD0 and SPCMD1 register settings. The text below explains the RSPI operations (1) to (7) as shown in Figure 28.34. It should be noted that the polarity of the SSLyi output signal depends on the SSLP register settings.

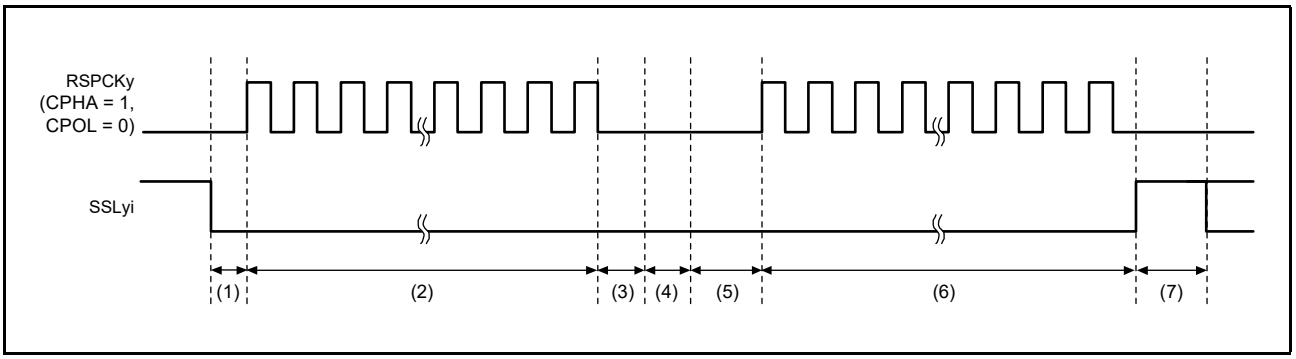


Figure 28.34 Example of Burst Transfer Operation Using SSLKP Bit

(1) Based on SPCMD0, the RSPI asserts the SSLyi signal and inserts RSPCK delays.
(2) The RSPI executes serial transfers according to SPCMD0.
(3) The RSPI inserts SSL negation delays.
(4) Since the SPCMD0.SSLKP bit is 1, the RSPI keeps the SSLyi signal value on SPCMD0. This period is sustained, at the shortest, for a period equal to the next-access delay of SPCMD0. If the shift register is empty after the passage of a minimum period, this period is sustained until the transmit data is stored in the shift register for the next transfer.
(5) Based on SPCMD1, the RSPI asserts the SSLyi signal and inserts RSPCK delays.
(6) The RSPI executes serial transfers according to SPCMD1.
(7) Because the SPCMD1.SSLKP bit is 0, the RSPI negates the SSLyi signal. In addition, a next-access delay is inserted according to SPCMD1.

If the SSLyi signal output settings in the SPCMDm register in which 1 is assigned to the SSLKP bit are different from the SSLyi signal output settings in the SPCMDm register to be used in the next transfer, the RSPI switches the SSLyi signal status to SSLyi signal assertion ((5) in Figure 28.34) corresponding to the command for the next transfer. Note that if such an SSLyi signal switching occurs, the slaves that drive the MISOy signal compete, and collision of signal levels may occur.
The RSPI in master mode references the SSLyi signal operation within the module for the case where the SSLKP bit is not used. Even when the SPCMDm.CPHA bit is 0, the RSPI can accurately start serial transfers by using the SSLyi signal assertion for the next transfer that is detected internally.
(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

(5) RSPCK Delay (t1)

The RSPCK delay value of the RSPI in master mode depends on the SPCMDm.SCKDEN bit setting and the SPCKD register setting. The RSPI determines the SPCMDm register to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SPCMDm.SCKDEN bit and SPCKD, as listed in Table 28.9. For a definition of RSPCK delay, refer to section 28.3.5, Transfer Format.

Table 28.9 Relationship among SCKDEN Bit, SPCKD, and RSPCK Delay Value

SPCMDm.SCKDEN Bit	SPCKD.SCKDL[2:0] Bits	RSPCK Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(6) SSL Negation Delay (t2)

The SSL negation delay value of the RSPI in master mode depends on the SPCMDm.SLNDEN bit setting and the SSLND register setting. The RSPI determines the SPCMDm register to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SPCMDm.SLNDEN bit and SSLND, as listed in Table 28.10. For a definition of SSL negation delay, refer to section 28.3.5, Transfer Format.

Table 28.10 Relationship among SLNDEN Bit, SSLND, and SSL Negation Delay Value

SPCMDm.SLNDEN Bit	SSLND.SLNDL[2:0] Bits	SSL Negation Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

(7) Next-Access Delay (t3)

The next-access delay value of the RSPI in master mode depends on the SPCMDm.SPNDEN bit setting and the SPND setting. The RSPI determines the SPCMDm register to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPCMDm.SPNDEN bit and SPND, as listed in Table 28.11. For a definition of next-access delay, refer to section 28.3.5, Transfer Format.

Table 28.11 Relationship among SPNDEN Bit, SPND, and Next-Access Delay Value

SPCMDm.SPNDEN Bit	SPND.SPNDL[2:0] Bits	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2 SERICLK
1	000	1 RSPCK + 2 SERICLK
	001	2 RSPCK + 2 SERICLK
	010	3 RSPCK + 2 SERICLK
	011	4 RSPCK + 2 SERICLK
	100	5 RSPCK + 2 SERICLK
	101	6 RSPCK + 2 SERICLK
	110	7 RSPCK + 2 SERICLK
	111	8 RSPCK + 2 SERICLK

(8) Initialization Flowchart

Figure 28.35 is a flowchart illustrating an example of initialization in SPI operation when the RSPI is used in master mode. For a description of how to set up the interrupt controller, DMAC, and I/O ports, refer to the descriptions given in the individual blocks.

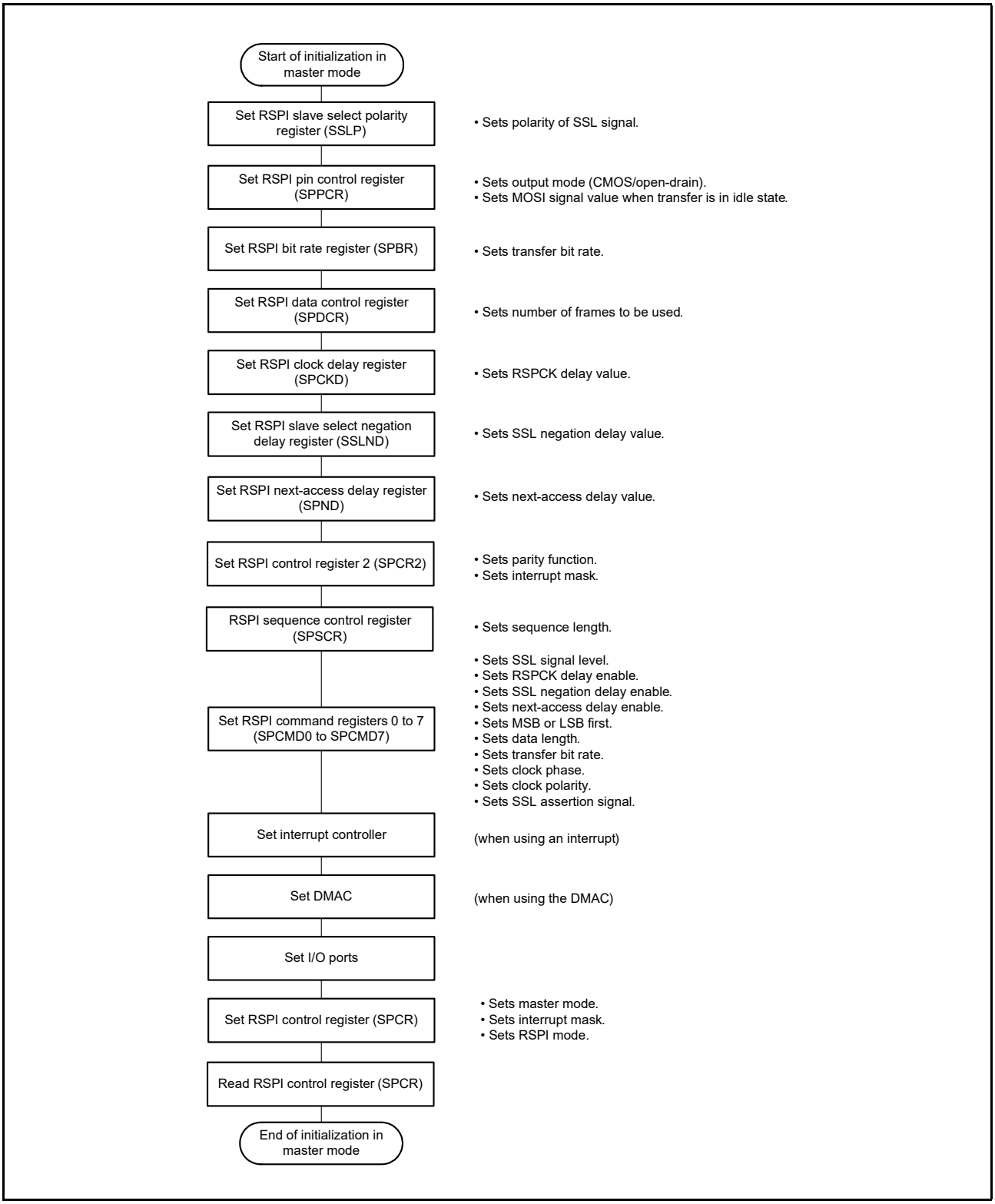


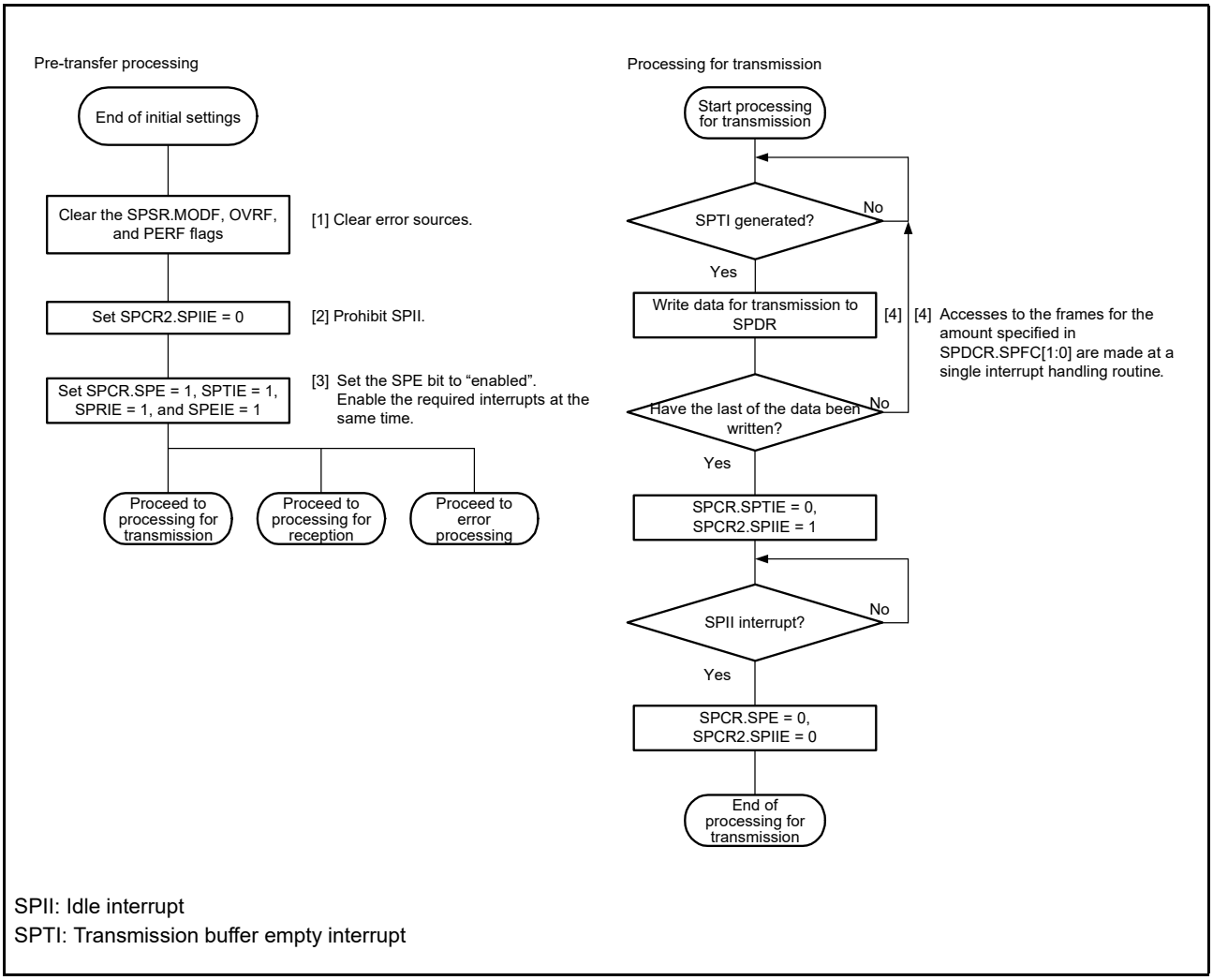
Figure 28.35 Example of Initialization Flowchart in Master Mode (SPI Operation)

(9) Software Processing Flow

Figure 28.36 to Figure 28.38 show examples of the flow of software processing.

(a) Transmit Processing Flow

When transmitting data, the CPU will be notified of the completion of data transmission after the last writing of data for transmission if the idle interrupt (SPII) is enabled.



SPII: Idle interrupt
SPTI: Transmission buffer empty interrupt

Figure 28.36 Flowchart in Master Mode (Transmission)

(b) Receive Processing Flow

The RSPI does not handle receive-only operation, so processing for transmission is required if reception is to proceed.

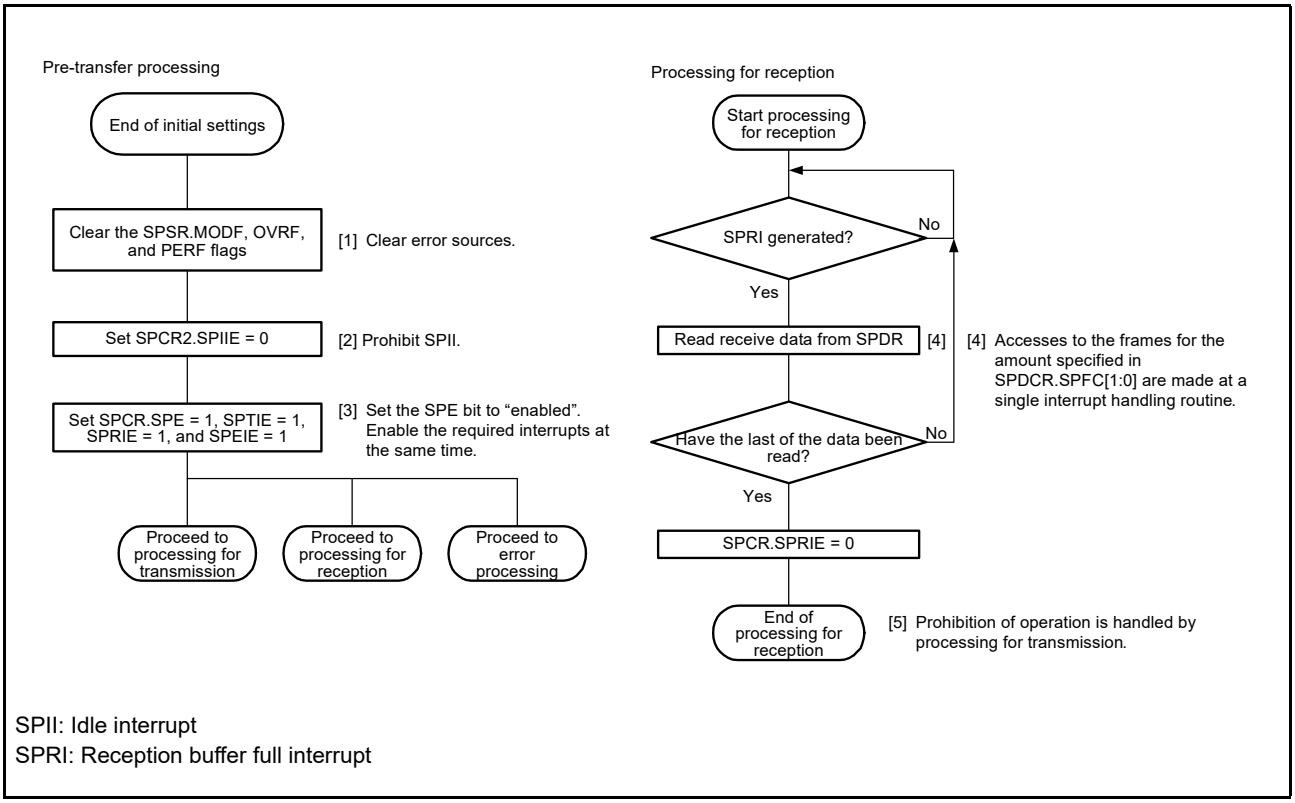
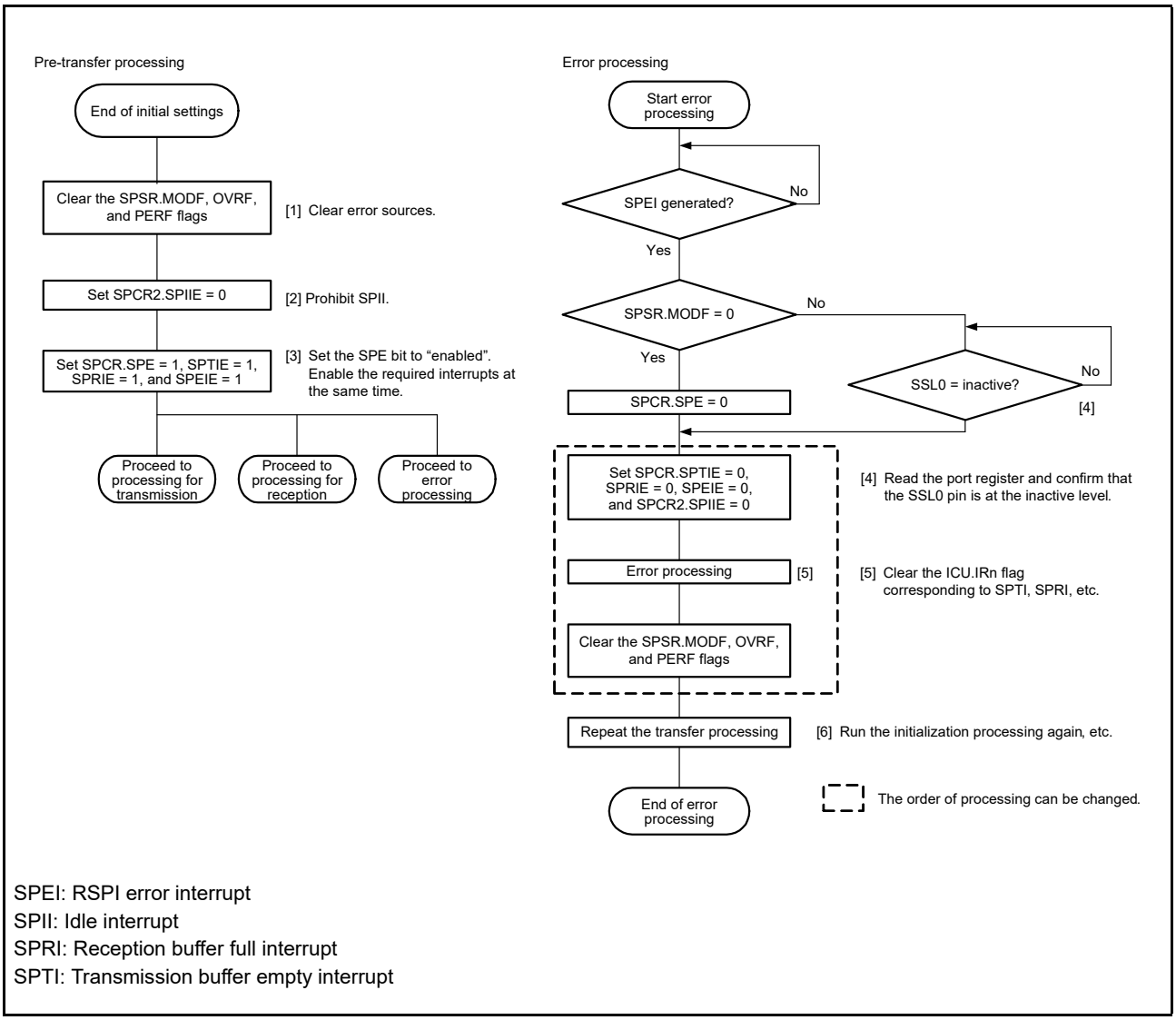


Figure 28.37 Flowchart in Master Mode (Reception)

(c) Flow of error processing

The RSPI has three types of error. When a mode fault error is generated, the SPCR.SPE bit is automatically cleared, stopping operations for transmission and reception. For errors from other sources, however, the SPCR.SPE bit is not cleared and operations for transmission and reception continue; accordingly, we recommend clearing of the SPCR.SPE bit to stop operations in the case of errors other than mode-fault errors. Not doing so will lead to updating of the SPSSR.SPECM[2:0] bits.

When an error occurs, clear the corresponding flag of the IRQ status register from within the error processing routine. If this is not done, the corresponding interrupt request flag in the IRQ status register may continue to indicate a transmission buffer empty interrupt (SPTI) or reception buffer full interrupt (SPRI) request. If the reception buffer full interrupt (SPRI) request is indicated, read the reception buffer and initialize the sequencer in the RSPI.



SPEI: RSPI error interrupt
SPII: Idle interrupt
SPRI: Reception buffer full interrupt
SPTI: Transmission buffer empty interrupt

Figure 28.38 Flowchart for Master Mode (Error Processing)

28.3.10.2 Slave Mode Operation

(1) Starting a Serial Transfer

If the SPCMD0.CPHA bit is 0, when detecting an SSLy0 input signal assertion, the RSPI needs to start driving valid data to the MISOy output signal. For this reason, when the CPHA bit is 0, the assertion of the SSLy0 input signal triggers the start of a serial transfer.
If the CPHA bit is 1, when detecting the first RSPCKy edge in an SSLy0 signal asserted condition, the RSPI needs to start driving valid data to the MISOy output signal. For this reason, when the CPHA bit is 1, the first RSPCKy edge in an SSLy0 signal asserted condition triggers the start of a serial transfer.
When detecting the start of a serial transfer in a condition in which the shift register is empty, the RSPI changes the status of the shift register to “full”, so that data cannot be copied from the transmission buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, the RSPI leaves the status of the shift register unchanged, in the full state.
Irrespective of CPHA bit setting, the timing at which the RSPI starts driving of the MISOy output signal is the SSLy0 signal assertion timing. The data which is output by the RSPI is either valid or invalid, depending on the CPHA bit setting.
For details on the RSPI transfer format, refer to section 28.3.5, Transfer Format. The polarity of the SSLy0 input signal depends on the setting of the SSLP.SSL0P bit.
(y = 0, 1 (for all channels))

(2) Terminating a Serial Transfer

Irrespective of the SPCMD0.CPHA bit, the RSPI terminates the serial transfer after detecting an RSPCKy edge corresponding to the final sampling timing. When free space is available in the reception buffer, upon termination of serial transfer the RSPI copies received data from the shift register to the reception buffer of the RSPI data register (SPDR). Upon termination of a serial transfer the RSPI changes the status of the shift register to “empty”, regardless of the reception buffer state. A mode fault error occurs if the RSPI detects an SSLy0 input signal negation from the beginning of serial transfer to the end of serial transfer (refer to section 28.3.8, Error Detection).
The final sampling timing changes depending on the bit length of transfer data. In slave mode, the RSPI data length depends on the SPCMD0.SPB[3:0] bit setting. The polarity of the SSLy0 input signal depends on the SSLP.SSL0P bit setting.
For details on the RSPI transfer format, refer to section 28.3.5, Transfer Format.
(y = 0, 1 (for all channels))

(3) Notes on Single-Slave Operations

If the SPCMD0.CPHA bit is 0, the RSPI starts serial transfers when it detects the assertion edge for an SSLy0 input signal. In the type of configuration shown in Figure 28.7 as an example, if the RSPI is used in single-slave mode, the SSLy0 signal is always fixed at the active state. Therefore, when the CPHA bit is set to 0, the RSPI cannot correctly start a serial transfer. To correctly execute transmit/receive operations by the RSPI in slave mode in a configuration in which the SSLy0 input signal is fixed at the active state, the CPHA bit should be set to 1. If there is a need for setting the CPHA bit to 0, the SSLy0 input signal should not be fixed.
(y = 0, 1 (for all channels))

(4) Burst Transfer

If the SPCMD0.CPHA bit is 1, continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSLy0 input signal. If the CPHA bit is 1, the period from the first RSPCKy edge to the sampling timing for the reception of the final bit in an SSLy0 signal active state corresponds to a serial transfer period. Even when the SSLy0 input signal remains at the active level, the RSPI can accommodate burst transfers because it can detect the start of an access.
If the CPHA bit is 0, the second and subsequent serial transfers during burst transfer cannot be executed correctly.

(5) Initialization Flowchart

Figure 28.39 is a flowchart illustrating an example of initialization in SPI operation when the RSPI is used in slave mode. For a description of how to set up the interrupt controller, DMAC, and I/O ports, refer to the descriptions given in the individual blocks.

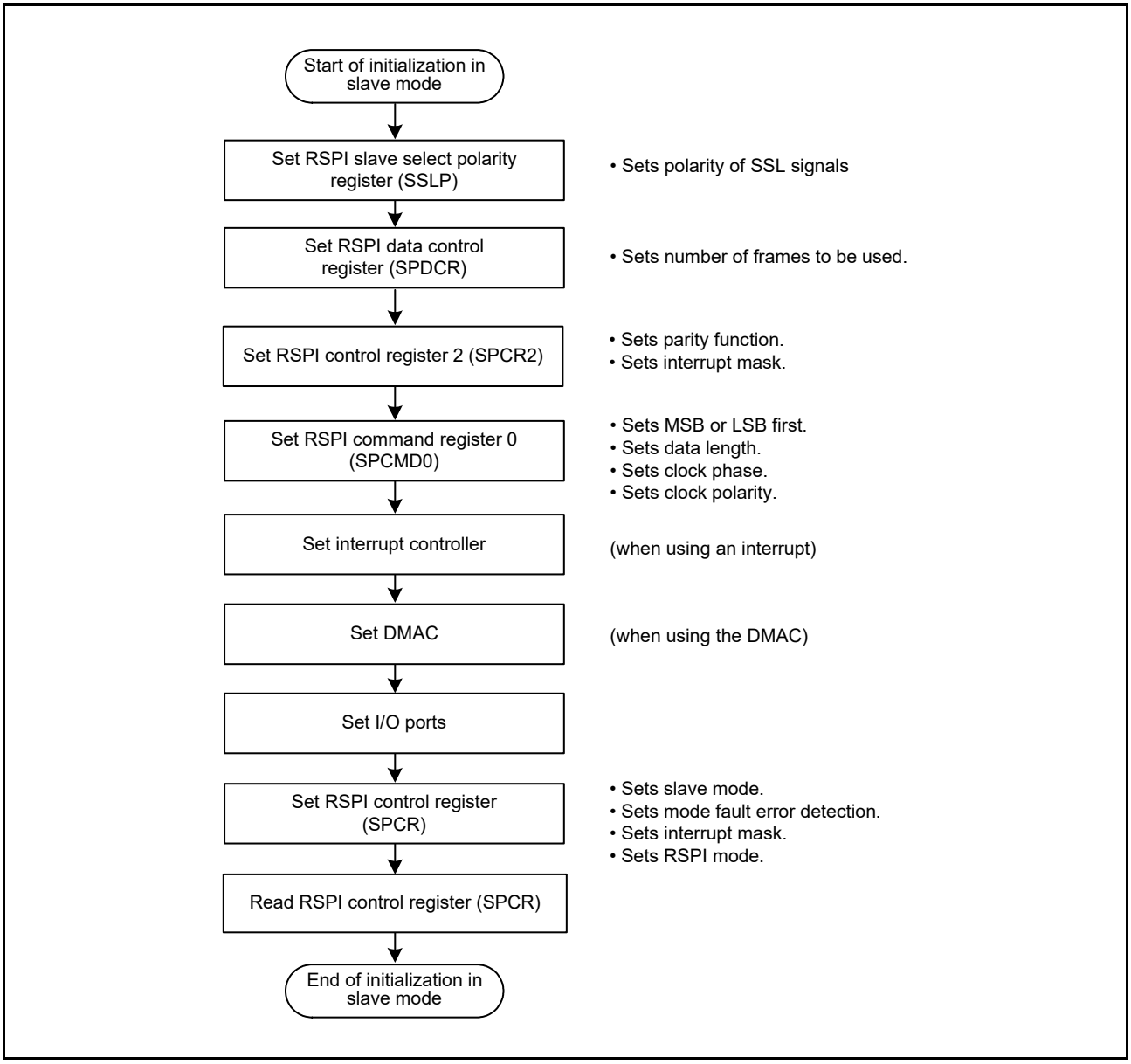


Figure 28.39 Example of Initialization Flowchart in Slave Mode (SPI Operation)

(6) Software Processing Flow

Figure 28.40 to Figure 28.42 show examples of the flow of software processing.

(a) Transmit Processing Flow

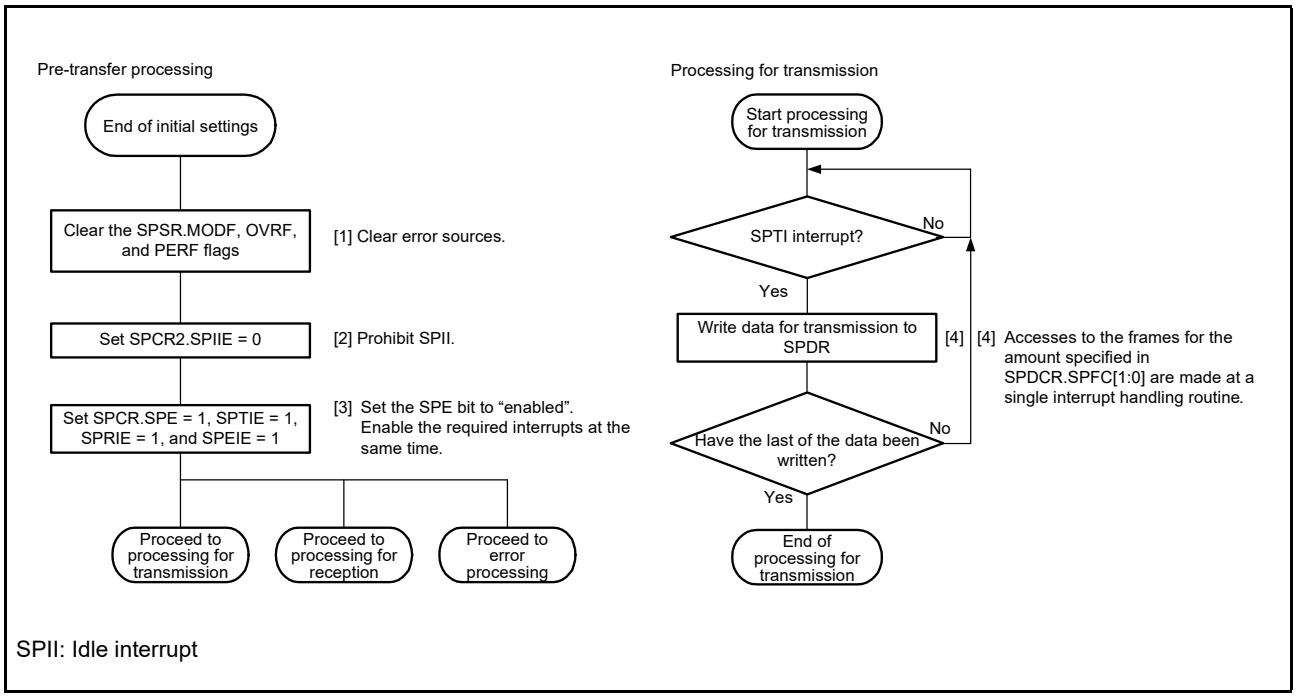


Figure 28.40 Flowchart in Slave Mode (Transmission)

(b) Receive Processing Flow

The RSPI does not handle receive-only operation, so processing for transmission is required if reception is to proceed.

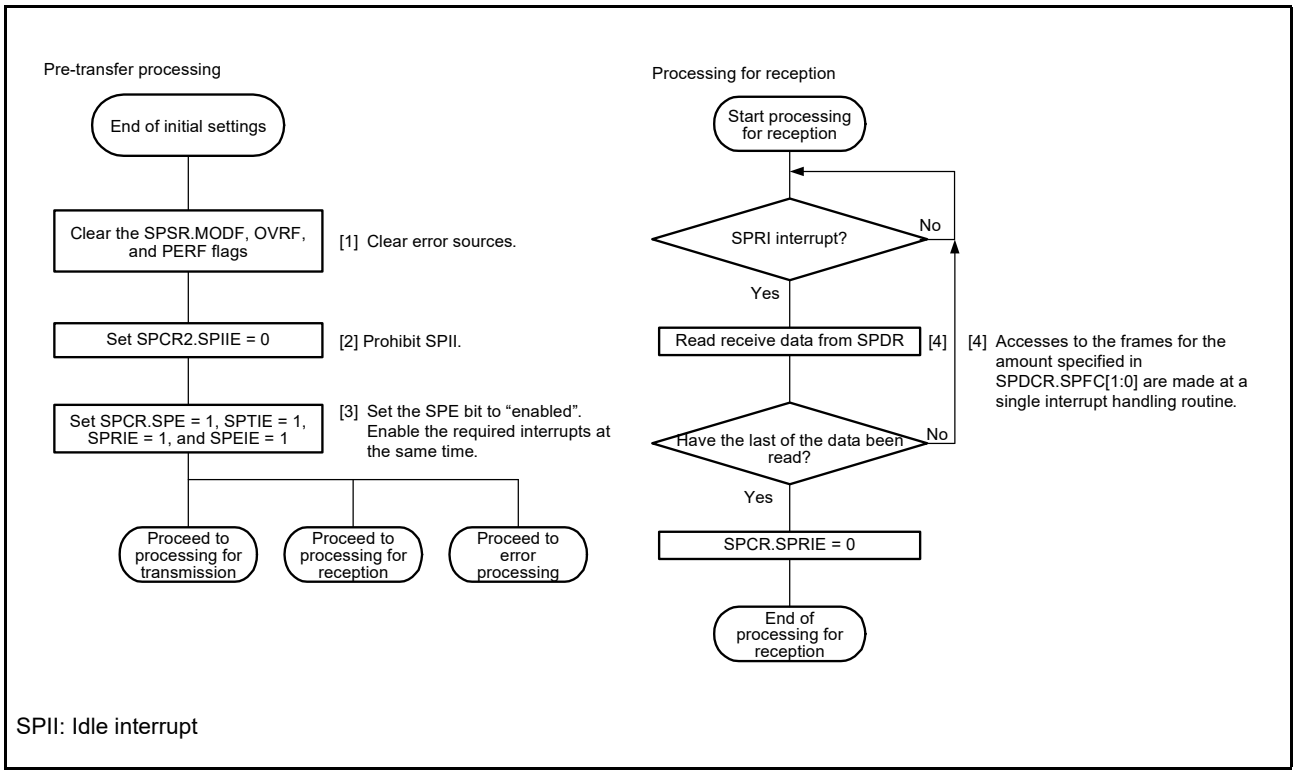
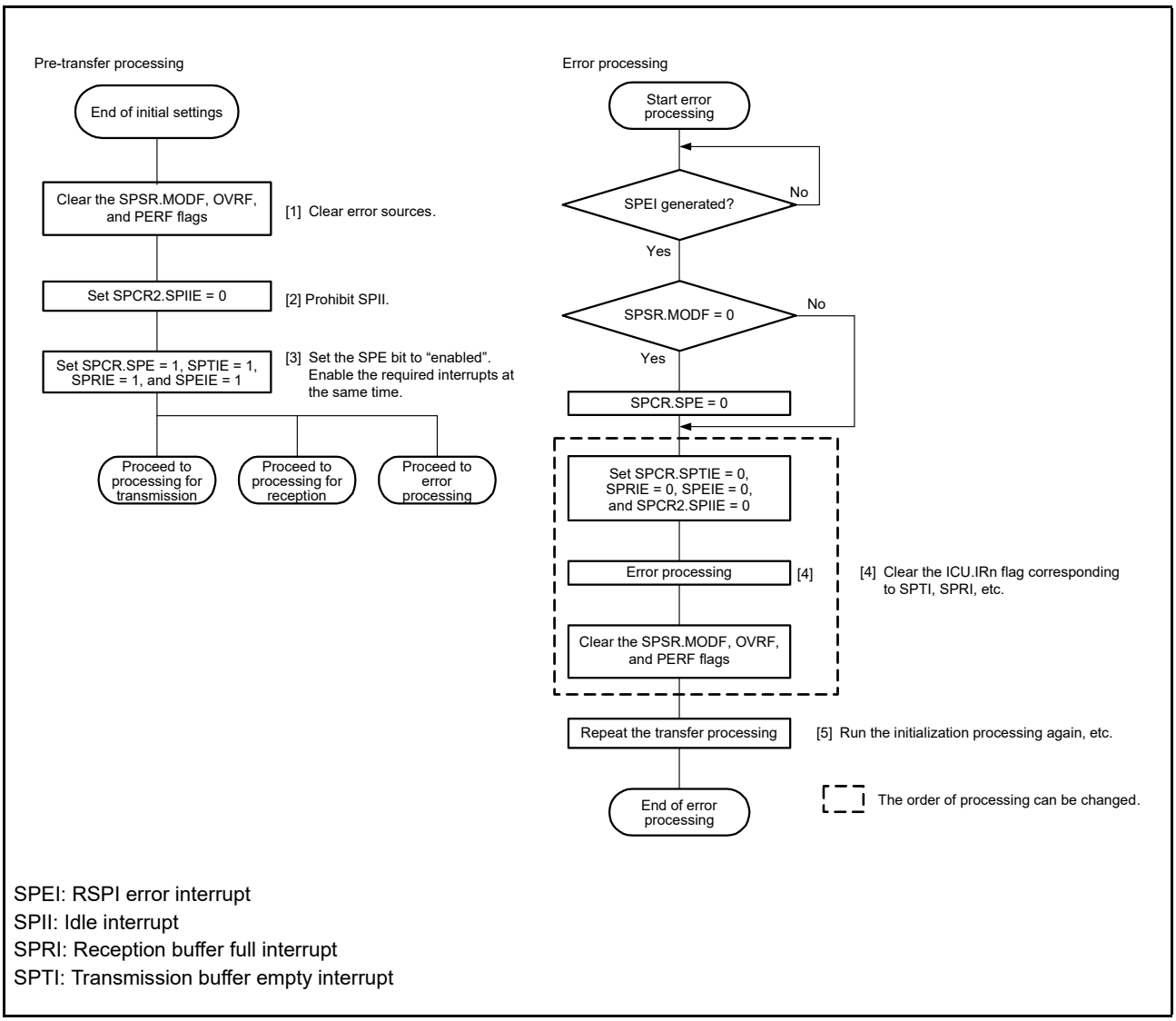


Figure 28.41 Flowchart in Slave Mode (Reception)

(c) Flow of error processing

In slave operation, even when a mode-fault error is generated, the SPSR.MODF flag can be cleared without de-asserting the pin.

When an error occurs, clear the corresponding flag of the IRQ status register (IRQSn) from within the error processing routine. If this is not done, the corresponding interrupt request flag in the IRQ status register (IRQSn) may continue to indicate a transmission buffer empty interrupt (SPTI) or reception buffer full interrupt (SPRI) request. If the reception buffer full interrupt (SPRI) request is indicated, read the reception buffer and initialize the sequencer in the RSPI.



SPEI: RSPI error interrupt
SPII: Idle interrupt
SPRI: Reception buffer full interrupt
SPTI: Transmission buffer empty interrupt

Figure 28.42 Flowchart for Slave Mode (Error Processing)

28.3.11 Clock Synchronous Operation

Setting the SPMS bit in the RSPI control register (SPCR) to 1 selects clock synchronous operation of the RSPI. In clock synchronous operation, the SSLyi pin is not used, and the three pins of RSPCKy, MOSIy, and MISOy handle communications. The SSLyi pin is available as I/O port pins.
Although clock synchronous operation does not require use of the SSLyi pin, operation of the module is the same as in SPI operation. That is, in both master and slave operations, communications can be performed with the same flow as in SPI operation. However, mode fault errors are not detected because the SSLyi pin is not used.
Furthermore, operation should not be performed if clock synchronous operation proceeds when the SPCMDm.CPHA bit is set to 0 in slave mode (SPCR.MSTR = 0).
(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

28.3.11.1 Master Mode Operation

(1) Starting a Serial Transfer

The RSPI updates the data in the transmission buffer (SPTX) of SPDR when data is written to the RSPI data register (SPDR) with the transmission buffer being empty (data for the next transfer is not set). When the shift register is empty after the number of frames set in the SPDCR.SPFC[1:0] bits are written to the SPDR, the RSPI copies data from the transmission buffer to the shift register and starts serial transmission. Upon copying transmit data to the shift register, the RSPI changes the status of the shift register to “full”, and upon termination of serial transfer, it changes the status of the shift register to “empty”. The status of the shift register cannot be referenced.
For details on the RSPI transfer format, refer to section 28.3.5, Transfer Format.
However, transfer in clock synchronous operation is conducted without the SSLy0 output signal (y = 0, 1 (for all channels)).

(2) Terminating a Serial Transfer

The RSPI terminates the serial transfer after transmitting an RSPCKy edge corresponding to the sampling timing. If free space is available in the reception buffer (SPRX), upon termination of serial transfer, the RSPI copies data from the shift register to the reception buffer of the RSPI data register (SPDR).
It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the RSPI data length depends on the SPCMDm.SPB[3:0] bit setting.
For details on the RSPI transfer format, refer to section 28.3.5, Transfer Format.
However, transfer in clock synchronous operation is conducted without the SSLy0 output signal (y = 0, 1 (for all channels)).

(3) Sequence Control

The transfer format employed in master mode is determined by SPSCR, SPCMDm, SPBR, SPCKD, SSLND, and SPND registers. Although the SSLyi signals are not output in clock synchronous operation, these settings are valid.
SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by the RSPI in master mode. The following items are set in SPCMDm register: SSLyi output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCKy polarity/phase, whether SPCKD is to be referenced, whether SSLND is to be referenced, and whether SPND is to be referenced. SPBR holds some of the bit rate settings; SPCKD, an RSPI clock delay value; SSLND, an SSL negation delay; and SPND, a next-access delay value.
According to the sequence length that is assigned to SPSCR, the RSPI makes up a sequence comprised of a part or all of SPCMDm register. The RSPI contains a pointer to the SPCMDm register that makes up the sequence. The value of this pointer can be checked by reading the SPSSR.SPCP[2:0] bits. When the SPCR.SPE bit is set to 1 and the RSPI function is enabled, the RSPI loads the pointer to the commands in SPCMD0 register, and incorporates the SPCMD0 register setting into the transfer format at the beginning of serial transfer. The RSPI increments the pointer each time the next-access delay period for a data transfer ends. Upon completion of the serial transfer that corresponds to the final command comprising the sequence, the RSPI sets the pointer in SPCMD0 register, and in this manner the sequence is executed repeatedly.
(i = 0 to 3; m = 0 to 7; y = 0, 1 (for all channels))

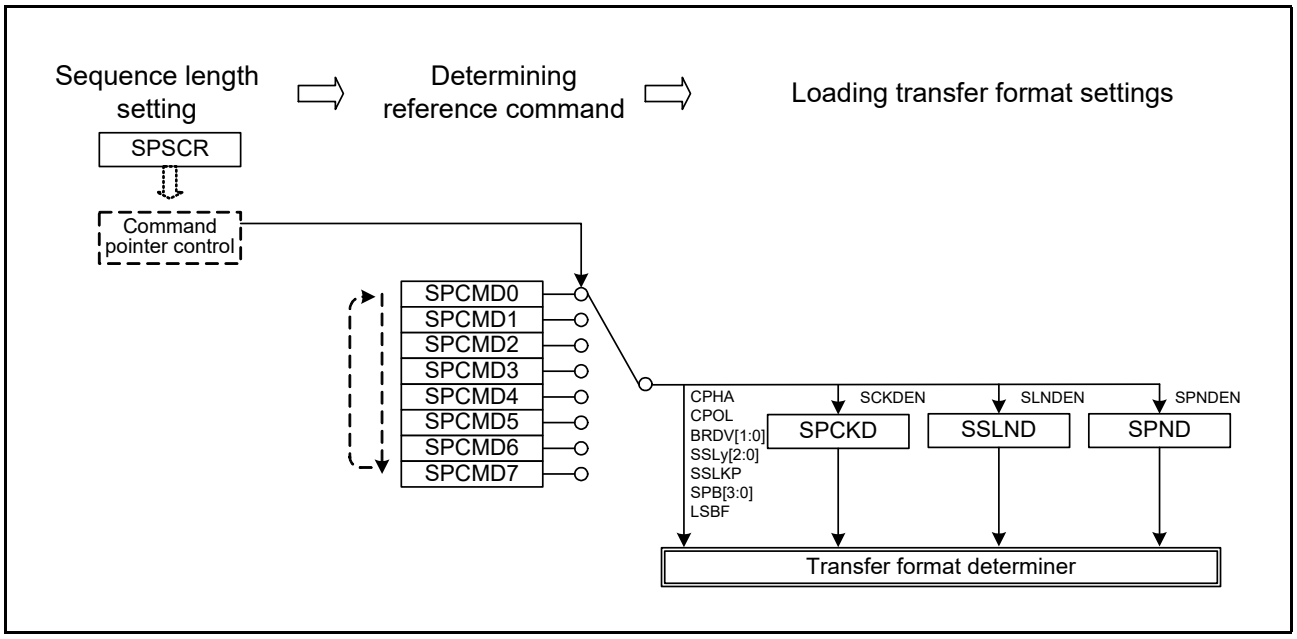


Figure 28.43 Procedure for Determining the Form of Serial Transmission in Master Mode

In this section, a frame is the combination of the data (SPDR) and the settings (SPCMDm) (m = 0 to 7).

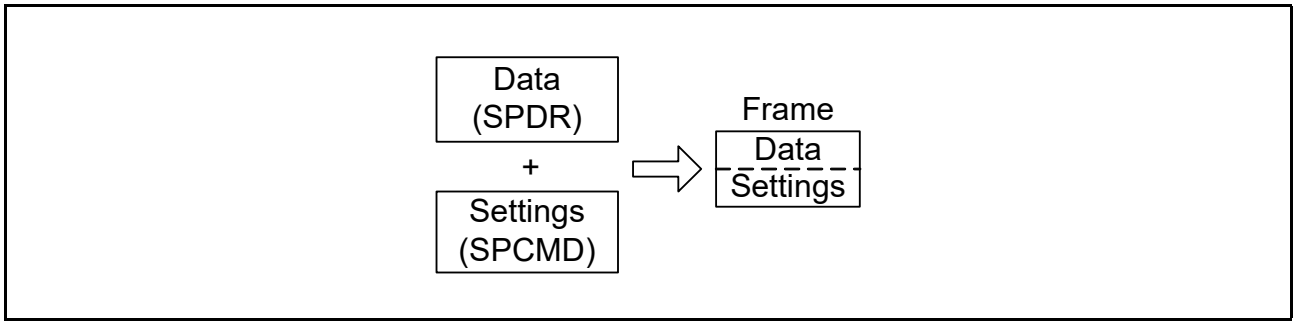


Figure 28.44 Concept of a Frame

Figure 28.45 shows the relationship between the command and the transmit and reception buffers in the sequence of operations specified by the settings in Table 28.4.

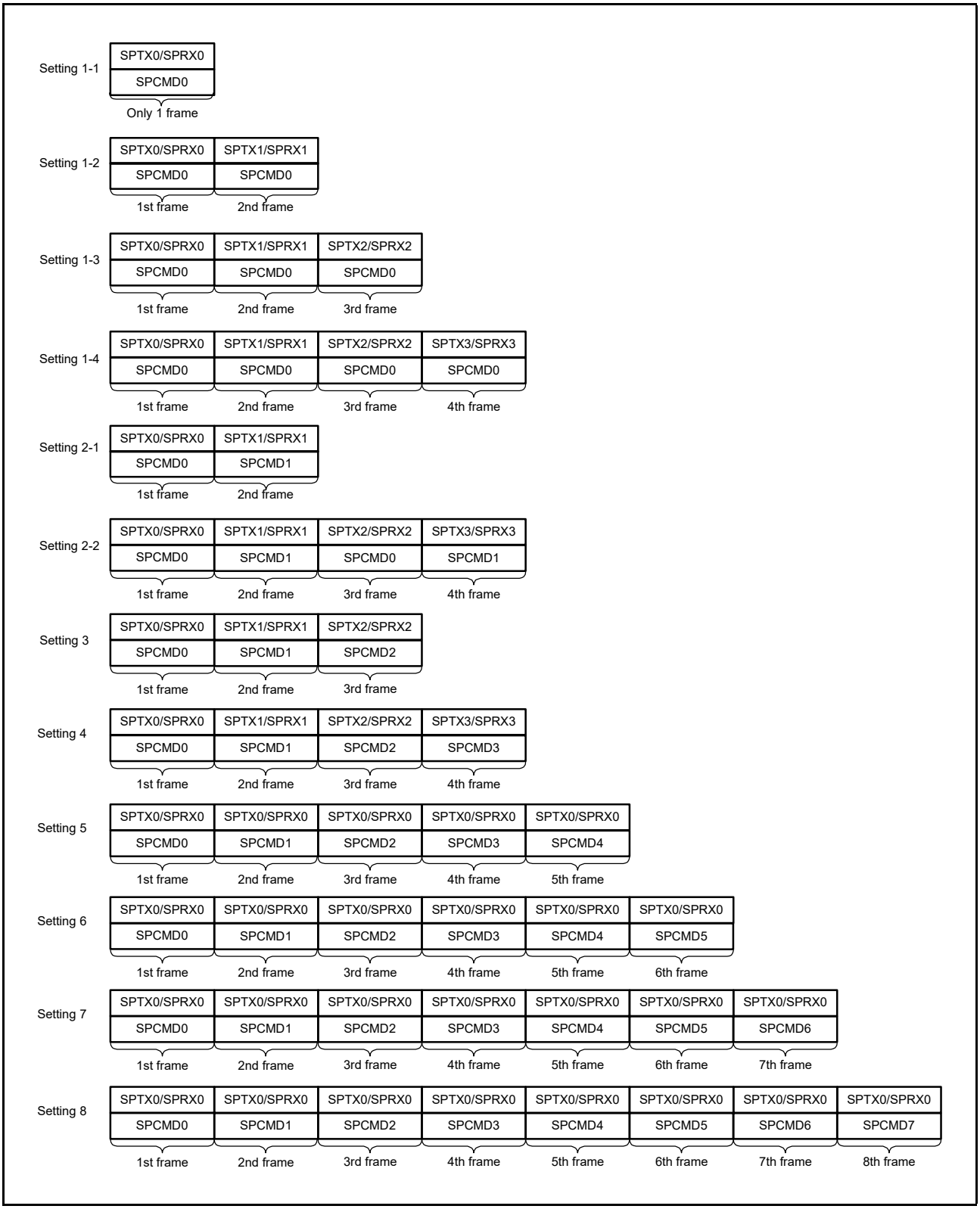


Figure 28.45 Correspondence between the RSPI Command Register and Transmission/Reception Buffers in Sequence Operations

(4) Initialization Flowchart

Figure 28.46 is a flowchart illustrating an example of initialization in clock synchronous operation when the RSPI is used in master mode. For a description of how to set up the interrupt controller, DMAC, and I/O ports, refer to the descriptions given in the individual blocks.

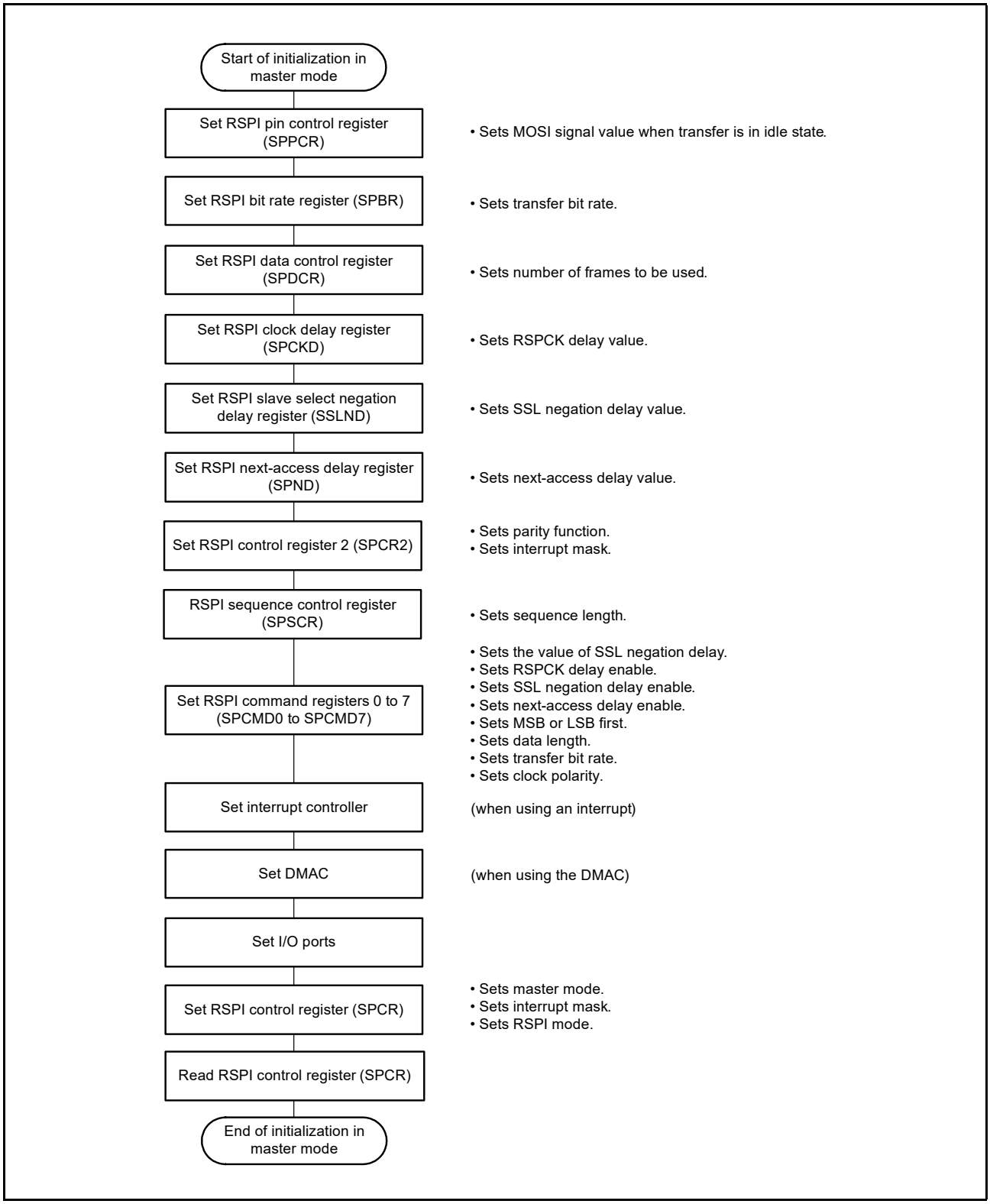


Figure 28.46 Example of Initialization Flowchart in Master Mode (Clock Synchronous Operation)

(5) Flow of Software Processing

Software processing during clock-synchronous master operation is the same as that for SPI master operation. For details, refer to section 28.3.10.1, (9) Software Processing Flow. Note that mode-fault errors will not occur.

28.3.11.2 Slave Mode Operation

(1) Starting a Serial Transfer

When the SPCR.SPMS bit is 1, the first RSPCKy edge triggers the start of a serial transfer in the RSPI.
When detecting the start of a serial transfer in a condition in which the shift register is empty, the RSPI changes the status of the shift register to “full”, so that data cannot be copied from the transmission buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, the RSPI keeps the status of the shift register unchanged, in the full state.
When the SPMS bit is 1, the RSPI always drives the MISOy output signal.
For details on the RSPI transfer format, refer to section 28.3.5, Transfer Format.
It should be noted that the SSL0 input signal is not used in clock synchronous operation (y = 0, 1 (for all channels)).

(2) Terminating a Serial Transfer

The RSPI terminates the serial transfer after detecting an RSPCKy edge corresponding to the final sampling timing. When free space is available in the reception buffer, upon termination of serial transfer the RSPI copies received data from the shift register to the reception buffer of the RSPI data register (SPDR). Upon termination of a serial transfer the RSPI changes the status of the shift register to “empty”. The final sampling timing changes depending on the bit length of transfer data. In slave mode, the RSPI data length depends on the SPCMD0.SPB[3:0] bit setting. For details on the RSPI transfer format, refer to section 28.3.5, Transfer Format. (y = 0, 1 (for all channels))

(3) Initialization Flowchart

Figure 28.47 is a flowchart illustrating an example of initialization in clock synchronous operation when the RSPI is used in slave mode. For a description of how to set up the interrupt controller, DMAC, and I/O ports, refer to the descriptions given in the individual blocks.

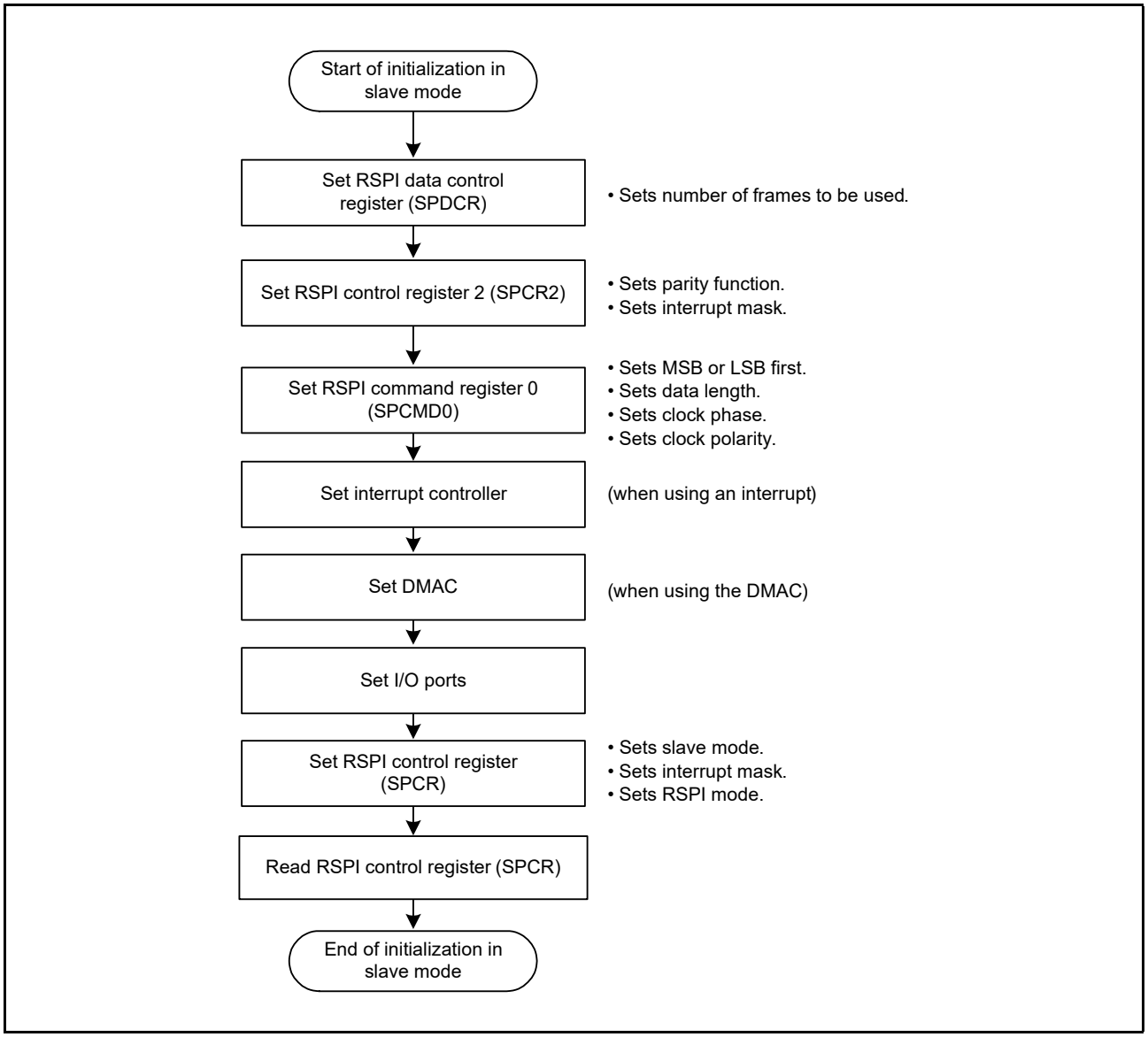


Figure 28.47 Example of Initialization Flowchart in Slave Mode (Clock Synchronous Operation)

(4) Flow of Software Processing

Software processing during clock-synchronous slave operation is the same as that for SPI slave operation. For details, refer to section 28.3.10.2, (6) Software Processing Flow. Note that mode-fault errors will not occur.

28.3.12 Loopback Mode

When 1 is written to the SPPCR.SPLP2 bit or SPPCR.SPLP bit, the RSPI shuts off the path between the MISOy pin and the shift register if the SPCR.MSTR bit is 1, and between the MOSIy pin and the shift register if the SPCR.MSTR bit is 0, and connects the input path and output path of the shift register. The RSPI does not shut off the path between the MOSIy pin and the shift register if the SPCR.MSTR bit is 1, and between the MISOy pin and the shift register if the SPCR.MSTR bit is 0. This is called loopback mode. When a serial transfer is executed in loopback mode, the transmit data for the RSPI or the inversed transmit data becomes the received data for the RSPI.
Table 28.12 lists the relationship among the SPLP2 and SPLP bits and the received data. Figure 28.48 shows the configuration of the shift register I/O paths for the case where the RSPI in master mode is set in loopback mode (SPPCR.SPLP2 = 0, SPPCR.SPLP = 1) (y = 0, 1 (for all channels)).

Table 28.12 SPLP2 and SPLP Bit Settings and Received Data

SPPCR.SPLP2 Bit	SPPCR.SPLP Bit	Received Data
0	0	Input data from the MOSIy pin or MISOy pin
0	1	Inversed transmit data
1	0	Transmit data
1	1	Transmit data

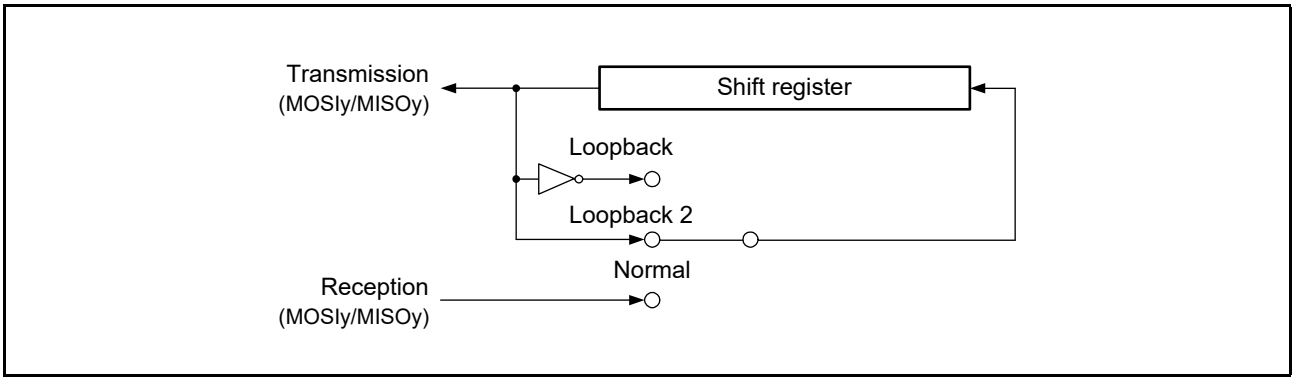


Figure 28.48 Configuration of Shift Register I/O Paths in Loopback Mode (Master Mode)

28.3.13 Self-Diagnosis of Parity Bit Function

The parity circuit consists of a parity bit adding unit used for transmit data and an error detecting unit used for received data. In order to detect defects in the parity bit adding unit and error detecting unit of the parity circuit, self-diagnosis is executed for the parity circuit following the flowchart shown in Figure 28.49.

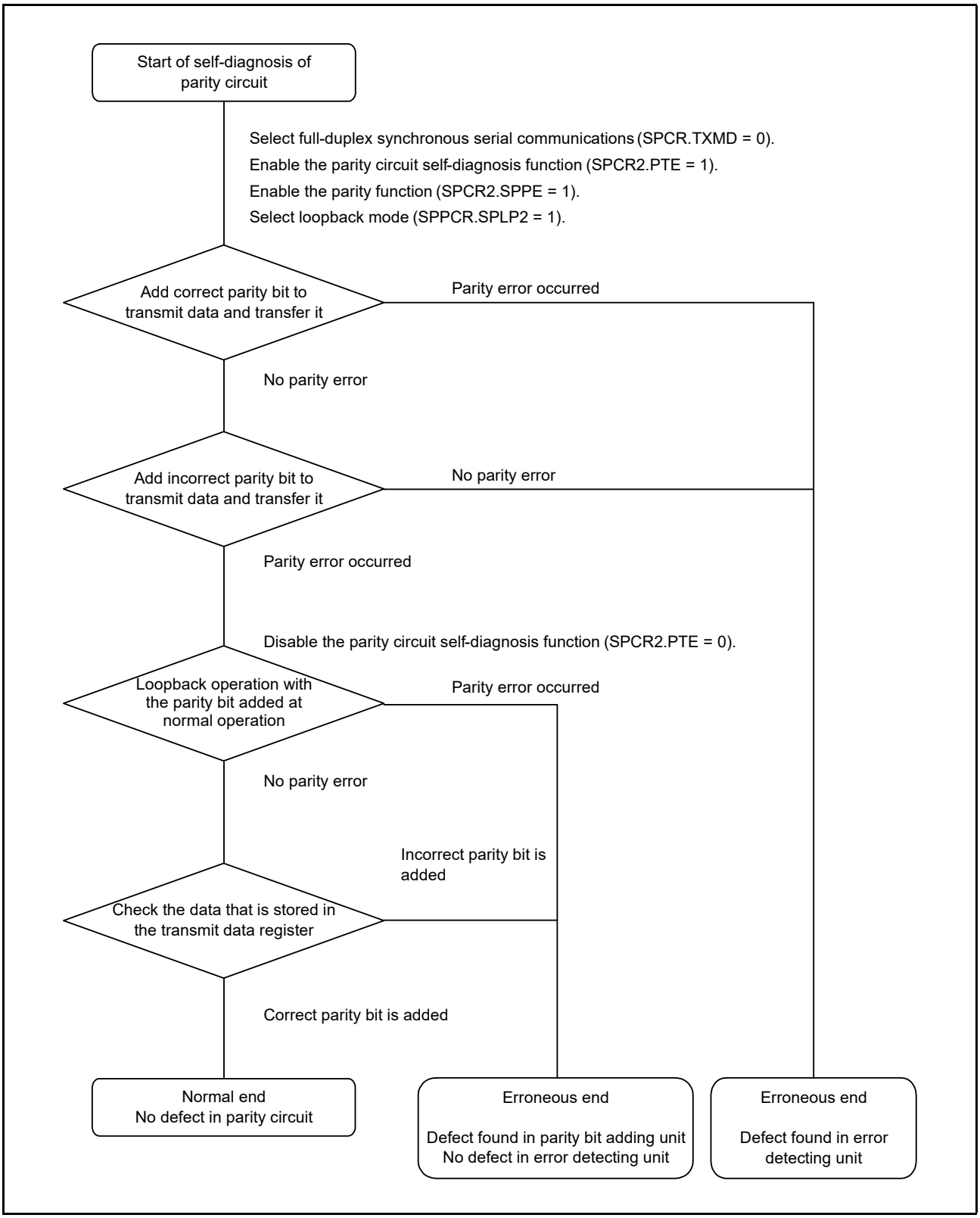


Figure 28.49 Flowchart for Self-Diagnosis of Parity Circuit

28.3.14 Interrupt Sources

The RSPI has interrupt sources of reception buffer full, transmission buffer empty, mode fault, overrun, parity error, and RSPI idle. In addition, the DMAC can be activated by the reception buffer full or transmission buffer empty interrupt to perform data transfer.

Since the common vector address is allocated to interrupt requests due to mode-fault, overrun, and parity errors, the actual interrupt source must be determined from the flags. Interrupt sources for the RSPI are listed in Table 28.13. An interrupt is generated on satisfaction of an interrupt condition in Table 28.13. Clear the reception buffer full and transmission buffer empty sources through data transfer.

When using the DMAC to perform data transmission/reception, the DMAC must be set up first to be in a status in which transfer is enabled before making the RSPI settings. For the method for setting the DMAC, refer to section 14, DMA Controller (DMACAa).

Table 28.13 Interrupt Sources of RSPI

Interrupt Source	Symbol	Interrupt Condition	DMAC/DTC Activation
Reception buffer full	SPRI	The reception buffer becomes full while the SPCR.SPRIE bit is 1.	Possible
Transmission buffer empty	SPTI	The transmission buffer becomes empty while the SPCR.SPTIE bit is 1.	Possible
RSPI errors (mode fault, overrun and parity error)	SPEI	The SPSR.MODF, OVRF, or PERF flag is set to 1 while the SPCR.SPEIE bit is 1.	Impossible
RSPI idle	SPII	The SPSR.IDLNF flag is set to 0 while the SPCR2.SPIIE bit is 1.	Impossible

28.4 Link Operation by Event Linking (only for RSPI channel 0)

The event link controller (ELC) is capable of producing the following event output signals. The event link output signal is output regardless of the setting of the corresponding interrupt enable bit (SPCR.SPEIE, SPCR.SPTIE or SPCR.SPRIE).

28.4.1 Reception Buffer Full Event Output

This event signal is output when received data have been transferred from the shift register to the SPDR on completion of serial transfer.

28.4.2 Transmission Buffer Empty Event Output

This event signal is output when data for transmission have been transferred from the transmission buffer to the shift register and when the value of the SPE bit has changed from 0 to 1.

28.4.3 Mode Fault, Overrun, or Parity Error Event Output

(1) Mode Fault

Table 28.14 lists the occurrence conditions of a mode fault event.

Table 28.14 Occurrence Conditions of Mode Fault Event

	SPCR.MODFEN Bit	SSLy0 Pin (y = 0, 1)	Remarks
Master (SPCR.MSTR bit = 1)	1	Active	When the setting of the SPCR.SPMS bit is 0 while the MSTR and SPCR.MODFEN bits are 1, mode fault error, overrun error, and parity error event output cannot be used. Do not set the ELSRn register to 52h.
Slave (SPCR.MSTR bit = 0)	1	Not active	Event is output only when the pin is deactivated during transmission.

(2) Overrun

The condition for this event signal being output in response to an overrun is completion of serial transfer while the reception buffer contains data that have not been read and the value of the SPCR.TXMD bit is 0, in which case the OVRF flag is set to 1.

(3) Parity Error

The condition for this event signal being output in response to a parity error is detection of a parity error on completion of serial transfer while the value of the TXMD bit in SPCR is 0 and the value of the SPPE bit in SPCR2 is 1.

28.4.4 RSPI Idle Event Output

(1) In Master Mode

In master mode, an event is output when the condition for setting the IDLNF flag (RSPI idle flag) to 0 is satisfied.

(2) In Slave Mode

In slave mode, an event is output when the SPE bit in the SPCR is set to 0 (RSPI is initialized).

28.4.5 Transmission-Completed Event Output

During both SPI operation and clock synchronous operation in master mode, an event is output under the condition for setting the IDLNF flag (RSPI idle flag) from 1 to 0.

Table 28.15 Conditions for Generation of a Transmission-Completed Event (Slave)

	Transmission Buffer State	Shift Register State	Others
SPI operation (SPCR.SPMS = 0)	Empty	Empty	Negation of SSL0 input
Clock synchronous operation (SPCR.SPMS = 1)	Empty	Empty	Edge detection of the last RSPCK

Regardless of whether the operation is in master mode or slave mode, an event is not output if 0 is written to the SPCR.SPE bit in transmission or the SPCR.SPE bit is cleared because of a mode fault error.

28.5 Usage Notes

28.5.1 Setting Module Stop Function

Module stop control register B (MSTPCRB) is used to enable or disable operation of the RSPI. The RSPI is stopped after a reset. The registers become accessible on release from the module-stop state. For details, refer to section 9, Low-Power Consumption Function.

28.5.2 Note on Low-Power Consumption Functions

Set the SPCR.SPE bit to 0 and terminate communications prior to entering module-stop state.

28.5.3 Notes on Starting Transfer

If the corresponding interrupt request flag in the IRQ status register (IRQSn) is 1 at the time transfer is to be started, the next interrupt request is generated after the start of transfer and an interrupt request is retained in the module. This can lead to unanticipated behavior of the interrupt request flag.
When the request flag for the RSPI interrupt request is 1 at the time transfer is to start, follow the procedure below to clear interrupt requests in the module or the IRQ status register (IRQSn) before enabling operations (by setting the SPCR.SPE bit to 1).

1. Confirm that transfer has stopped (i.e. that the SPCR.SPE bit is 0).
2. Set the relevant interrupt enable bit (the SPCR.SPTIE or SPCR.SPRIE bit) to 0.
3. Read the relevant interrupt enable bit (the SPCR.SPTIE or SPCR.SPRIE bit) and confirm that its value is 0.
4. Set the relevant interrupt request flag in the IRQ status register (IRQSn) to 0.

29. SPI Multi I/O Bus Controller (SPIBSC)

The SPI multi I/O bus controller outputs control signals to the serial flash memory connected to the SPI multi I/O bus space, thus enabling direct connection of the serial flash memory.
This LSI incorporates one channel of SPI multi I/O bus controller.

29.1 Overview

This module allows using direct reading or SPI operating mode to transmit and receive data with the serial flash memory connected to the SPI multi I/O bus space. The specification of the SPIBSC is listed in Table 29.1.

Table 29.1 SPIBSC Specifications

Item	Description
Serial flash interface	• One serial flash memory can be connected. • A data bus size of 1 bit, 2 bits, or 4 bits can be selected.
External address space read mode	• A maximum of 4 Gbytes of address space is supported. • The SPBSSL pin can be automatically controlled by access address monitoring. • Efficient data reception is possible due to the internal read cache (line size: 64 bits × 16 entries).
SPI operating mode	• Any read and write operations are available for a serial flash memory.
Bit rate	• The internal baud rate generator divides the frequency of PCLKA to generate SPBCLK. • The frequency division ratio of SPBCLK can be set in the range from 2 to 4080.
SPBSSL pin control	• The delay from the time the SPBSSL signal is activated to the time SPBCLK starts operating (clock delay) can be set. Setting range: 1 to 8 SPBCLK; Unit of the setting: 1 SPBCLK • The delay from the time SPBCLK stops to the time the SPBSSL output is inactivated (SPBSSL negation delay) can be set. Setting range: 1.5 to 8.5 SPBCLK; Unit of the setting: 1 SPBCLK • The wait of the SPBSSL output for the next access (next access delay) can be set. Setting range: 1 to 8 SPBCLK; Unit of the setting: 1 SPBCLK • The polarity of the SPBSSL signal can be changed.

Figure 29.1 is a block diagram of this module.

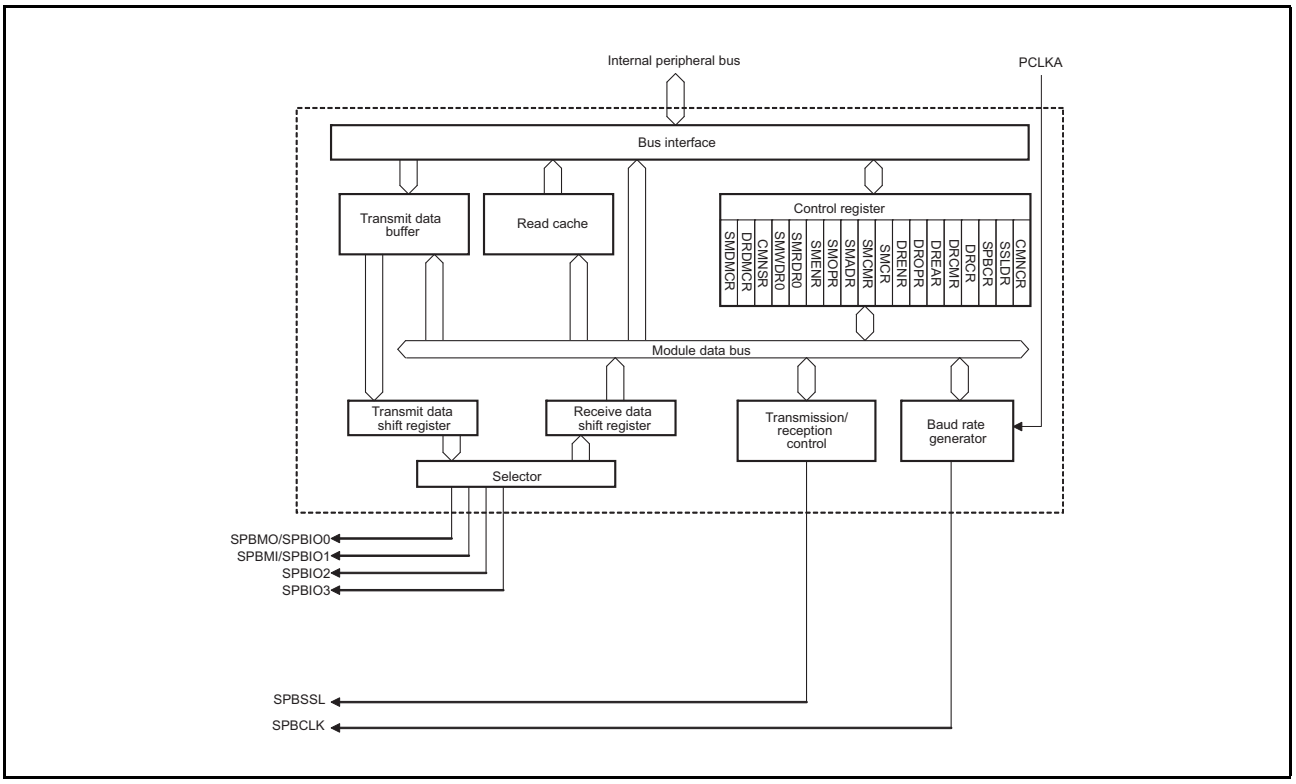


Figure 29.1 Block Diagram of SPIBSC

Table 29.2 lists the input/output pins of this module.

Table 29.2 Pin Configuration of the SPIBSC

Pin Name	Symbol	I/O	Function
Clock pin	SPBCLK	Output	Clock output
Slave select pin	SPBSSL	Output	Slave selection
Port data 0 pin	SPBMO/SPBIO0	I/O	Port master transmit data/data 0
Port data 1 pin	SPBMI/SPBIO1	I/O	Port master input data/data 1
Port data 2 pin	SPBIO2	I/O	Port data 2
Port data 3 pin	SPBIO3	I/O	Port data 3

29.2 Register Descriptions

29.2.1 Common Control Register (CMNCR)

The CMNCR register is a 32-bit register that controls the SPI multi I/O bus controller. The settings of this register are reflected both in external address space read mode and SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5000h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	MD	—	—	—	—	—	—	SFDE	MOIIO3[1:0]		MOIIO2[1:0]		MOIIO1[1:0]		MOIIO0[1:0]	
Value after reset:	0	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	IO3FV[1:0]		IO2FV[1:0]		—	—	IO0FV[1:0]		—	CPHAT	CPHAR	SSLP	CPOL	—	BSZ[1:0]	
Value after reset:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	BSZ[1:0]	Data Bus Size	Specifies the number of serial flash memories to be connected. This bit must be set to 00b because only one memory can connect to this product. If another value is set, the operation cannot be guaranteed. b1 b0 0 0: 1 memory 0 1: Setting prohibited 1 X: Setting prohibited	R/W
b2	—	Reserved	This bit is always read as 0. The write value should always be 0.	R/W
b3	CPOL	SPBSSL Negation Period SPBCLK Output Direction	Sets the output level of the SPBCLK pin during inactive period of the SPBSSL signal. 0: The output of the SPBCLK pin is 0 during inactive period of the SPBSSL signal. 1: The output of the SPBCLK pin is 1 during inactive period of the SPBSSL signal.	R/W
b4	SSLP	SPBSSL Signal Polarity	Sets the polarity of SPBSSL signal. 0: Active low SPBSSL signal 1: Active high SPBSSL signal	R/W
b5	CPHAR	Input Latch	Sets the edge of the SPBCLK signal for the reception data. The CPHAT bit and this bit should be set according to the following table. 0: Data reception at odd edge 1: Data reception at even edge Settings of the CPHAT and CPHAR Bits CPHAT / CPHAR: 0/0 Setting enabled; 0/1 Setting enabled; 1/0 Setting prohibited; 1/1 Setting enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b6	CPHAT	Output Shift	Sets the edge of the SPBCLK signal for data transmission. This bit and the CPHAR bit should be set according to the description of the CPHAR bit. 0: Data transmission at even edge 1: Data transmission at odd edge	R/W
b7	—	Reserved	This bit is always read as 0. The write value should always be 0.	R/W
b9, b8	IO0FV[1:0]	SPBIO0 Fixed Value for 1-bit Size Input	Fixes the output value of the SPBIO0 pins for 1-bit size input. b9 b8 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b11, b10	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b13, b12	IO2FV[1:0]	SPBIO2 Fixed Value for 1-bit/2-bit Size	Fixes the output value of the SPBIO2 pins for 1-bit/2-bit size. b13 b12 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b15, b14	IO3FV[1:0]	SPBIO3 Fixed Value for 1-bit/2-bit Size	Fixes the output value of the SPBIO3 pins for 1-bit/2-bit size. b15 b14 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b17, b16	MOIIO0[1:0]	SPBIO0 Fixed Value for SPBSSL Idle	Fixes output values of the SPBIO0 pins during inactive period of the SPBSSL signal. b17 b16 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b19, b18	MOIIO1[1:0]	SPBIO1 Fixed Value for SPBSSL Idle	Fixes output values of the SPBIO1 pins during inactive period of the SPBSSL signal. b19 b18 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b21, b20	MOIIO2[1:0]	SPBIO2 Fixed Value for SPBSSL Idle	Fixes output values of the SPBIO2 pins during inactive period of the SPBSSL signal. b21 b20 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W

Bit	Symbol	Bit Name	Description	R/W
b23, b22	MOIIO3[1:0]	SPBIO3 Fixed Value for SPBSSL Idle	Fixes output values of the SPBIO3 pins during inactive period of the SPBSSL signal. b23 b22 0 0: Output value 0 0 1: Output value 1 1 0: Output value is the last bit value of the previous transfer (or the state is Hi-Z, if Hi-Z was the state in the last bit period of the previous transfer). 1 1: Output value Hi-Z	R/W
b24	SFDE	Data Swap Setting for Serial Flash Memory	Specifies whether or not swapping of data in serial flash memory is performed. 0: Swapping is not performed. 1: Swapping is performed in 8-bit units. For details, see section 29.3.4, Data Alignment.	R/W
b30 to b25	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b31	MD	Operating Mode Switch	Switches the operating modes. 0: External address space read mode 1: SPI operating mode	R/W

29.2.2 SSL Delay Register (SSLDR)

The SSLDR register is a 32-bit register that adjusts the timing between the SPBSSL signal and the SPBCLK signal. The settings of this register are reflected both in external address space read mode and SPI operating mode.
The settings of this register should be changed when the TEND flag in the CMNSR register is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SPNDL[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	SLNDL[2:0]			—	—	—	—	—	SCKDL[2:0]		
Value after reset:	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	SCKDL[2:0]	Clock Delay	Sets the period from the time the SPBSSL signal is activated to the time the clock is output from the SPBCLK signal (clock delay). b2 b0 000: 1 SPBCLK cycle 001: 2 SPBCLK cycles 010: 3 SPBCLK cycles 011: 4 SPBCLK cycles 100: 5 SPBCLK cycles 101: 6 SPBCLK cycles 110: 7 SPBCLK cycles 111: 8 SPBCLK cycles	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b10 to b8	SLNDL[2:0]	SPBSSL Negation Delay	Sets the period from the time the last edge of the SPBCLK signal is transferred to the time the SPBSSL signal is inactivated (SPBSSL negation delay). b10 b8 000: 1.5 SPBCLK cycles 001: 2.5 SPBCLK cycles 010: 3.5 SPBCLK cycles 011: 4.5 SPBCLK cycles 100: 5.5 SPBCLK cycles 101: 6.5 SPBCLK cycles 110: 7.5 SPBCLK cycles 111: 8.5 SPBCLK cycles	R/W
b15 to b11	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b18 to b16	SPNDL[2:0]	Next Access Delay	Sets the period from transfer end to next transfer start (next access). b18 b16 000: 1 SPBCLK cycle 001: 2 SPBCLK cycles 010: 3 SPBCLK cycles 011: 4 SPBCLK cycles 100: 5 SPBCLK cycles 101: 6 SPBCLK cycles 110: 7 SPBCLK cycles 111: 8 SPBCLK cycles	R/W
b31 to b19	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

29.2.3 Bit Rate Register (SPBCR)

The SPBCR register is a 32-bit register that sets the bit rate.
The settings of this register are reflected both in external address space read mode and SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5008h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	SPBR[7:0]								—	—	—	—	—	—	BRDV[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

Bit	Symbol	Bit Name	Description	R/W
b1, b0	BRDV[1:0]	Bit Rate Frequency Division	Sets the bit rate. The bit rate is determined by a combination of these bits with the SPBR[7:0] bits. The setting value of the SPBR bit determines the base bit rate. This bit is used to select a division ratio of the base bit rate from among no division, 2, 4, and 8. b1 b0 0 0: Base bit rate 0 1: Base bit rate divided by 2 1 0: Base bit rate divided by 4 1 1: Base bit rate divided by 8	R/W
b7 to b2	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b15 to b8	SPBR[7:0]	Bit Rate	Sets the bit rate. The bit rate is determined by a combination of these bits with the BRDV[1:0] bits. For details, see Table 29.3, Relationship between SPBR[7:0] and BRDV[1:0] Settings.	R/W
b31 to b16	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

(1) Bit Rate

The SPBR[7:0] and BRDV[1:0] bits are used for setting the bit rate.
The following formula is used to calculate the bit rate when SPBR[7:0] ≠ 0:

Bit rate = PCLKA / ($2 \times n \times 2^N$)

n: SPBR[7:0] setting (1, …, 255)

N: BRDV[1:0] setting (0 to 3)

The following formula is used to calculate the bit rate when SPBR[7:0] = 0:

Bit rate = PCLKA / 2^N

Setting both the SPBR[7:0] and BRDV[1:0] bits to 0 is prohibited.

Table 29.3 Relationship between SPBR[7:0] and BRDV[1:0] Settings

			Bit Rate
SPBR[7:0] (n)	**BRDV[1:0] (N)**	**Division Ratio**	**PCLKA = 150 MHz**
0	0	1	Setting prohibited
0	1	2	75 Mbps
0	2	4	37.5 Mbps
0	3	8	18.75 Mbps
1	0	2	75 Mbps
2	0	4	37.5 Mbps
3	0	6	25 Mbps
4	0	8	18.75 Mbps
5	0	10	15 Mbps
6	0	12	12.5 Mbps
6	1	24	6.25 Mbps
6	2	48	3.13 Mbps
6	3	96	1.56 Mbps
255	3	4080	36.76 Kbps

Note: The bit rate should be set so that it will satisfy the AC characteristics of this module.

29.2.4 Data Read Control Register (DRCR)

The DRCR register is a 32-bit register that sets the operation in external address space read mode.
The bits except the SSLN bit should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 500Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	SSLN	—	—	—	—	RBURST[3:0]			
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	RCF	RBE	—	—	—	—	—	—	—	SSLE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SSLE	SPBSSL Negation	Sets the inactivation conditions for the SPBSSL signal during read burst. The SPBSSL signal is inactivated for each access during normal read. 0: The SPBSSL signal is inactivated after transfer of data set in burst length. 1: The SPBSSL signal is inactivated when the accessed address is not continuous with the previously transferred address.	R/W
b7 to b1	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b8	RBE	Read Burst	Turns burst read ON or OFF. 0: Data is read according to the access size. 1: Read cache is enabled, and as many data units as the burst count specified in RBURST[3:0] bits is read.	R/W
b9	RCF	Read Cache Flush	When 1 is written to this bit, all the entries in the read cache are cleared. This bit is always read as 0. Note: After flushing the read cache by writing 1 to this bit, read the DRCR register before proceeding to read from the external address space.	R/W
b15 to b10	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b19 to b16	RBURST [3:0]	Read Data Burst Length	Sets the burst length (data unit count) when reading. This bit is enabled when the RBE bit of this register is set to 1. b19 b16 0000: 1 data unit 0001: 2 continuous data units : 1110: 15 continuous data units 1111: 16 continuous data units One data unit is 64 bits long.	R/W
b23 to b20	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b24	SSLN	SPBSSL Negation	Writing 1 to this bit when both the RBE and SSLE bits of this register are 1 inactivates the active SPBSSL signal. This bit is always read as 0. Note: To start next access after the SPBSSL signal is inactivated by this bit, read the SSLF bit of the CMNSR register = 0 to confirm that the SPBSSL signal has been inactivated.	R/W

Bit	Symbol	Bit Name	Description	R/W
b31 to b25	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

29.2.5 Data Read Command Setting Register (DRCMR)

The DRCMR register is a 32-bit register that sets the commands issued in external address space read mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5010h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	CMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OCMD[7:0]	Optional Command	Sets the optional command.	R/W
b15 to b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b23 to b16	CMD[7:0]	Command	Sets the command. For details, see the details about the serial flash memory to be used.	R/W
b31 to b24	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

29.2.6 Data Read Extended Address Setting Register (DREAR)

The DREAR register is an address setting register when the serial flash address is output in 32-bit mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5014h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	EAV[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	EAC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	EAC[2:0]	32-Bit Extended External Address Valid Range	Sets the range of the external address to be used as serial flash address when the serial flash address is output in 32-bit mode. This setting is valid when the ADE[3] bit in DRENR is 1. b2 b0 000: External address bits [24:0] enabled 001: External address bits [25:0] enabled Other than above: Setting prohibited	R/W
b15 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b23 to b16	EAV[7:0]	32-Bit Extended Upper Address Fixed Value	The upper address of the external address specified by the EAC[2:0] bits of this register are set to these bits when the serial flash address is output in 32-bit mode. Bit 0 corresponds to the serial flash address bit [25], and bit 7 corresponds to the bit [32]. This setting is valid only when the ADE[3] bit in DRENR is 1. When EAC[2:0] are 000, serial flash address [32:25] are set to EAV[7:0]. When EAC[2:0] are 001, serial flash address [32:26] are set to EAV[7:1].	R/W
b31 to b24	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

29.2.7 Data Read Option Setting Register (DROPR)

The DROPR register is a 32-bit register that sets the option data in external address space read mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5018h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	OPD3[7:0]								OPD2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OPD1[7:0]								OPD0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OPD0[7:0]	Option Data 0	Sets the option data 0.	R/W
b15 to b8	OPD1[7:0]	Option Data 1	Sets the option data 1.	R/W
b23 to b16	OPD2[7:0]	Option Data 2	Sets the option data 2.	R/W
b31 to b24	OPD3[7:0]	Option Data 3	Sets the option data 3.	R/W

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

29.2.8 Data Read Enable Setting Register (DRENR)

The DRENR register is a 32-bit register that sets the bit size of the command, optional command, address, option data, and read data in external address space read mode and enables outputting them other than read data.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 501Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	DRDB[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DME	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				—	—	—	—
Value after reset:	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b7 to b4	OPDE[3:0]	Option Data Enable	Sets the option data to be output. Be sure to use the following setting; otherwise, the operation is not guaranteed. b7 b4 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited	R/W
b11 to b8	ADE[3:0]	Address Enable	Sets the address to be output. Be sure to use the following setting; otherwise, the operation is not guaranteed. b11 b8 0000: Output disabled 0111: Address[23:0] 1111: Address[31:0] Other than above: Setting prohibited	R/W
b12	OCDE	Optional Command Enable	Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled	R/W
b13	—	Reserved	This bit is always read as 0. The write value should always be 0.	R/W
b14	CDE	Command Enable	Sets the command to be output. 0: Command output disabled 1: Command output enabled	R/W
b15	DME	Dummy Cycle Enable	Enables insertion of the dummy cycle before the read data. Note: A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled	R/W
b17, b16	DRDB[1:0]	Data Read Bit Size	Sets the data read size in bit units. b17 b16 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W

Bit	Symbol	Bit Name	Description	R/W
b19, b18	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b21, b20	OPDB[1:0]	Option Data Bit Size	Sets the option data size in bit units. b21 b20 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b23, b22	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b25, b24	ADB[1:0]	Address Bit Size	Sets the address size in bit units. b25 b24 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b27, b26	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b29, b28	OCDB[1:0]	Optional Command Bit Size	Sets the optional command size in bit units. b29 b28 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b31, b30	CDB[1:0]	Command Bit Size	Sets the command size in bit units. For details, see the details about the serial flash memory to be used. b31 b30 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W

29.2.9 SPI Mode Control Register (SMCR)

The SMCR register is a 32-bit register that sets the operation in SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5020h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	SSLKP	—	—	—	—	—	SPIRE	SPIWE	SPIE
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	SPIE	SPI Data Transfer Enable	Data is transferred by setting this bit to 1. This bit is enabled only when the TEND bit in CMNSR is set to 1. The operation cannot be guaranteed when this bit is set to 1 with the TEND bit set to 0. This bit is always read as 0. Note: When the SPBSSL signal is inactive, the command, optional command, address, and option data that are set as the output by the DRENR register are output even if the SPIRE and SPIWE bits are set to 0. When the SPBSSL signal is active, follow the notes described in section 29.4.2, Notes on Starting Transfer from the SPBSSL Signal Retained State in SPI Operating Mode.	R/W
b1	SPIWE	Data Write Enable	Sets write operation in SPI operating mode. 0: Data writing disabled 1: Data writing enabled Note: When the transfer data bit size is set to 2 bits or 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.	R/W
b2	SPIRE	Data Read Enable	Sets read operation in SPI operating mode. 0: Data reading disabled 1: Data reading enabled Note: When the transfer data bit size is set to 2 bits or 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.	R/W
b7 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b8	SSLKP	SPBSSL Signal Level	Determines the status of the SPBSSL signal after the end of transfer. 0: SPBSSL signal is inactivated at the end of transfer. 1: SPBSSL signal level is maintained from the end of transfer to the start of next access.	R/W
b31 to b9	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

29.2.10 SPI Mode Command Setting Register (SMCMR)

The SMCMR register is a 32-bit register that sets the commands issued in SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5024h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	CMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	OCMD[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OCMD[7:0]	Optional Command	Sets the optional command.	R/W
b15 to b8	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b23 to b16	CMD[7:0]	Command	Sets the command. For details, see the details about the serial flash memory to be used.	R/W
b31 to b24	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

29.2.11 SPI Mode Address Setting Register (SMADR)

The SMADR register is a 32-bit register that sets the addresses of the serial flash memory in SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5028h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ADRE[7:0]								ADR[23:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ADR[23:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b23 to b0	ADR[23:0]	Address	Sets the address of the serial flash memory.	R/W
b31 to b24	ADRE[7:0]	Address	Sets the value of bits 31 to 24 when the serial flash address is output in 32-bit units. This setting is valid when ADE[3] in SMENR is 1.	R/W

29.2.12 SPI Mode Option Setting Register (SMOPR)

The SMOPR register is a 32-bit register that sets the option data in SPI operating mode.
The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 502Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	OPD3[7:0]								OPD2[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	OPD1[7:0]								OPD0[7:0]							
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	OPD0[7:0]	Option Data 0	Sets the option data 0.	R/W
b15 to b8	OPD1[7:0]	Option Data 1	Sets the option data 1.	R/W
b23 to b16	OPD2[7:0]	Option Data 2	Sets the option data 2.	R/W
b31 to b24	OPD3[7:0]	Option Data 3	Sets the option data 3.	R/W

Note: OPD3, OPD2, OPD1, and OPD0 are output in this order.

29.2.13 SPI Mode Enable Setting Register (SMENR)

The SMENR register is a 32-bit register that sets the bit size of the command, optional command, address, option data, and transfer data in SPI operating mode and enables various outputs. Disabling output of all of the command, optional command, address, option data, dummy cycle, and transfer data is prohibited. At least one of them except dummy cycle must be enabled.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5030h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CDB[1:0]		OCDB[1:0]		—	—	ADB[1:0]		—	—	OPDB[1:0]		—	—	SPIDB[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DME	CDE	—	OCDE	ADE[3:0]				OPDE[3:0]				SPIDE[3:0]			
Value after reset:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

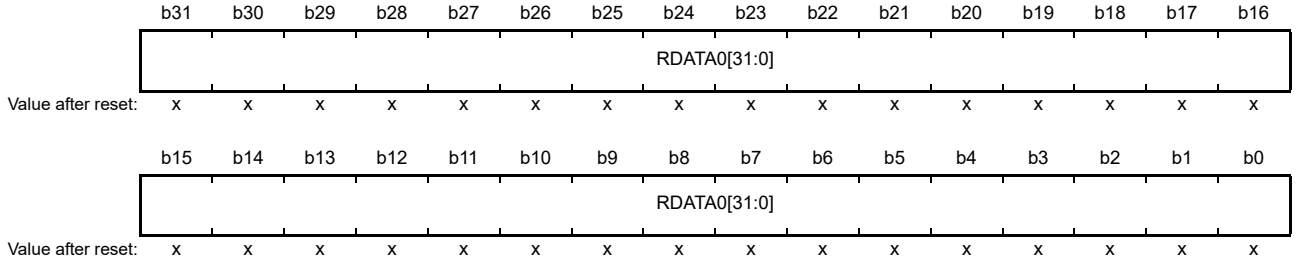
Bit	Symbol	Bit Name	Description	R/W
b3 to b0	SPIDE[3:0]	Transfer Data Enable	Sets transfer data. The settings below must be used. Otherwise, the operation is not guaranteed. b3 b0 0000: Not transferred 1000: 8 bits transferred (enables data at address 0 of the SPI mode read/write data registers 0) 1100: 16 bits transferred (enables data at addresses 0 and 1 of the SPI mode read/write data registers 0) 1111: 32 bits transferred (enables data at addresses 0 to 3 of the SPI mode read/write data registers 0) Other than above: Setting prohibited	R/W
b7 to b4	OPDE[3:0]	Option Data Enable	Sets the option data to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. b7 b4 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited	R/W
b11 to b8	ADE[3:0]	Address Enable	Sets the address to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. b11 b8 0000: Output disabled 0100: ADR[23:16] 0110: ADR[23:8] 0111: ADR[23:0] 1111: ADR[31:0] Other than above: Setting prohibited	R/W
b12	OCDE	Optional Command Enable	Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled	R/W
b13	—	Reserved	This bit is always read as 0. The write value should always be 0.	R/W
b14	CDE	Command Enable	Sets the command to be output. 0: Command output disabled 1: Command output enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b15	DME	Dummy Cycle Enable	Enables insertion of the dummy cycle before the read data. Note 1. Dummy cycle insertion is prohibited for write in SPI operating mode including the case in which a transfer ends with a dummy cycle. Note 2. A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled	R/W
b17, b16	SPIDB[1:0]	Transfer Data Bit Size	Sets the transfer data size in bit units. b17 b16 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b19, b18	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b21, b20	OPDB[1:0]	Option Data Bit Size	Sets the option data size in bit units. b21 b20 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b23, b22	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b25, b24	ADB[1:0]	Address Bit Size	Sets the address size in bit units. b25 b24 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b27, b26	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b29, b28	OCDB[1:0]	Optional Command Bit Size	Sets the optional command size in bit units. b29 b28 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b31, b30	CDB[1:0]	Command Bit Size	For details, see the details about the serial flash memory to be used. b31 b30 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W

29.2.14 SPI Mode Read Data Register 0 (SMRDR0)

The SMRDR0 register is a 32-bit register that stores the read data in SPI operating mode.
The setting of this register should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.
The alignment of data depends on the access size. For details, see section 29.3.4, Data Alignment.
This register must be accessed in the same unit as the transfer size set in the SPIDE[3:0] bits in the SPI mode enable register (SMENR). It must also be accessed from its own address 0.

Address(es): A000 5038h



x: Undefined

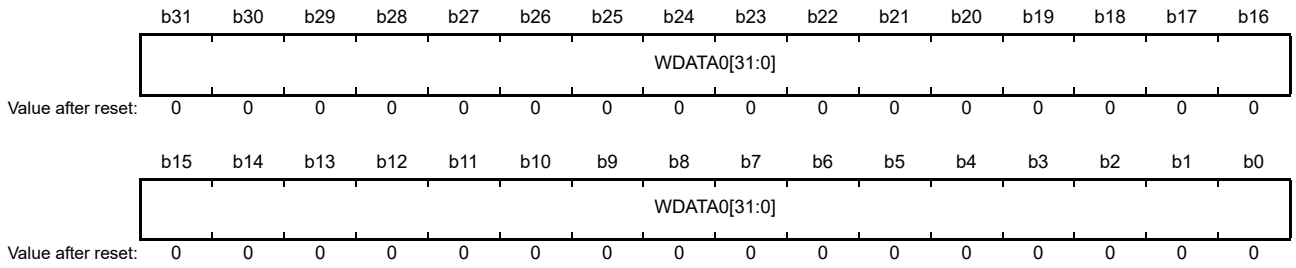
Bit	Symbol	Bit Name	Description	R/W
b31 to b0	RDATA0 [31:0]	Read Data	Holds the data read in SPI operating mode.	R

The contents of this register are modified upon completion of reception in SPI operating mode. Be sure to read data when reception in SPI operating mode is completed.

29.2.15 SPI Mode Write Data Register 0 (SMWDR0)

The SMWDR0 register is a 32-bit register that sets the write data in SPI operating mode.
The setting of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.
The alignment of data depends on the access size. For details, see section 29.3.4, Data Alignment.
This register must be accessed in the same unit as the transfer size set in the SPIDE[3:0] bits in the SPI mode enable register (SMENR). It must also be accessed from its own address 0.

Address(es): A000 5040h



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	WDATA0 [31:0]	Write Data	Holds the data to be written in SPI operating mode.	R/W

29.2.16 Common Status Register (CMNSR)

The CMNSR register is a 32-bit register that holds flags indicating the operating state.
The settings of this register are reflected both in external address space read mode and SPI operating mode.

Address(es): A000 5048h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SSLF	TEND
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	TEND	Transfer End Flag	Indicates whether the data transfer has ended. 0: Indicates that data transfer is in progress. 1: Indicates that data transfer has ended.	R
b1	SSLF	SPBSSL Pin Monitor	0: SPBSSL pin is inactive. 1: SPBSSL pin is active.	R
b31 to b2	—	Reserved	These bits are always read as 0.	R

29.2.17 Data Read Dummy Cycle Setting Register (DRDMCR)

The DRDMCR register is a 32-bit register that sets the size and number of dummy cycles to be inserted in external address space read mode.

The settings of this register are enabled when the DME bit in the data read enable setting register (DRENR) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5058h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMDB[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DMCYC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	DMCYC [2:0]	Number of Dummy Cycles	Sets the number of dummy cycles to be inserted when the DME bit in the data read enable setting register (DRENR) is 1. b2 b0 000: 1 cycle 001: 2 cycles 010: 3 cycles 011: 4 cycles 100: 5 cycles 101: 6 cycles 110: 7 cycles 111: 8 cycles	R/W
b15 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b17, b16	DMDB [1:0]	Dummy Cycle Bit Size	Sets the dummy cycle size in bit units. The setting of these bits is combined with the setting of the IO0FV, IO2FV, and IO3FV bits in the common control register (CMNCR) to determine the state of the unused pins during the dummy cycles. The state of the used pins is Hi-Z. b17 b16 0 0: 1 bit 0 1: 2 bits 1 0: 4 bits 1 1: Setting prohibited	R/W
b31 to b18	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

29.2.18 SPI Mode Dummy Cycle Setting Register (SMDMCR)

The SMDMCR register is a 32-bit register that sets the size and number of dummy cycles to be inserted in SPI operating mode.

The settings of this register are enabled when the DME bit in the SPI mode enable setting register (SMENR) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Address(es): A000 5060h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMDB[1:0]	
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DMCYC[2:0]		
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b2 to b0	DMCYC[2:0]	Number of Dummy Cycles	Sets the number of dummy cycles to be inserted when the DME bit in the SPI mode enable setting register (SMENR) is 1. b2 b0 000: 1 cycle 001: 2 cycles 010: 3 cycles 011: 4 cycles 100: 5 cycles 101: 6 cycles 110: 7 cycles 111: 8 cycles	R/W
b15 to b3	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W
b17, b16	DMDB [1:0]	Dummy Cycle Bit Size	Sets the dummy cycle size in bit units. The setting of these bits is combined with the setting of the IO0FV, IO2FV, and IO3FV bits in the common control register (CMNCR) to determine the state of the unused pins during the dummy cycles. The state of the used pins is Hi-Z. b17 b16 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited	R/W
b31 to b18	—	Reserved	These bits are always read as 0. The write value should always be 0.	R/W

29.3 Operation

29.3.1 System Configuration

With this module, one serial flash memory can be directly connected (data size of 1, 2, and 4 bits).
Figure 29.2 shows a system configuration example.

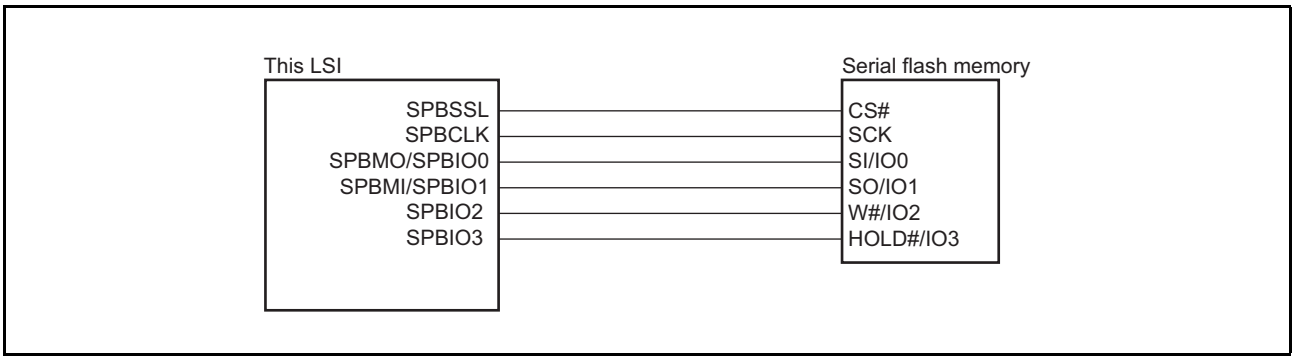


Figure 29.2 System Configuration Example with 4-Bit Data Size of a Serial Flash Memory Connected (BSZ[1:0] Bits in CMNCR = 00)

29.3.2 Address Map

In external address space read mode, the serial flash connected is assigned in the SPI multi I/O bus space. By the DREAR register setting, a maximum of 4 Gbytes can be accessed.

Table 29.4 Address Map

Number of Serial Flash Memories Connected	Internal Address	Max. Access Area
1	1000 0000h to 13FF FFFFh	4 Gbytes
	3000 0000h to 33FF FFFFh (mirror area)	

29.3.3 32-Bit Serial Flash Addresses

Since the SPI multi I/O bus space is 64 Mbytes, only a part of the 32-bit serial flash address area can be directly accessed. Here, the fixed value set in the DREAR register is used as the upper bit value of a 32-bit address.

To output serial flash addresses in 32 bits, set the ADE[3] bit in DRENR to 1, set the range of the external addresses used as the serial flash addresses to the EAC[2:0] bits in DREAR, and set the upper bit value of the 32-bit address as the fixed value to the EAV[7:0] bits in DREAR.

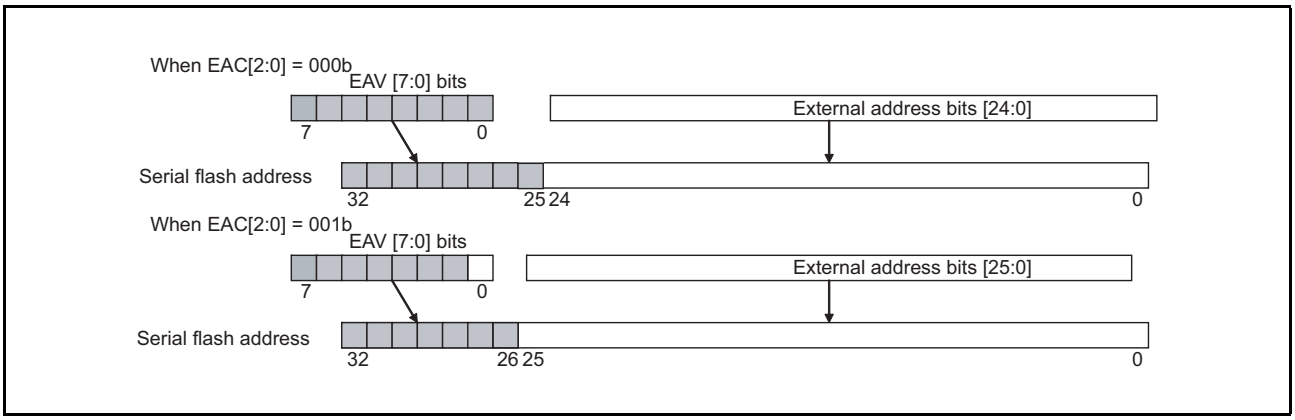


Figure 29.3 32-Bit Address Setting

Setting the ADE[3] bit in DRENR to 1 allows the serial flash address to be output using [31:0] bits.

When EAC[2:0] = 000b, external address bits [24:0] are valid; set the value for [32:25] bits to EAV[7:0].

When EAC[2:0] = 001b, external address bits [25:0] are valid; set the value for [32:26] bits to EAV[7:1].

When one serial flash memory is connected, address bits [31:0] are used.

29.3.4 Data Alignment

Data alignment can be set by using the SFDE bit in the common control register (CMNCR). Data alignment in data read mode and in SPI mode are shown in Figure 29.4 and Figure 29.5, respectively.

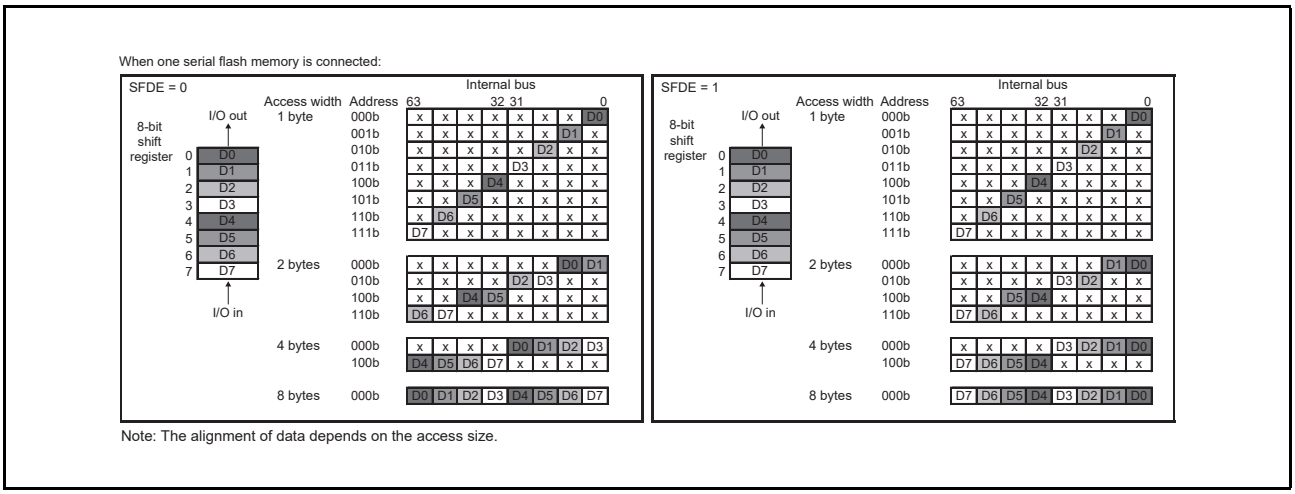


Figure 29.4 Data Alignment in External Address Space Read Mode

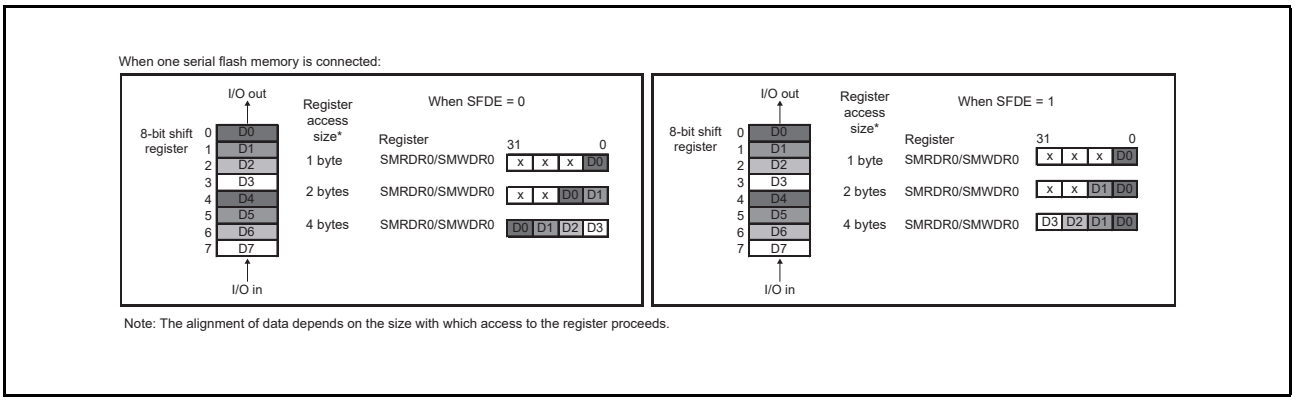


Figure 29.5 Data Alignment in SPI Operating Mode

29.3.5 Operating Modes

This module has two operating modes: external address space read mode and SPI operating mode.
In external address space read mode, a read access to the SPI multi I/O bus space is converted into SPI communication and data is received. After data acquisition, data is returned to the bus master that is the issuing source. For details, see section 29.3.6, External Address Space Read Mode.
In SPI operating mode, arbitrary SPI communication is carried out based on the register settings. For details, see section 29.3.8, SPI Operating Mode.

29.3.6 External Address Space Read Mode

A read access to the SPI multi I/O bus space can be converted into SPI communication in external address space read mode. Further, the commands, optional commands, option data, and dummy cycle issued for reading can be modified by the register settings.
In external address space read mode, either normal read operation or burst read operation can be selected. The transfer format is determined by the settings of the common control register (CMNCR), SSL delay register (SSLDR), bit rate setting register (SPBCR), data read control register (DRCR), data read command setting register (DRCMR), data read extended address setting register (DREAR), data read option setting register (DROPR), data read enable setting register (DRENR), and data read dummy cycle setting register (DRDMCR).

(1) Normal Read Operation

When the RBE bit in DRCR is set to 0, normal read operation is performed.
In the normal read operation, the data of 8 bits, 16 bits, and 32 bits are read for respectively a byte, a word, and a longword read access. After reading, the SPBSSL signal is inactivated.
The normal read operation timing is shown in Figure 29.6.
t1 is the time period from the time the SPBSSL signal is activated to the time the clock is output from the SPBCLK signal (clock delay). t2 is the time period from transmission of the last edge of the SPBCLK signal of a transfer to the time the SPBSSL signal is inactivated (SPBSSL negation delay). t3 is the time period from one transfer end to the next transfer start (next access). For details of t1, t2, and t3, see section 29.3.9, Transfer Format.

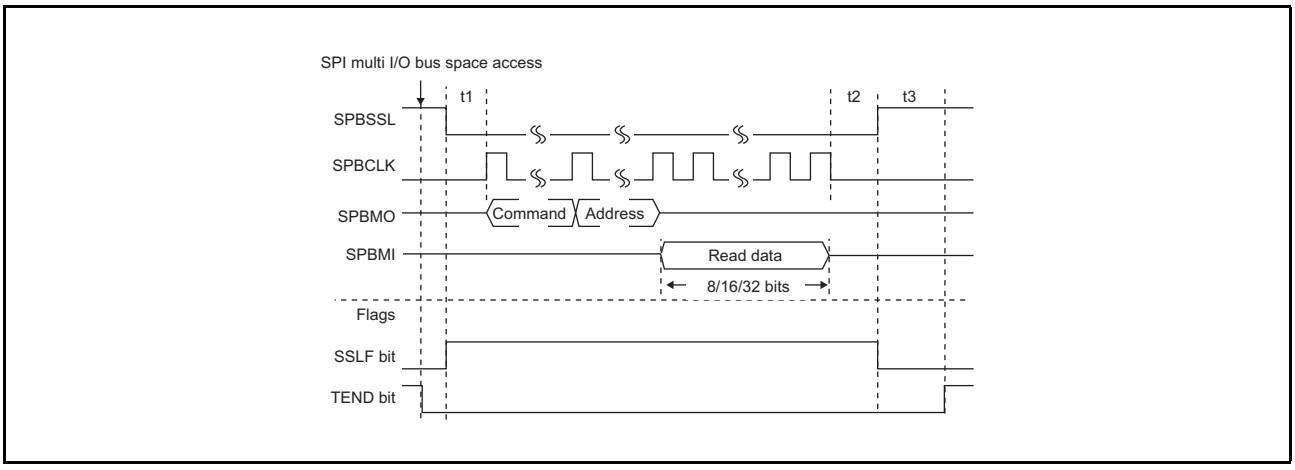


Figure 29.6 Normal Read Operation Timing

(2) Burst Read Operation

When the RBE bit in DRCR is set to 1, burst read operation is performed. Read cache is enabled in the burst read operation. For read cache operation, see section 29.3.7, Read Cache.

For reading bytes, words, or longwords, the read cache is first referred to for the data. When the read cache contains the data, the data is read from the read cache without accessing the serial flash memory. When the read cache does not contain the data, burst read operation is performed in the serial flash memory and the read data is stored in the read cache. The data transfer length at that time is 64 bits x RBURST[3:0] bits and the data is always read from the 64-bit boundary.

The status of the SPBSSL signal after data transfer can be selected by using the SSLE bit in DRCR. When the SSLE bit is set to 0, the SPBSSL signal is always inactivated after data transfer. For an operation performed when the SSLE bit is set to 1, see section 29.3.6, (3) Burst Read Operation with Automatic SPBSSL Inactivation, just below.

A pattern diagram of this operation and a burst read operation timing diagram when SSLE bit is set to 0 are shown in Figure 29.7 and Figure 29.8.

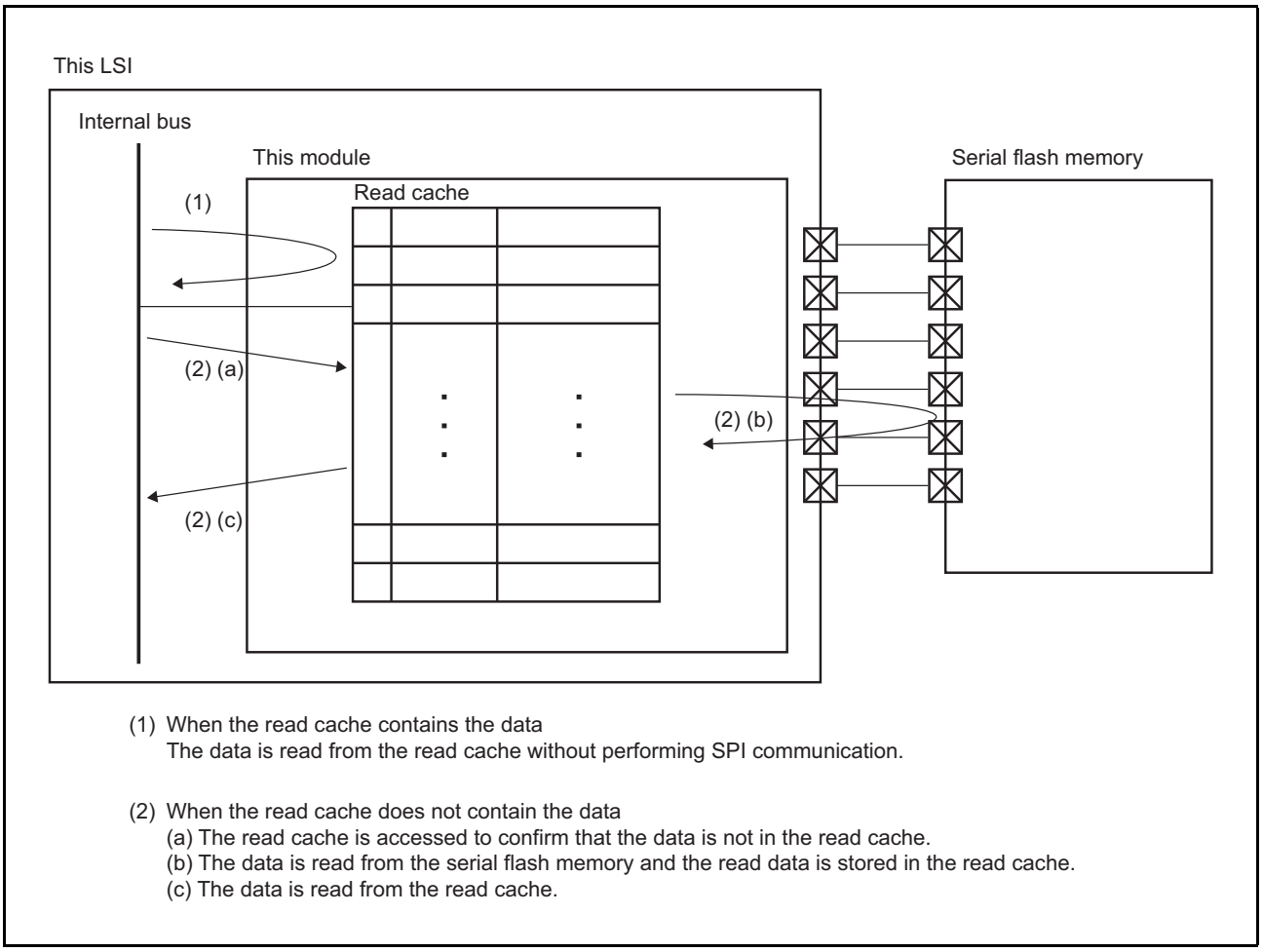


Figure 29.7 Burst Read Operation

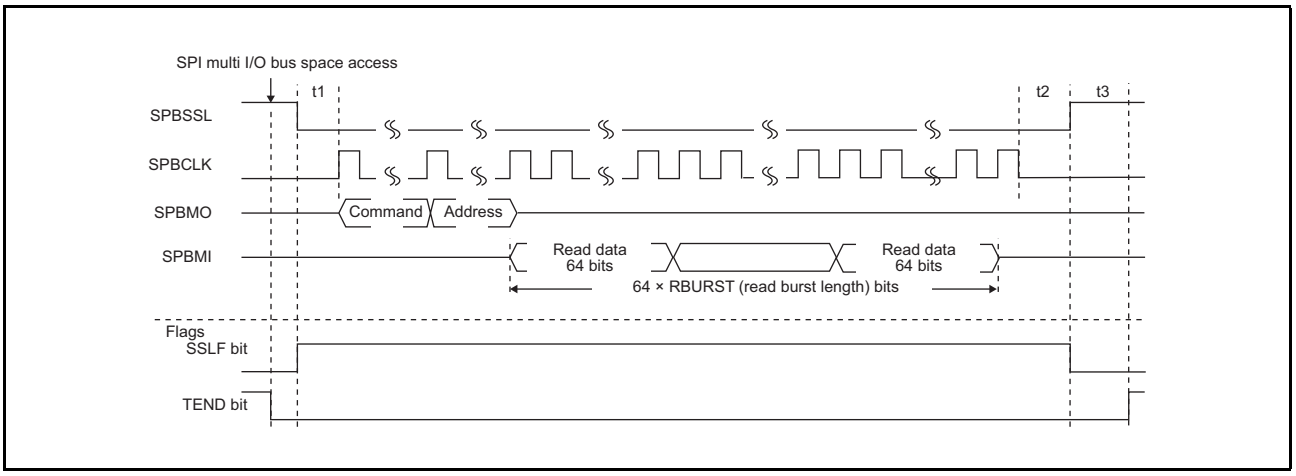


Figure 29.8 Burst Read Operation Timing (SSLE Bit = 0)

(3) Burst Read Operation with Automatic SPBSSL Inactivation

When SSLE bit in DRCR is set to 1, this module does not inactivate the SPBSSL signal after the burst read transfer. When accessing the next time, if the address is continuous with the previous read address, the burst read operation is performed without issuing the command, optional command, address, option data, or dummy cycle. If the address is not continuous with the previous read address, the SPBSSL signal is once inactivated and the burst read operation is performed after issuing the command, optional command, address, option data, or dummy cycle.
Burst read timing diagrams for continuous address and non-continuous address are shown in Figure 29.9 and Figure 29.10.

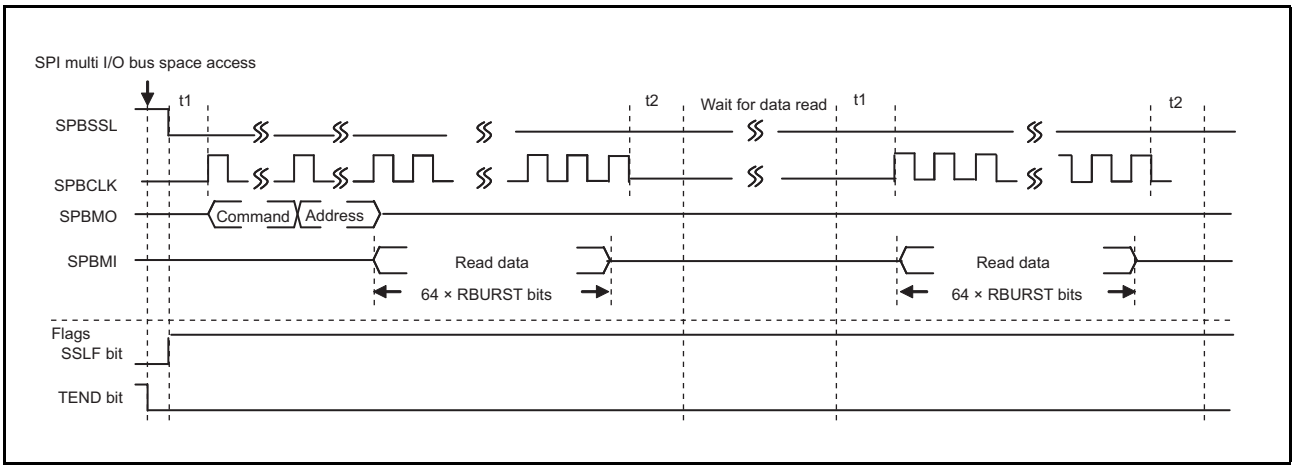


Figure 29.9 Burst Read Timing for Continuous Address (SSLE Bit = 1)

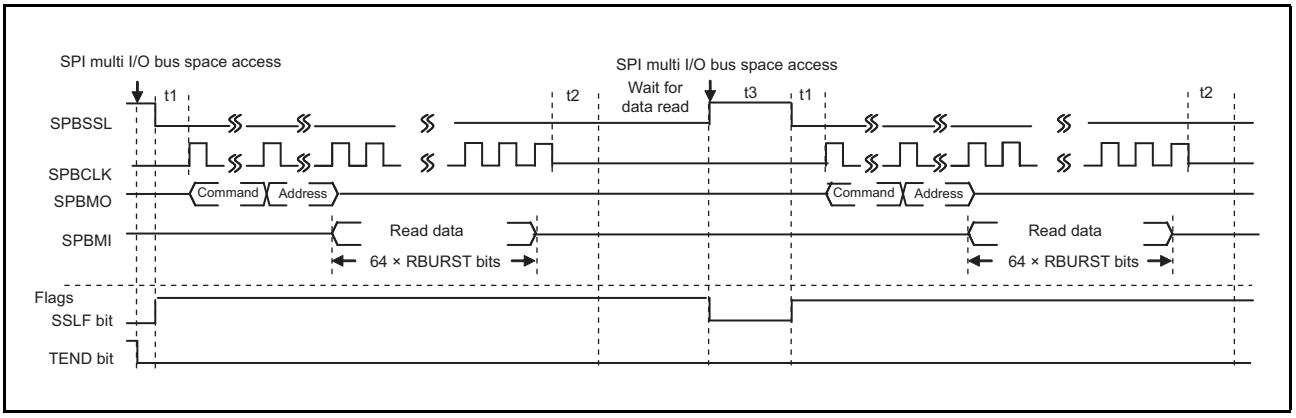


Figure 29.10 Burst Read Timing for Non-Continuous Address (SSLE Bit = 1)

For the next access after inactivation of the SPBSSL signal with the SSLN bit in DRCR with this operation, read SSLF = 0 in CMNSR to confirm that the SPBSSL signal has been inactivated.

(4) Initial Setting Flow

An example of an initial setting flow in external address space read mode is shown in Figure 29.11.

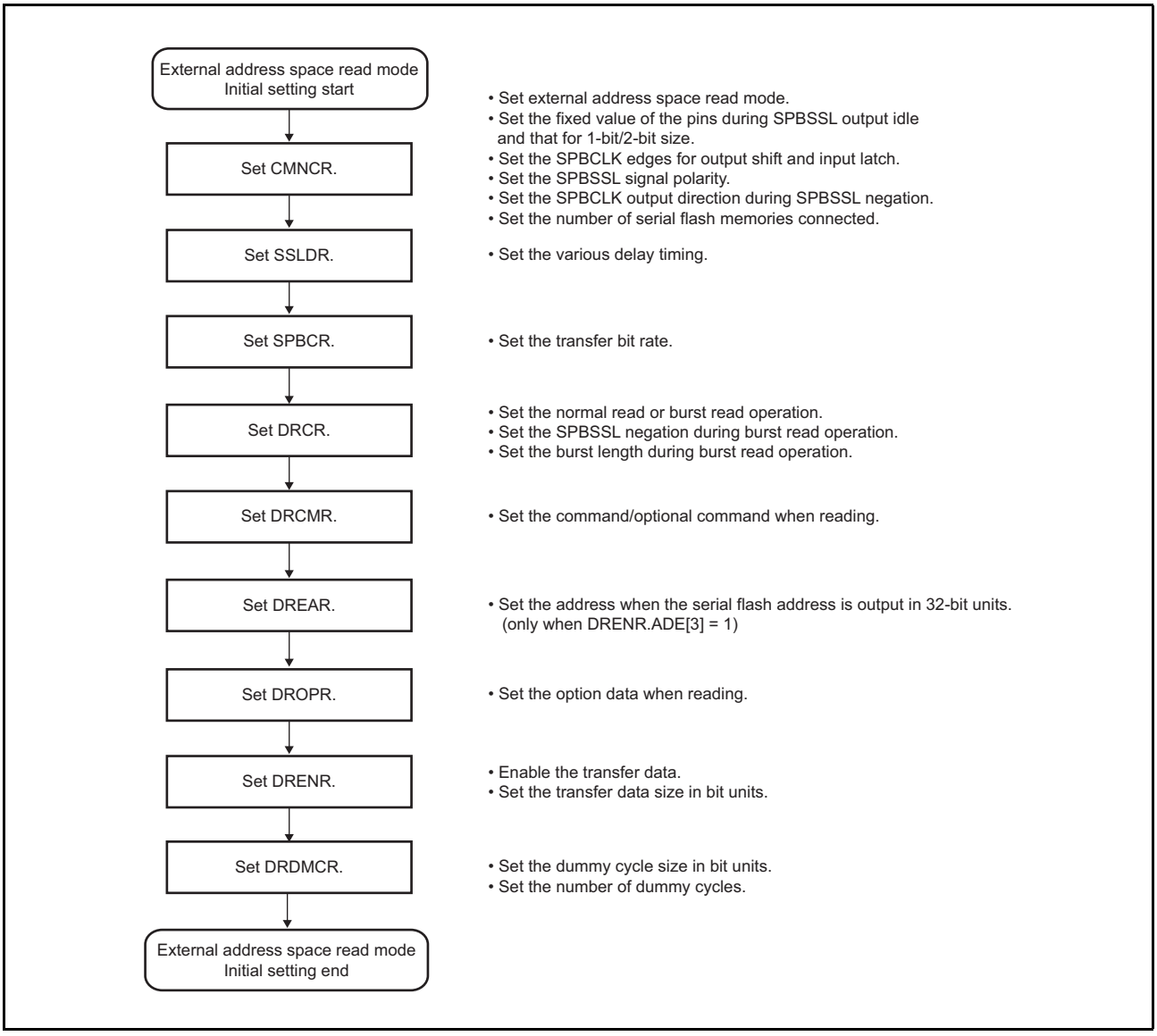


Figure 29.11 Example of Initial Setting Flow in External Address Space Read Mode

29.3.7 Read Cache

This module has a simple built-in read cache. The read cache can be used during external address space read mode and burst read operation. The read cache is configured with a line size of 64 bits and 16 entries.
Read cache configuration is shown in Figure 29.12.

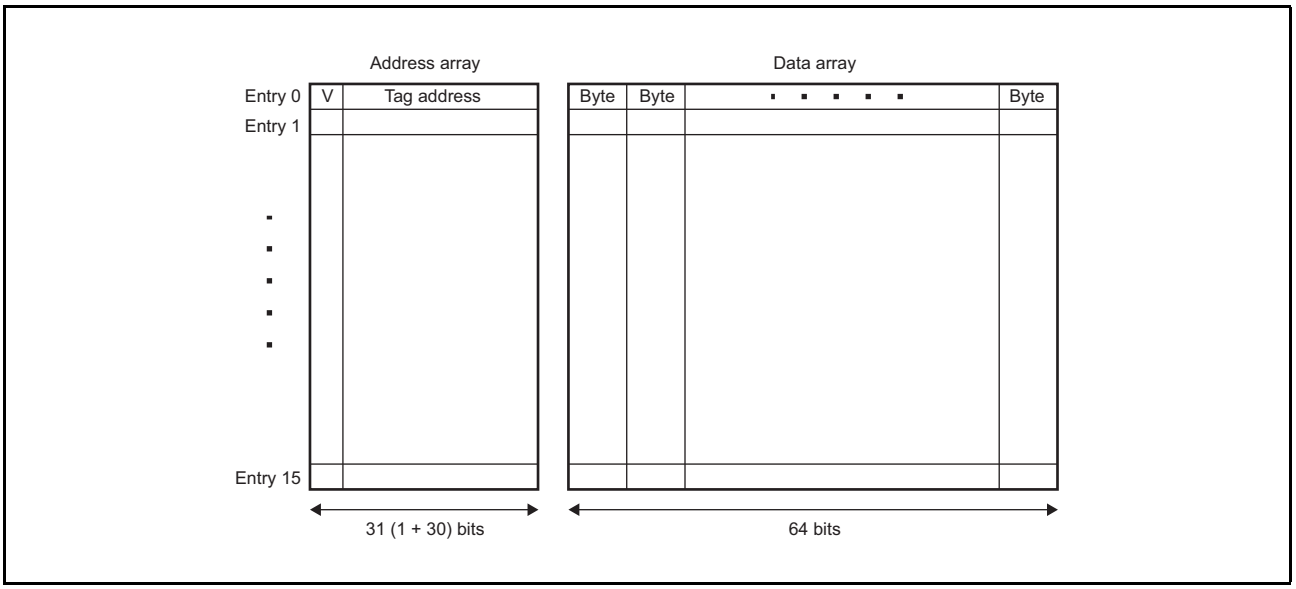


Figure 29.12 Read Cache Configuration

(1) Address Array

The V bit in Figure 29.12 indicates whether the entry data is valid. When the V bit is 1, the data is valid and when V bit is 0, the data is invalid.
The tag address bits hold the address used by the serial flash memory. An address is composed of bits 32 to 3. Address bits 23 to 3 are enabled when address output is 24 bits.
Address bits 31 to 3 are enabled when address output is 32 bits.

(2) Data Array

It retains the 64-bit read data. Registration in the read cache is performed in line units.

(3) Read Operation

If read data is hit in the cache, data is read from the read cache. In case of miss-hit, after the 64 × RBURST (read burst length) data is read from the serial flash memory and the read cache is updated, the data is returned to the bus master.

(4) Data Replacement

Data update is managed by the write pointer. In case of miss-hit of read data, the RBURST (read burst length) portion data is replaced starting at the entry specified by the write pointer. In other words, the data is replaced in the storage order of the data. Whether data is referred to or not will not affect the replacement order of data.

29.3.8 SPI Operating Mode

This module can carry out an arbitrary SPI operation by using the register settings.
The transfer format is determined by the settings of the common control register (CMNCR), SSL delay register (SSLDR), bit rate setting register (SPBCR), SPI mode control register (SMCR), SPI mode command setting register (SMCMR), SPI mode address setting register (SMADR), SPI mode option setting register (SMOPR), and SPI mode enable setting register (SMENR), SPI mode read data register (SMRDR), SPI mode write data register (SMWDR), and SPI mode dummy cycle setting register (SMDMCR).
SPI operating mode can be used for reading the status of the serial flash memory and writing to the serial flash memory. In this mode, one transfer refers to the operation from when the SPIE bit in SMCR is set to 1 to when the TEND bit is set to 1.

(1) Transfer Start

The transfer of data is started in the set transfer format by setting the SPIE bit in SMCR to 1. When write operation is enabled, the SPI mode write data register is transmitted to the serial flash memory. When read operation is enabled, data read from the serial flash memory is stored into the SPI mode read data register.
The SPI operation timing is shown in Figure 29.13.

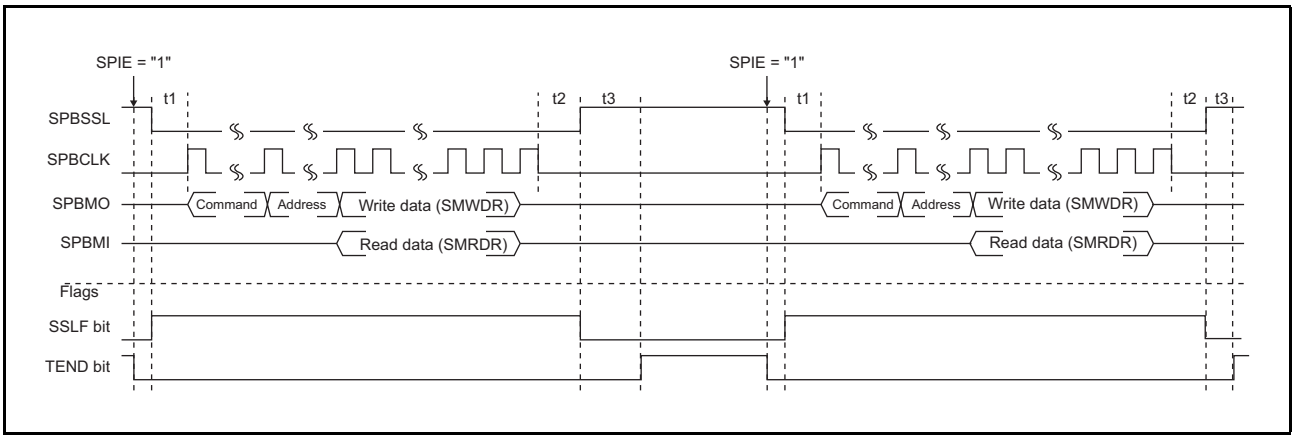


Figure 29.13 SPI Operation Timing

(2) Read/Write Enable

- Read operation: Data can be read by setting the SPIRE bit in SMCR to 1. The read data is stored into SMRDR.
- Write operation: Data can be written by setting the SPIWE bit in SMCR to 1. The data stored in SMWDR is output.

When the data size is set to 1 bit using the SPIDB[1:0] bits in SMENR, data can be transmitted and received by setting the SPIRE and SPIWE bits to 1. However, when the data size is set to 2 or 4 bits by using the SPIDB[1:0] bits, only one of the SPIRE and SPIWE bits should be enabled. The operation is not guaranteed if both the bits are enabled.

(3) Retention of SPBSSL Pin Activation

By setting the SSLKP bit in SMCR to 1, activation of the SPBSSL signal can be continued till the next transfer. With this function, the transfer can be carried out continuously with the SPBSSL signal kept in the active state.
The data transfer timing using the SSLKP bit is shown in Figure 29.14.

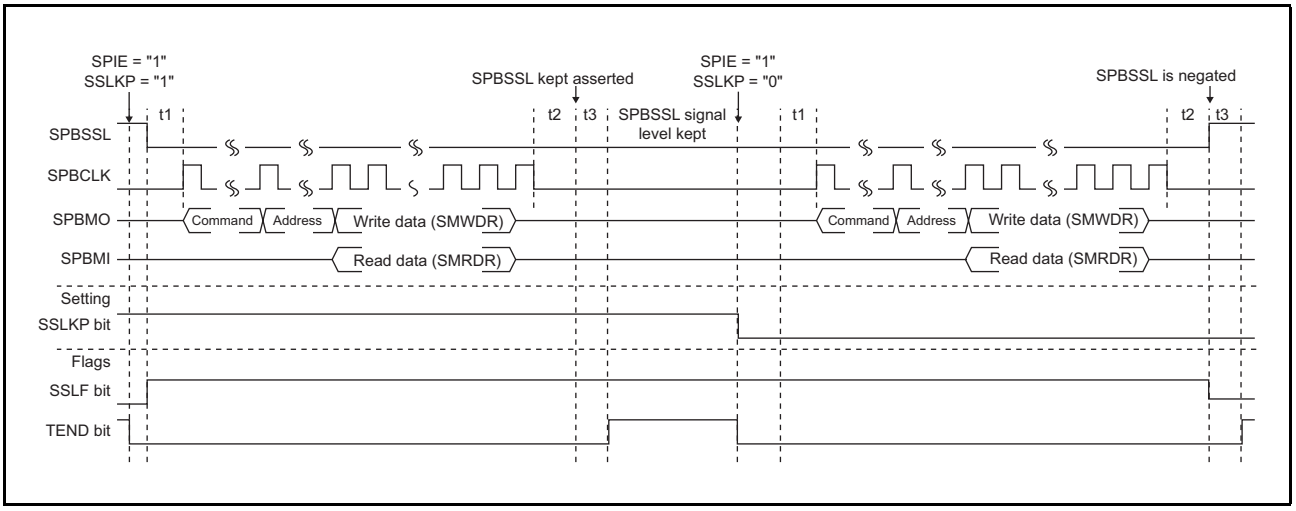


Figure 29.14 Data Transfer Timing using the SSLKP Bit

(4) Initial Setting Flow

An example of an initial setting flow in SPI operating mode is shown in Figure 29.15.

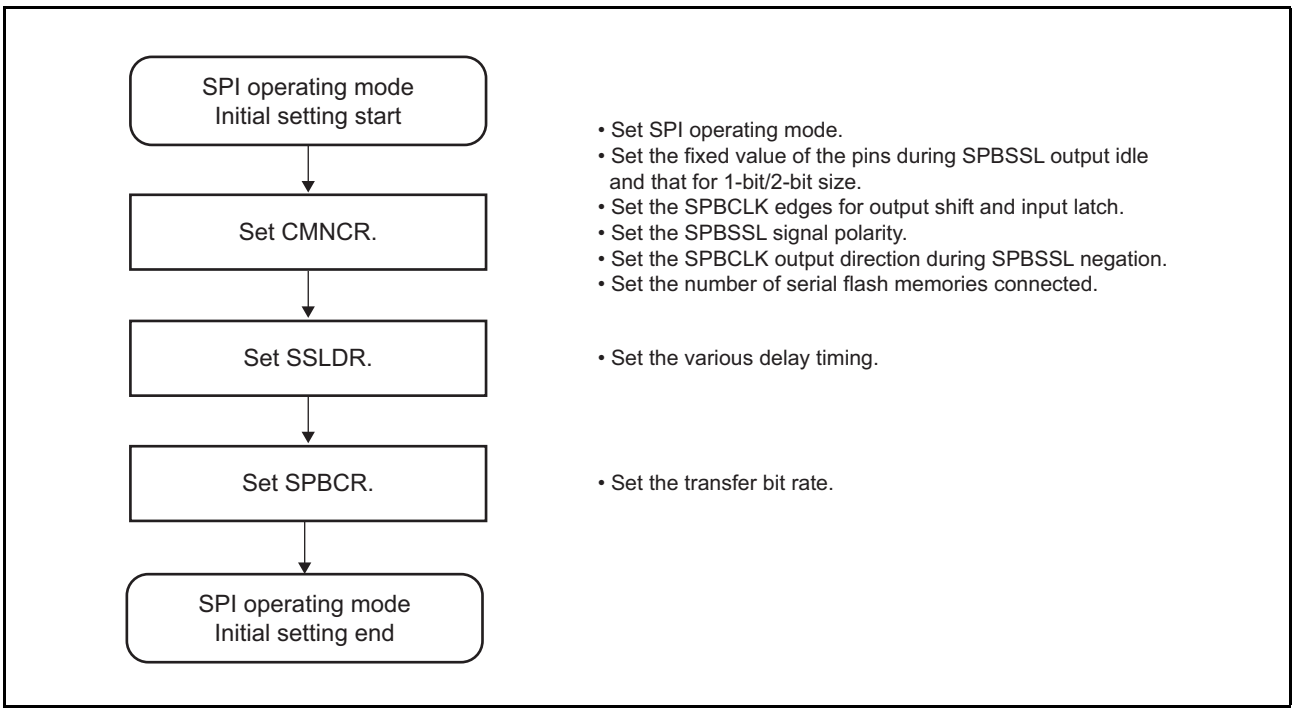


Figure 29.15 Example of Initial Setting Flow in SPI Operating Mode

(5) Data Transfer Setting Flow

An example of a data transfer setting flow in SPI operating mode is shown in Figure 29.16.

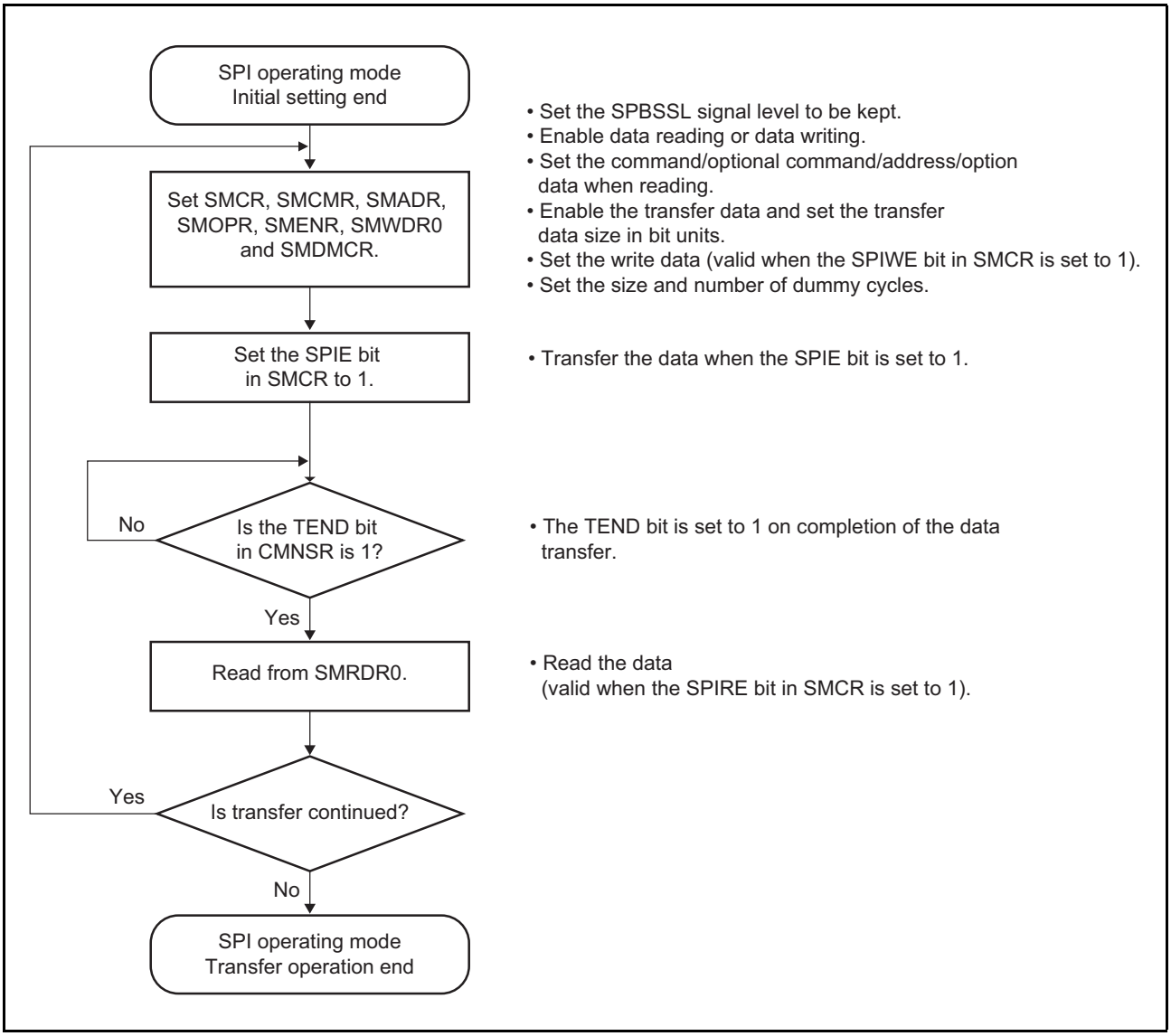


Figure 29.16 Example of a Data Transfer Setting Flow in SPI Operating Mode

29.3.9 Transfer Format

(1) SPBSSL Pin Enable Polarity Control

The enable polarity of the SPBSSL signal can be changed with the SSLP bit in CMNCR.

(2) SPBCLK Output

The output level of the SPBCLK signal while the SPBSSL signal is inactive can be set with the CPOL bit in CMNCR.

(3) Data Transmission and Reception Timing

Data is transmitted and received at either the odd or even edges. The data transmission timing can be set to the odd or even edge with the CPHAT bit in CMNCR. Similarly, the data reception timing can be set to the odd or even edge with the CPHAR bit in CMNCR.

(4) Delay Settings

t1 is the time period from the time the SPBSSL signal is activated to the clock output of the SPBCLK signal (clock delay). It can be set with the SCKDL[2:0] bits in SSLDR. t2 is the time period from the time the clock output of the SPBCLK signal is stopped to the time the SPBSSL signal is inactivated (SPBSSL negation delay). It can be set with the SLNDL[2:0] bits in SSLDR. t3 is the time period required to prevent SPBSSL signal activation for the next transfer after the end of the previous transfer (next access delay). It can be set with the SPNDL[2:0] bits in SSLDR.

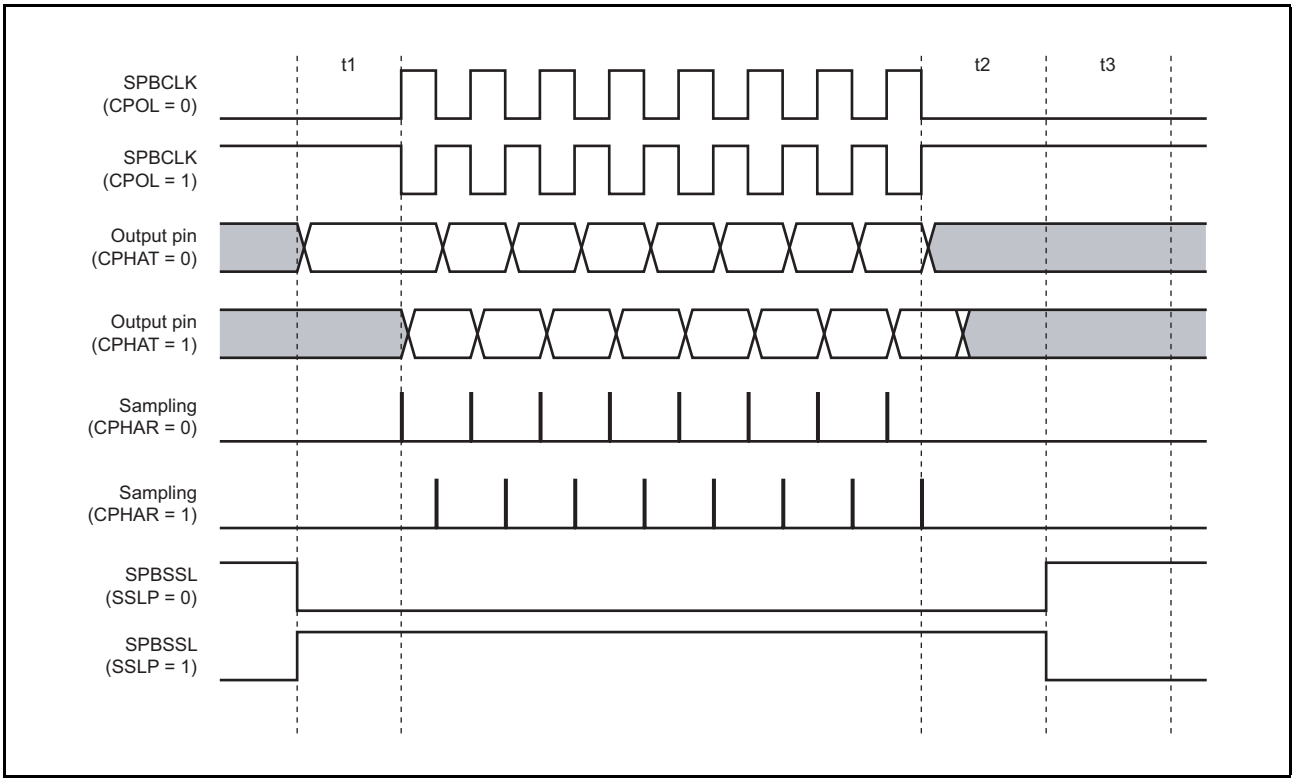


Figure 29.17 SDR Transfer Format

29.3.10 Data Format

This module can input and output data in the order of command, optional command, address, option data, dummy cycle, and data.

(1) Data Registers

Table 29.5 shows the input and output data.

Table 29.5 Data Registers

Data	External Address Space Read Mode	SPI Operating Mode
Command (8 bits)	CMD[7:0] bits in DRCMR	CMD[7:0] bits in SMCMR
Optional command (8 bits)	OCMD[7:0] bits in DRCMR	OCMD[7:0] bits in SMCMR
Address (32/24 bits)	32 bits: DREAR.EAV[6:1 to 0] bits + lower [25 to 24:0] bits of the read address. 24 bits: Lower [23:0] bits of the read address	32 bits: ADR[31:0] bits in SMADR 24 bits: ADR[23:0] bits in SMADR
Option data (8 bits × 4)	DROPR	SMOPR
Dummy cycle (1 to 8 cycles)	DRDMCR	SMDMCR (only when read)
Transfer data	Normal read: 8/16/32 bits Burst read: 64 × RBURST bits	Read: SMRDR0, SMRDR1 Write: SMWDR0, SMWDR1

(2) Data Enable

In external address space read mode, transfer enable of the command, optional command, address, option data, and dummy cycle can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in DRENR, respectively. The size and number of dummy cycles can be controlled with the data read dummy cycle setting register (DRDMCR). Similarly, in SPI operating mode, enable of the command, optional command, address, option data, dummy cycle, and transfer data can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], DME, and SPIDE[3:0] bits in SMENR, respectively. However, disabling all the above parameters is prohibited in SPI operating mode. At least one of them except dummy cycle must be enabled. The size and number of dummy cycles can be controlled with the SPI mode dummy cycle setting register (SMDMCR).

For the address and option data in external address space read mode; and the address, option data, and transfer data in SPI operating mode, the enable bit setting allowed is determined according to the transfer data size. For the allowed setting combinations of the enable bits and transfer data size, refer to the description of the pertinent register.

If these enable bits are disabled, the data is not output, and input and output of the next data is carried out. The command, optional command, address, and option data are always output. During dummy cycles, the state of the used pins is Hi-Z. In external address space read mode, data is always input; and in SPI operating mode, input and output of data is determined based on the settings of the SPIRE and SPIWE bits in SMCR.

There are some restrictions on dummy cycle insertion; refer to the description of the DME bits in DRENR and SMENR for details.

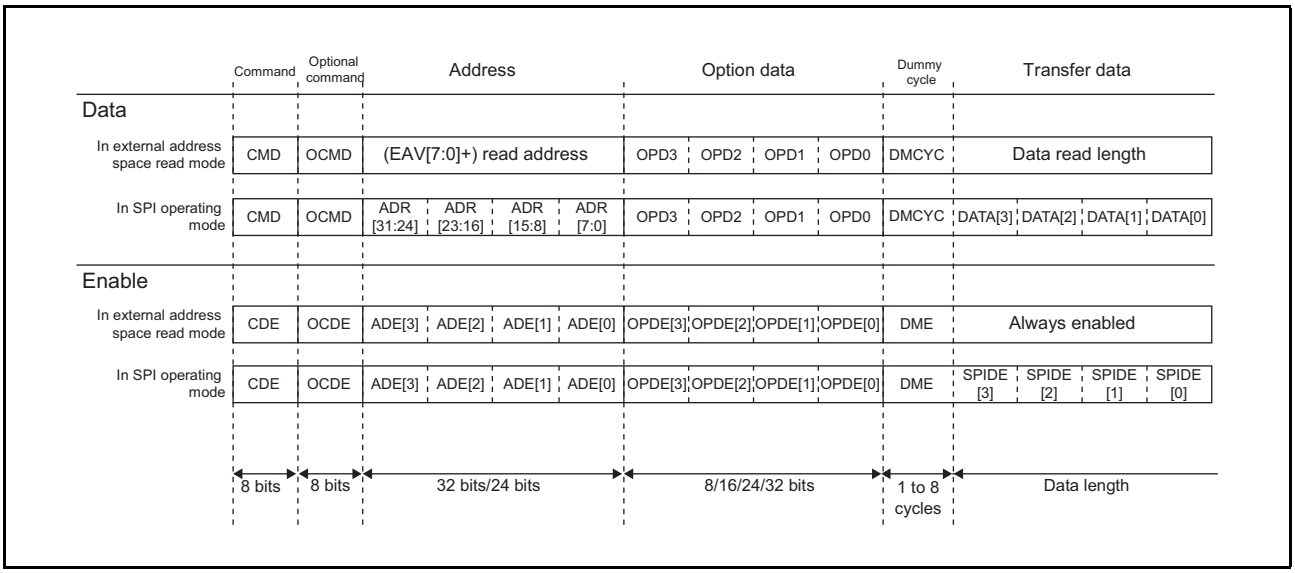


Figure 29.18 Data and Enable

(3) Bit Size

In external address space read mode, the size of the command, optional command, address, option data, and the read data in bit units is respectively controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and DRDB[1:0] bits in DRENR. The size of the dummy cycle in bit units is also controlled with the DMDB[1:0] bits in DRDMCR.
Similarly, in SPI operating mode, the size of the command, optional command, address, option data, and read write data in bit units is controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and SPIDB[1:0] bits in SMENR. The size of the dummy cycle in bit units is also controlled with the DMDB[1:0] bits in SMDMCR.

(a) 1-bit size

When the size is set to 1 bit, SPBMI pin will be the input pins and SPBMO pin will be the output pins. SPBIO2, and SPBIO3 pins are not used.
Figure 29.19 show the transfer format example.

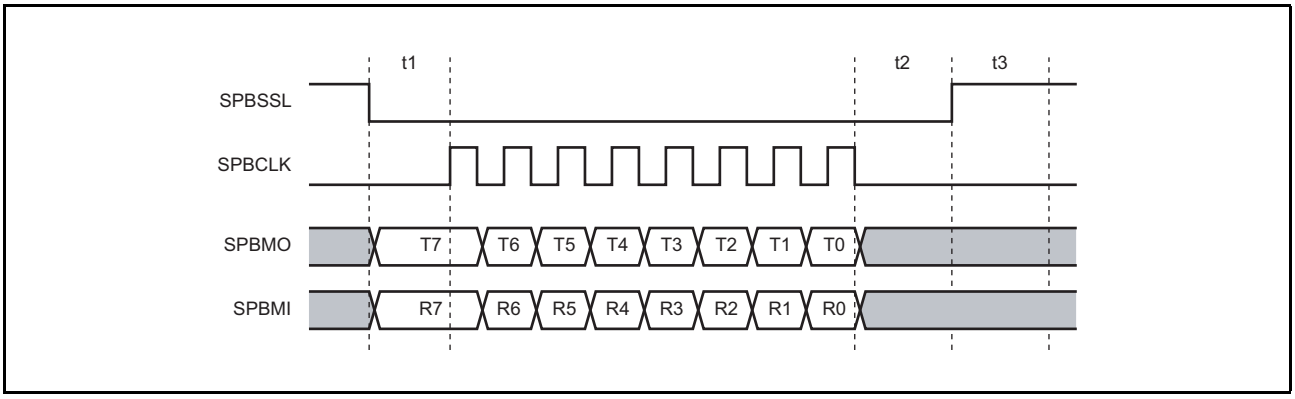


Figure 29.19 Transfer Format Example with 1-Bit Data Size and One Serial Flash Memory Connected

(b) 2-bit size

When the size is set to 2 bits, SPBIO0 and SPBIO1 pins will be either the input pins or the output pins. SPBIO2, and SPBIO3 pins are not used.
Figure 29.20 show the transfer format example.

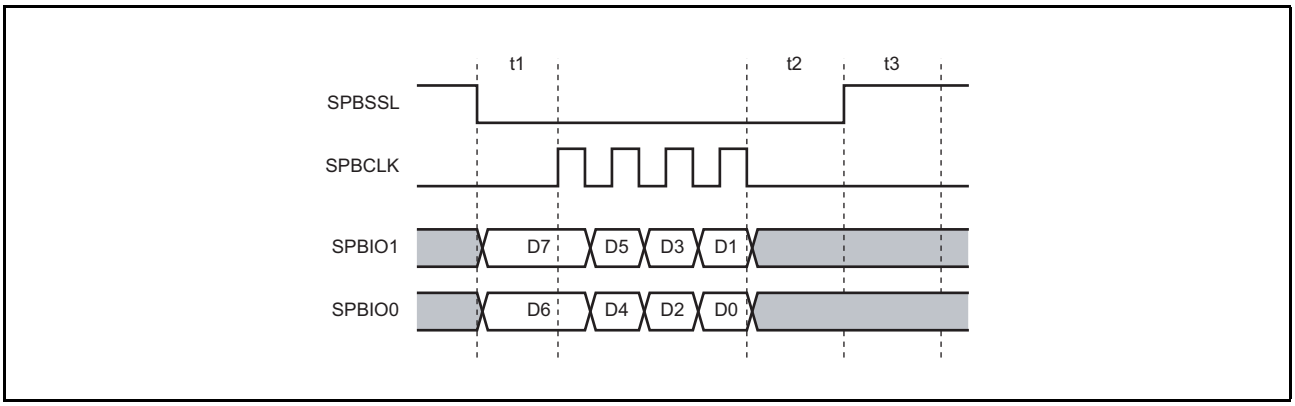


Figure 29.20 Transfer Format Example with 2-Bit Data Size and One Serial Flash Memory Connected

(c) 4-bit size

When the size is set to 4 bits, SPBIO0, SPBIO1, SPBIO2, and SPBIO3 pins will be either the input pins or the output pins.

Figure 29.21 show the transfer format examples.

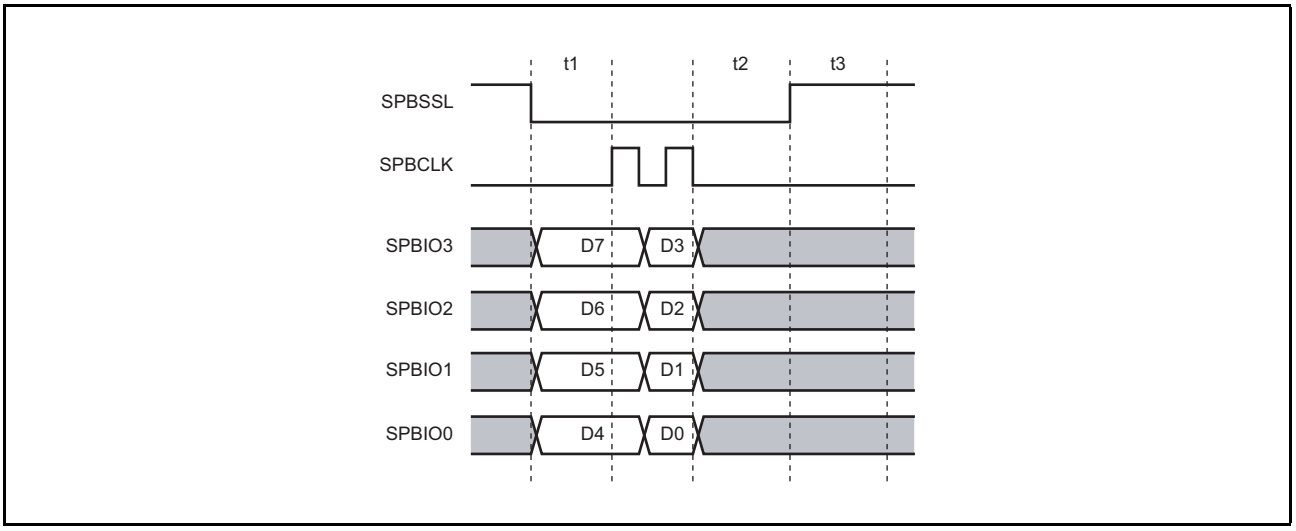


Figure 29.21 Transfer Format Example with 4-Bit Data Size and One Serial Flash Memory Connected

29.3.11 Data Pin Control

With this module, the status of pins can be automatically changed based on the data size to be used and the read/write settings. The inactivation status of the SPBSSL signal can be set with the MOIIO3, MOIIO2, MOIIO1, and MOIIO0 bits in CMNCR. The SPBSSL and SPBCLK pins are always output pins. The status of respective pins is specified in Table 29.6 to Table 29.9.

Table 29.6 Pin Status (1)

		SPBSSL Activation		
		Command, Optional Command, Address, Option Data		
Pin	**SPBSSL Inactivation**	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**
SPBMO/SPBIO0	MOIIO0 bit value	Output	Output	Output
SPBMI/SPBIO1	MOIIO1 bit value	Hi-Z	Output	Output
SPBIO2	MOIIO2 bit value	IO2FV bit value	IO2FV bit value	Output
SPBIO3	MOIIO3 bit value	IO3FV bit value	IO3FV bit value	Output

Table 29.7 Pin Status (2)

	Transfer Data					
	External Address Space Read Mode			SPI Operating Mode		
				SPIRE Bit = 1, SPIWE Bit = 0		
Pin	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**
SPBMO/SPBIO0	IO0FV bit value	Input	Input	IO0FV bit value	Input	Input
SPBMI/SPBIO1	Input	Input	Input	Input	Input	Input
SPBIO2	MOIIO2 bit value	MOIIO2 bit value	Input	MOIIO2 bit value	MOIIO2 bit value	Input
SPBIO3	MOIIO3 bit value	MOIIO3 bit value	Input	MOIIO3 bit value	MOIIO3 bit value	Input

Table 29.8 Pin Status (3)

	Transfer Data					
	SPI Operating Mode					
	SPIRE Bit = 0, SPIWE Bit = 1			SPIRE Bit = 1, SPIWE Bit = 1		
Pin	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**
SPBMO/SPBIO0	Output	Output	Output	Output	Setting prohibited	Setting prohibited
SPBMI/SPBIO1	Hi-Z	Output	Output	Input	Setting prohibited	Setting prohibited
SPBIO2	MOIIO2 bit value	MOIIO2 bit value	Output	MOIIO2 bit value	Setting prohibited	Setting prohibited
SPBIO3	MOIIO3 bit value	MOIIO3 bit value	Output	MOIIO3 bit value	Setting prohibited	Setting prohibited

Table 29.9 Pin Status (4)

	Dummy Cycle		
Pin	**1-Bit Size**	**2-Bit Size**	**4-Bit Size**
SPBMO/SPBIO0	IO0FV bit value	Hi-Z	Hi-Z
SPBMI/SPBIO1	Hi-Z	Hi-Z	Hi-Z
SPBIO2	IO2FV bit value	IO2FV bit value	Hi-Z
SPBIO3	IO3FV bit value	IO3FV bit value	Hi-Z

29.3.12 SPBSSL Pin Control

Inactivation conditions of the SPBSSL signal are as follows.

(1) External Address Space Read Mode

(a) Normal read operation (RBE bit in DRCR = 0)

SPBSSL inactivated after the data transfer and t2 cycle are completed.

(b) Burst read without automatic SPBSSL inactivation (RBE bit in DRCR = 1, SSLE bit in DRCR = 0)

SPBSSL inactivated after the data transfer and t2 cycle are completed.

(c) Burst read with automatic SPBSSL inactivation (RBE bit in DRCR = 1, SSLE bit in DRCR = 1)

- SPBSSL inactivated after t2 cycle when the read address is not continuous with the previously read address
- SPBSSL inactivated after the SSLN bit in DRCR is set to 1

(2) SPI Operating Mode

(a) SPBSSL pin activation not retained (SSLKP bit in SMCR = 0)

SPBSSL inactivated after the data transfer and t2 cycle are completed.

(b) SPBSSL pin activation retained (SSLKP bit in SMCR = 1)

SPBSSL not inactivated.
When to be inactivated, data should be transferred after setting the SSLKP bit to 0.

29.3.13 Flags

This module has two flag bits SSLF and TEND in CMNSR. These bits are read-only bits.

(1) SSLF Bit

This bit indicates the SPBSSL pin status. The status is 1 when the SPBSSL signal is active, and the status is 0 when the SPBSSL signal is inactive.

(2) TEND Bit

This bit indicates whether transfer of data is in progress or the transfer of data has ended.
During t1 time period, data transfer, t2 time period, t3 time period, and waiting for read access by burst read and SPBSSL automatic inactivation, the TEND bit is read as 0 to indicate that the transfer of data is in progress.
When other than the above, the TEND bit is read as 1 to indicate that transfer of data has ended.

(3) Register Re-writing Timing

The status of the TEND bit determines the rewritable registers.
The registers which can be written to, except the SSLN bit in DRCR, should be modified when TEND = 1.
Read SMRDR0 when TEND = 1.
CMNSR can always be read.

29.4 Usage Notes

29.4.1 Notes on Transfer to Read Data in SPI Operating Mode

If the setting for the bit mode is for division by two or more in SPI operating mode, take note of the following points for caution when setting the SPI mode enable setting register (SMENR) to enable transfer only for reading data.
“Transfer only for reading data” indicates transfer to read data while the CDE, OCDE, ADE[3:0], and OPDE[3:0] bits in SMENR are all 0.

(1) Transfer to read data while the signal on the SPBSSL pin is inactivated

Set the SMENR.SPIDE[3:0] bits to 1100b or 1111b when transfer only for reading data is to proceed.
Transfer will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000b.

(2) Transfer to read data while the signal on the SPBSSL pin is activated

When transfer only for reading data is to proceed, set the SMENR.SPIDE[3:0] bits to 1100b or 1111b, or end the immediately preceding transfer with reading data.
When the immediately preceding transfer is of a command, optional command, address, or option data, or is transfer for writing data, the subsequent transfer only for reading data will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000b.

29.4.2 Notes on Starting Transfer from the SPBSSL Signal Retained State in SPI Operating Mode

Be sure to set the SPIWE bit in the SMCR register to 1 when the transfer of a command, optional command, address, or option data is started while the SPBSSL signal is active in SPI operating mode.

29.4.3 Note on Initialization

When using this module, do not set both SPBR[7:0] = 00h and BRDV[1:0] = 00b in the bit rate setting register (SPBCR).

30. CRC Operation Units (CRC)

Cyclic redundancy check (CRC) operation units generate CRC codes.

30.1 Overview

Table 30.1 describes the CRC operation unit specifications. Figure 30.1 shows a block diagram of a CRC operation unit.

Table 30.1 CRC Operation Unit (CRC) Specifications

Item	Specifications
Data subject to CRC operation	A CRC code can be generated for any data that is 8, 16, or 32 bits long.
CRC generation polynomial expression	One of the following polynomials can be selected: • 32-bit Ethernet CRC (32-Ethernet) $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ • 16-bit CCITT CRC (16-CCITT) $X^{16} + X^{12} + X^5 + 1$ • 8-bit SAE J1850 CRC (8-SAE J1850) $X^8 + X^4 + X^3 + X^2 + 1$ • 8-bit 0x2F CRC (8-0x2F) $X^8 + X^5 + X^3 + X^2 + X + 1$
Low-power consumption function	Module-stop state can be set.

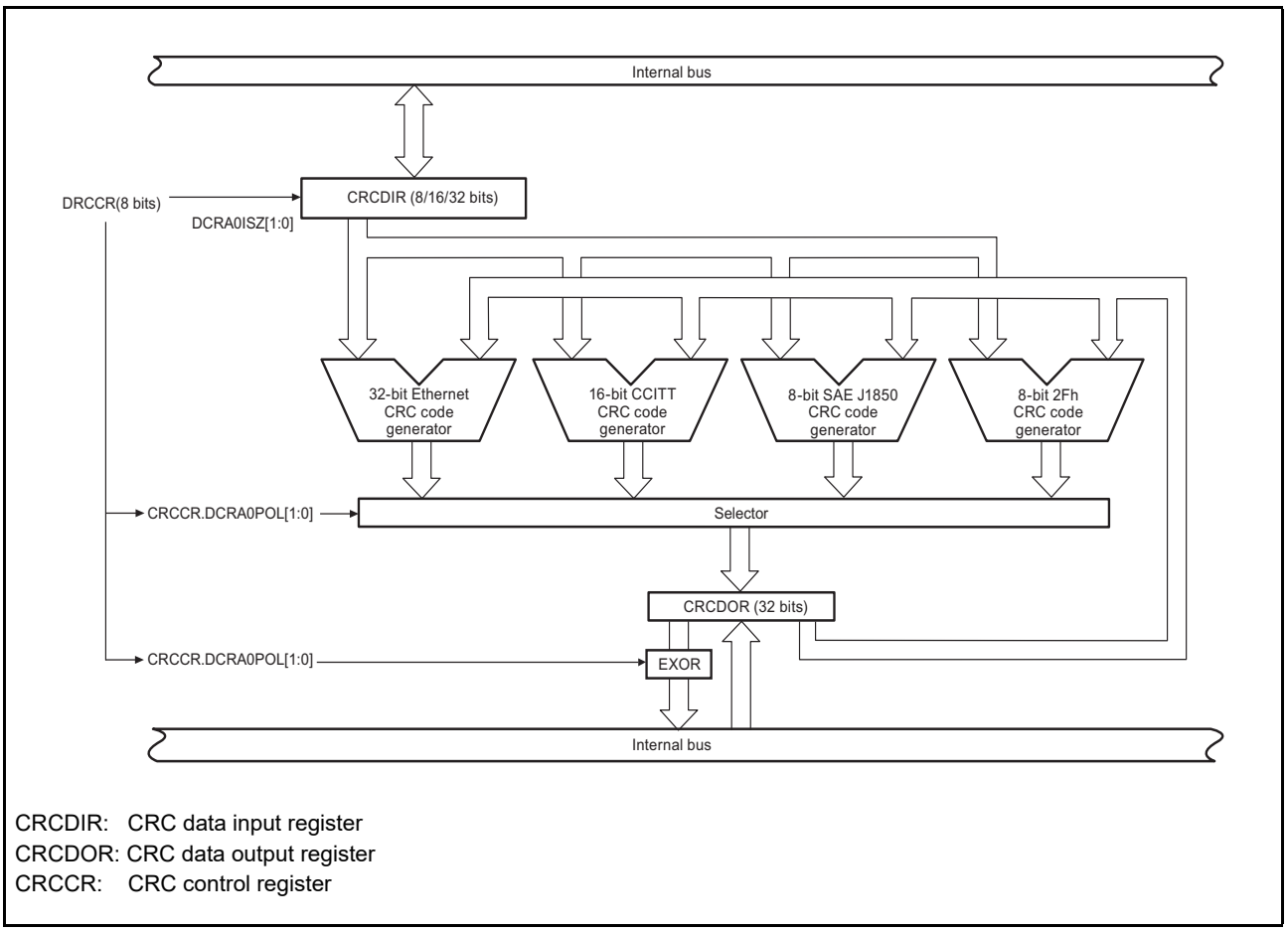


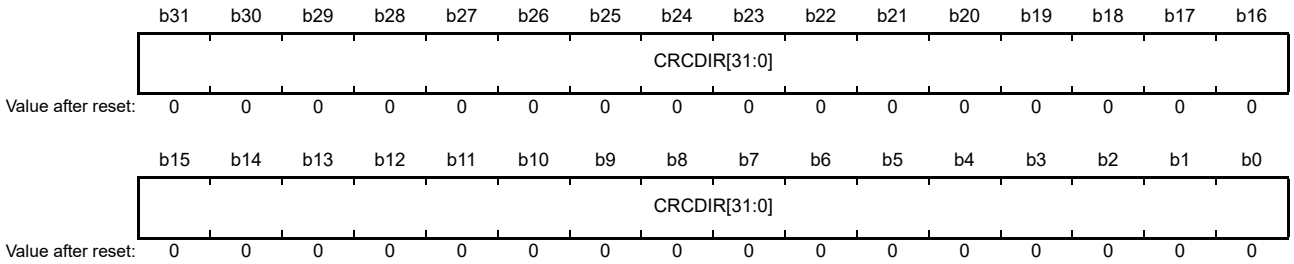
Figure 30.1 Block Diagram of a CRC Operation Unit (CRC)

30.2 Register Descriptions

30.2.1 CRC Data Input Register (CRCDIR)

The CRCDIR register stores the input data for CRC calculation. CRC calculation starts when data is written to this register.
The valid bit width used for CRC calculation must be set for CRCCR.DCRA0ISZ[1:0]. Before the first data is written to this register, the CRCDOR register must be initialized by writing the initial starting value. For details on initialization, see section 30.3.1, Initializing the CRC Data Output Register (CRCDOR).

Address(es): A007 C000h



Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRCDIR[31:0]	Input Data for CRC Calculation	The following bit widths are supported: • CRC input bit width of 32 bits: CRCDIR[31:0] • CRC input bit width of 16 bits: CRCDIR[15:0] • CRC input bit width of 8 bits: CRCDIR[7:0]	R/W

Byte order

The byte order of the CRCDIR register differs depending on the selected CRC generation method.

- If the 32-Ethernet CRC polynomial expression is used to generate a code (CRCCR.DCRA0POL[1:0] = 00b), the LSB-first order is used (LSB: least significant byte). For example, if the CRC input bit width is 8 bits (DCRA0ISZ[1:0] = 10b), the bits 7 to 0 of the CRCDIR register are the LSB.
- If the 16-CCITT CRC polynomial expression is used to generate a code (CRCCR.DCRA0POL[1:0] = 01b), the MSB-first order is used (MSB: most significant byte). For example, if the CRC input bit width is 8 bits (DCRA0ISZ[1:0] = 10b), the bits 7 to 0 of the CRCDIR register are the MSB.
- If the 8-SAE J1850 CRC polynomial expression is used to generate a code (CRCCR.DCRA0POL[1:0] = 10b), the MSB-first order is used (MSB: most significant byte). For example, if the CRC input bit width is 8 bits (DCRA0ISZ[1:0] = 10b), the bits 7 to 0 of the CRCDIR register are the MSB.
- If the 8-0x2F CRC polynomial expression is used to generate a code (CRCCR.DCRA0POL[1:0] = 11b), the MSB-first order is used (MSB: most significant byte). For example, if the CRC input bit width is 8 bits (DCRA0ISZ[1:0] = 10b), the bits 7 to 0 of the CRCDIR register are the MSB.

30.2.2 CRC Data Output Register (CRCDOR)

The CRCDOR register stores the CRC code calculated based on the selected CRC generation polynomial expression.

Address(es): A007 C004h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	CRCDOR[31:0]															
Value after reset:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	CRCDOR[31:0]															
Value after reset: *1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Note 1. After a reset, 32-bit Ethernet CRC is selected as the CRC generation polynomial expression. Therefore, when the bits are read, 0000 0000h is obtained as the result of EXOR operation.

Bit	Symbol	Bit Name	Description	R/W
b31 to b0	CRCDOR[31:0]	Resulting CRC Code	• 32-Ethernet: CRCDOR[31:0] • 16-CCITT: CRCDOR[15:0] (Bits 31 to 16 are undefined.) • 8-SAE J1850/8-0x2F: CRCDOR[7:0] (Bits 31 to 8 are undefined.) The value read from this register is the result of EXOR operation with the following value: • 32-Ethernet: FFFF FFFFh • 16-CCITT: 0000h • 8-SAE J1850/8-0x2F: FFh	R/W

Note: This register must be initialized (by setting the initial starting value) before the first data for CRC calculation is written to the CRCDIR register. For details on initialization, see section 30.3.1, Initializing the CRC Data Output Register (CRCDOR).

CRCDOR[31:0] Bits

The CRC code calculated based on the CRC generation polynomial expression selected by CRCCR.DCRA0POL[1:0] is stored.

Resulting CRC code:

- If 32-Ethernet is used, the resulting CRC code is returned to CRCDOR[31:0].
- If 16-CCITT is used, the resulting CRC code is returned to CRCDOR[15:0].
  The settings of bits 31 to 16 are undefined.
- If 8-SAE J1850/8-0x2F is used, the resulting CRC code is returned to CRCDOR[7:0].
  The settings of bits 31 to 8 are undefined.

The value read from the bits is the result of EXOR operation with the following EXOR value:

EXOR value:

- 32-Ethernet: FFFF FFFFh
- 16-CCITT: 0000h
- 8-SAE J1850/8-0x2F: FFh

After a reset, 32-bit Ethernet is selected as the CRC generation polynomial expression. Therefore, when the bits are read, 0000 0000h is obtained as the result of EXOR operation of FFFF FFFFh (initial value stored in the bits) and FFFF FFFFh (EXOR value).

Example:

Assume that the value of the CRCDOR[31:0] bits is 5555 5555h as the CRC code produced by the 32-bit Ethernet CRC algorithm. In this case, when these bits are read, AAAA AAAAh is obtained as the result of EXOR operation with the EXOR value for this algorithm, i.e. FFFF FFFFh.

30.2.3 CRC Control Register (CRCCR)

The CRCCR register controls the CRC generation polynomial expression and the CRC input bit width.

Address(es): A007 C020h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	DCRA0ISZ[1:0]		—	—	DCRA0POL[1:0]	
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	DCRA0POL [1:0]	CRC Generation Mode Specification	Specify the mode in which to generate a CRC code. b1 b0 0 0: 32-Ethernet ($X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$) 0 1: 16-CCITT ($X^{16} + X^{12} + X^5 + 1$) 1 0: 8-SAE J1850 ($X^8 + X^4 + X^3 + X^2 + 1$) 1 1: 8-0x2F ($X^8 + X^5 + X^3 + X^2 + X + 1$)	R/W
b3, b2	—	Reserved	These bits are read as 0.	R
b5, b4	DCRA0ISZ [1:0]	CRC Input Bit Width Specification	Specify the CRC input bit width. b1 b0 0 0: 32 bits (CRCDIR[31:0]) 0 1: 16 bits (CRCDIR[15:0]) 1 0: 8 bits (CRCDIR[7:0]) 1 1: Setting prohibited	R/W
b7, b6	—	Reserved	These bits are read as 0.	R

Note 1. If the CRC generation mode (CRCCR.DCRA0POL) is changed or the CRC input bit width (CRCCR.DCRA0ISZ) is changed, the CRCDOR register must be initialized (by setting the initial starting value). For details, see section 30.3.1, Initializing the CRC Data Output Register (CRCDOR).

Note 2. The CRC input bit width (CRCCR.DCRA0ISZ[1:0]) must be set according to the block unit of data for CRC calculation. The CRC input bit width must not be changed during CRC calculation. The CRC input bit width can be changed after the final CRC calculation result is read from the CRCDOR register. In this case, before the next data for CRC calculation is written to the CRCDIR register, the CRCDOR register must be initialized (by setting the initial starting value).

30.3 Operation

The CRC operation unit calculates and generates the CRC code for a block of a specific length. Data for which the CRC is to be calculated can be set in the CRC data input register (CRCDIR) in 8-, 16-, or 32- bit units. When data are written to the CRC data input register (CRCDIR), CRC calculation based on the selected CRC generation polynomial expression starts. Before writing the first value to the CRC data input register (CRCDIR), the CRCDOR register must be initialized by setting it to its initial value.
The following shows an overview of using the CRC operation unit.

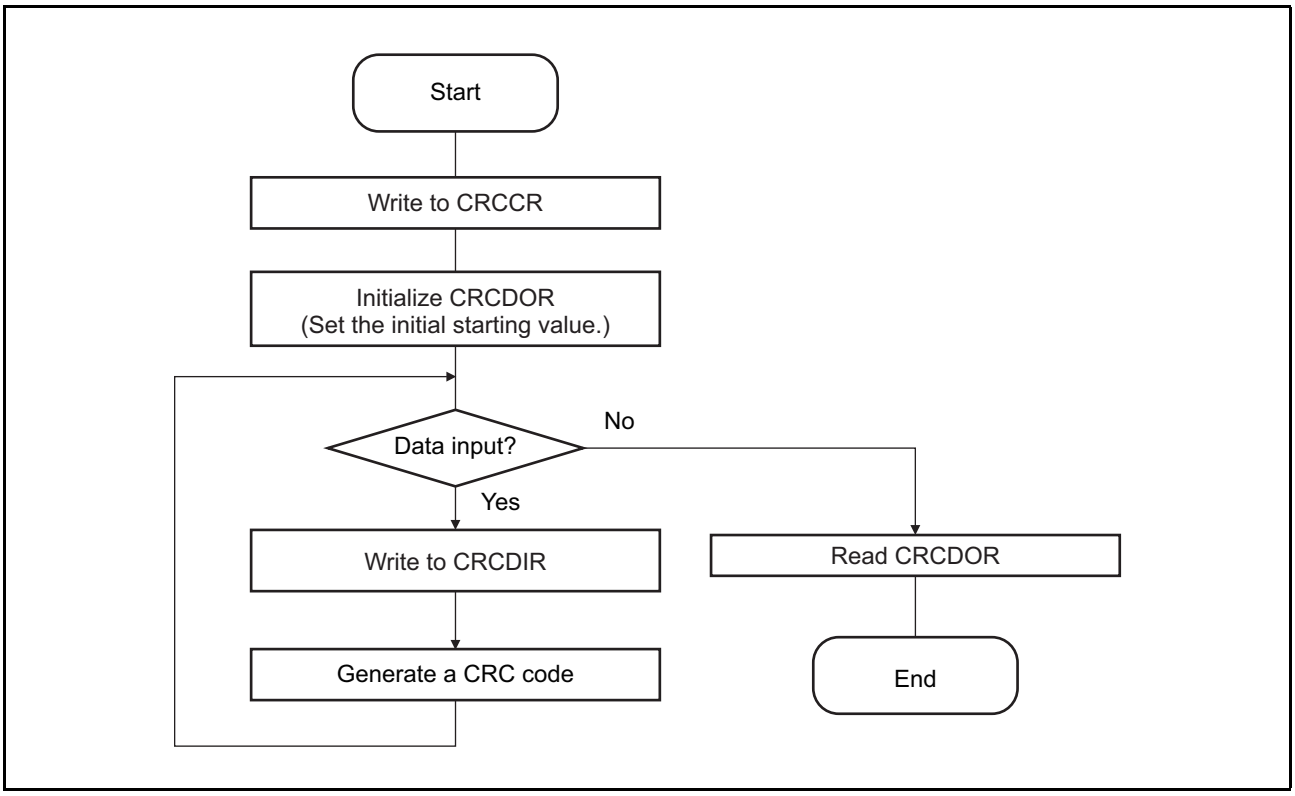


Figure 30.2 Use of the CRC Operation Unit

Note: If the CRC generation polynomial expression is changed by changing the value of CRCCR.DCRA0POL[1:0], the CRC data output register (CRCDOR) must be re-initialized (by setting the initial starting value).

30.3.1 Initializing the CRC Data Output Register (CRCDOR)

Before the first data is written to the CRC data input register (CRCDIR), the CRC data output register (CRCDOR) must be initialized by setting the initial starting value. Table 30.2 lists the initial starting value for each CRC generation polynomial expression.
Table 30.2 also shows the value obtained from the CRCDOR register after the initial starting value is set, as well as the EXOR value used for EXOR operation during a read. For details on the value obtained from the CRCDOR register, see section 30.2.2, CRC Data Output Register (CRCDOR).

Table 30.2 Initial Starting Value for Each CRC Generation Polynomial Expression

CRC Generation Polynomial Expression	Initial Starting Value	EXOR Value	Value Obtained from CRCDOR after the Initial Starting Value Is Set
32-Ethernet (DCRA0POL[1:0] = 00b)	FFFF FFFFh	FFFF FFFFh	0000 0000h
16-CCITT (DCRA0POL[1:0] = 01b)	0000 FFFFh	0000 0000h	0000 FFFFh
8-SAE J1850 (DCRA0POL[1:0] = 10b)	0000 00FFh	0000 00FFh	0000 0000h
8-0x2F (DCRA0POL[1:0] = 11b)	0000 00FFh	0000 00FFh	0000 0000h

31. Boundary Scan

This LSI has the boundary scan function.
The boundary scan is a serial I/O interface based on the JTAG (Joint Test Action Group, IEEE Std.1149.1 and IEEE Standard Test Access Port and Boundary-Scan Architecture).

31.1 Overview

Table 31.1 lists the specifications of boundary scan.
Figure 31.1 shows a block diagram of the boundary scan function.

Table 31.1 Specifications of Boundary Scan

Item	Specifications
Boundary scan enabled/disabled	Boundary scan is enabled when the BSCANP pin is driven high.
Dedicated boundary scan pins	The TDO, TCK, TDI, TMS, and TRST# pins are dedicated for the JTAG when the boundary scan function is enabled.
Six test modes	BYPASS mode EXTEST mode SAMPLE/PRELOAD mode CLAMP mode HIGHZ mode IDCODE mode

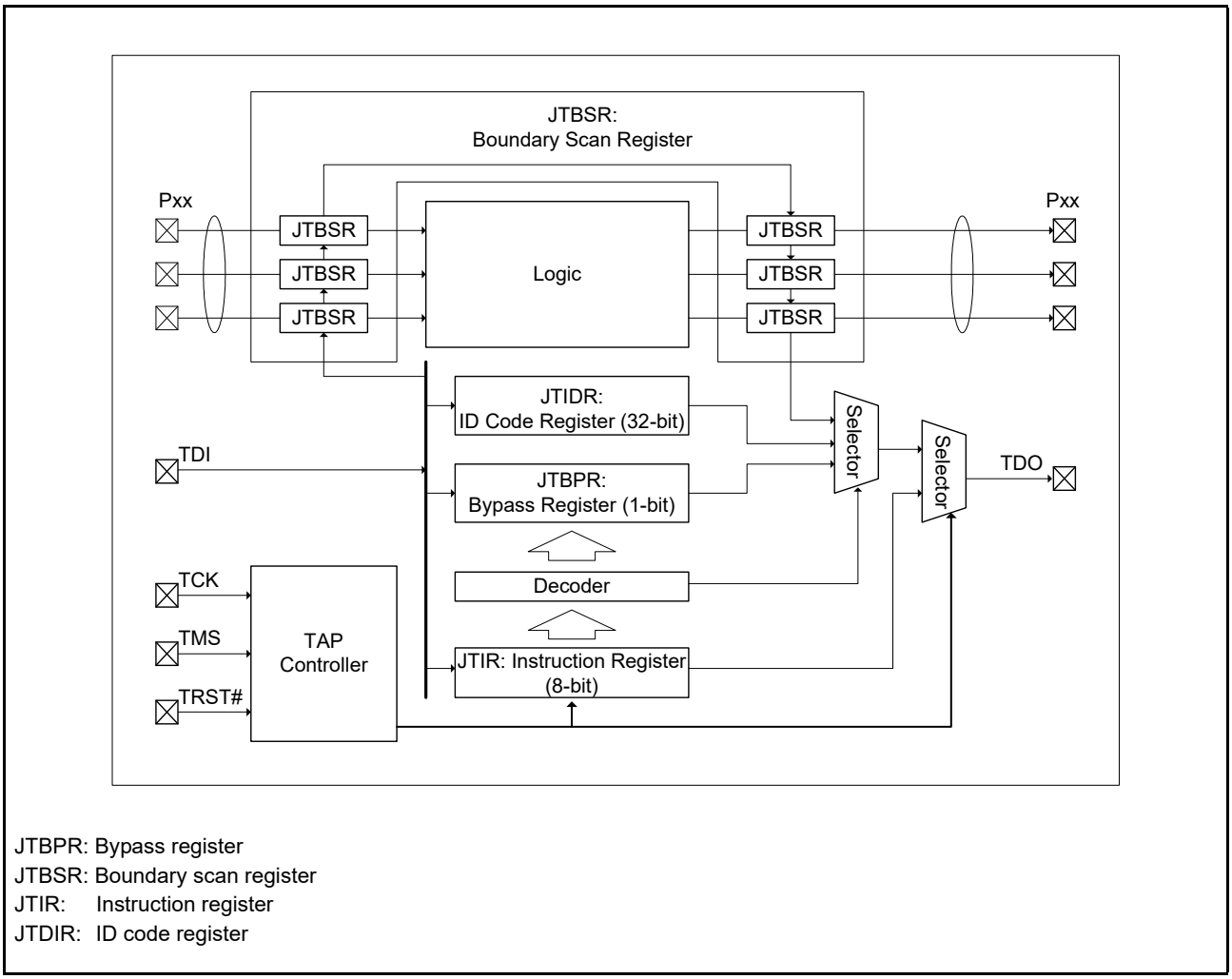


Figure 31.1 JTAG Block Diagram

Table 31.2 shows the I/O pins used in the boundary scan function.

Table 31.2 Pin Configuration of JTAG

Pin Name	I/O	Description
TCK	Input	Test clock input pin Clock signal for boundary scan. Input the clock the duty cycle of which is 50% when boundary scan function is used.
TMS	Input	Test mode select pin
TDI	Input	Test data input pin
TDO	Output	Test data output pin
TRST#	Input	Test reset input pin

31.2 Register Descriptions

Instructions can be input to the instruction register via the TDI pin by serial transfer.
The bypass register, which is a 1-bit register, is connected between the TDI and TDO pins in BYPASS mode.
The boundary scan register, which is configured according to Table 31.5, is connected between the TDI and TDO pins when test data are being shifted in.
None of the registers is accessible from the CPU. Table 31.3 shows the availability of serial transfer for the registers.

Table 31.3 Serial Transfer for the Registers

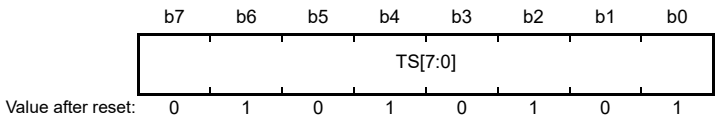
Register	Serial Input	Serial Output
Instruction register	Available	Available
ID code register	Available	Available
Bypass register	Available	Available
Boundary scan register	Available	Available

Note: Any serial transfer is available if standards for the boundary scan function are satisfied.

31.2.1 Instruction Register (JTIR)

The JTIR register is an 8-bit register.
Boundary scan instructions can be transferred to the instruction register by serial input from the TDI pin.
The instruction register is initialized when the TRST# pin is driven to the low level, or when the TAP controller is in the Test-Logic-Reset state.

Address(es): —



Bit	Symbol	Bit Name	Description	R/W
b7 to b0	TS[7:0]	Test Bit Set	The command configuration is as shown in Table 31.4.	—

Table 31.4 Command Configuration

TS7	TS6	TS5	TS4	TS3	TS2	TS1	TS0	Instruction
0	0	0	0	0	0	0	0	EXTEST
0	1	0	0	0	0	0	0	SAMPLE/PRELOAD
0	1	0	1	0	1	0	1	IDCODE (value after reset)
1	1	0	1	0	0	0	0	CLAMP
1	0	0	0	0	0	0	0	HIGHZ
1	1	1	1	1	1	1	1	BYPASS
Other than above								Reserved

31.2.2 ID Code Register (JTIDR)

The JTIDR register is a 32-bit register.
ID code register data is output from the TDO pin when the IDCODE instruction is executed.

Address(es): —

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	DID[31:0]															
Value after reset:	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	DID[31:0]															
Value after reset:	0	1	1	0	0	1	0	0	0	1	0	0	0	1	1	1

Bit	Symbol	Bit Name	Description	R/W
b31-b0	DID[31:0]	Reserved	This register has the fixed value that indicates the device IDCODE.	—

31.2.3 Bypass Register (JTBPR)

The JTBPR register is a 1-bit register.
This register is connected between the TDI and TDO pins when BYPASS mode is set.
The JTBPR register cannot be read from or written to by the CPU.

31.2.4 Boundary Scan Register (JTBSR)

The JTBSR register is a shift register which is located in the pad to control the external input and output pins of this LSI chip.
The EXTEST, SAMPLE/PRELOAD, CLAMP, and HIGHZ commands can be used to perform boundary-scan testing.
Table 31.5 shows the correspondence between the JTBSR bits and the pins of this LSI.
The value is undefined after a reset.

Table 31.5 Boundary Scan Register (196BGA) (1 / 8)

From TDI			
Pin No	**Pin Name**	**Type**	**Bit Name**
M4	P60	Output	630
		Output enable	629
		Input	628
N4	P61	Output	624
		Output enable	623
		Input	622
L4	P62	Output	621
		Output enable	620
		Input	619
N5	P63	Output	615
		Output enable	614
		Input	613
M5	P64	Output	609
		Output enable	608
		Input	607
P5	P65	Output	606
		Output enable	605
		Input	604
P4	P66	Output	603
		Output enable	602
		Input	601
P6	PC6	Input	590
P7	PC7	Input	585
L6	PG2	Output	566
		Output enable	565
		Input	564
M6	PG3	Output	560
		Output enable	559
		Input	558
N6	PG4	Output	554
		Output enable	553
		Input	552
M7	PG5	Output	551
		Output enable	550
		Input	549
N7	PG6	Output	545
		Output enable	544
		Input	543
L8	PR1	Output	509
		Output enable	508
		Input	507
M8	P21	Output	497
		Output enable	496
		Input	495
N8	P22	Output	491
		Output enable	490
		Input	489

Table 31.5 Boundary Scan Register (196BGA) (2 / 8)

From TDI			
Pin No	**Pin Name**	**Type**	**Bit Name**
P8	P23	Output	485
		Output enable	484
		Input	483
N9	P20	Output	482
		Output enable	481
		Input	480
P9	P25	Output	479
		Output enable	478
		Input	477
M9	P26	Output	473
		Output enable	472
		Input	471
L9	P27	Output	467
		Output enable	466
		Input	465
M10	P44	Output	464
		Output enable	463
		Input	462
N10	P42	Output	455
		Output enable	454
		Input	453
P10	P40	Output	449
		Output enable	448
		Input	447
M11	PS0	Output	440
		Output enable	439
		Input	438
N11	PS1	Output	434
		Output enable	433
		Input	432
P11	P10	Output	428
		Output enable	427
		Input	426
M12	P00	Output	425
		Output enable	424
		Input	423
P12	PS2	Output	422
		Output enable	421
		Input	420
N12	PS3	Output	416
		Output enable	415
		Input	414
P13	PS4	Output	407
		Output enable	406
		Input	405

Table 31.5 Boundary Scan Register (196BGA) (3 / 8)

From TDI			
Pin No	**Pin Name**	**Type**	**Bit Name**
N13	PS5	Output	398
		Output enable	397
		Input	396
L12	PS6	Output	392
		Output enable	391
		Input	390
N14	PE0	Output	386
		Output enable	385
		Input	384
M13	PS7	Output	383
		Output enable	382
		Input	381
K11	PE1	Output	380
		Output enable	379
		Input	378
M14	PE2	Output	377
		Output enable	376
		Input	375
L13	PE3	Output	371
		Output enable	370
		Input	369
L14	PE4	Output	365
		Output enable	364
		Input	363
K12	PE5	Output	362
		Output enable	361
		Input	360
K13	PE6	Output	356
		Output enable	355
		Input	354
J11	PE7	Output	353
		Output enable	352
		Input	351
K14	P70	Output	350
		Output enable	349
		Input	348
J12	P71	Output	344
		Output enable	343
		Input	342
J13	P72	Output	338
		Output enable	337
		Input	336
J14	P73	Output	335
		Output enable	334
		Input	333

Table 31.5 Boundary Scan Register (196BGA) (4 / 8)

From TDI			
Pin No	**Pin Name**	**Type**	**Bit Name**
H11	P74	Output	332
		Output enable	331
		Input	330
H12	P75	Output	329
		Output enable	328
		Input	327
H13	P76	Output	323
		Output enable	322
		Input	321
G12	P77	Output	317
		Output enable	316
		Input	315
H14	PA0	Output	314
		Output enable	313
		Input	312
G14	PA1	Output	308
		Output enable	307
		Input	306
G13	PA2	Output	305
		Output enable	304
		Input	303
G11	PA3	Output	299
		Output enable	298
		Input	297
F12	PA4	Output	293
		Output enable	292
		Input	291
F13	PA5	Output	284
		Output enable	283
		Input	282
D14	P12	Output	281
		Output enable	280
		Input	279
F14	PA6	Output	278
		Output enable	277
		Input	276
D12	PA7	Output	272
		Output enable	271
		Input	270
E12	P90	Output	269
		Output enable	268
		Input	267
F11	P91	Output	266
		Output enable	265
		Input	264

Table 31.5 Boundary Scan Register (196BGA) (5 / 8)

From TDI			
Pin No	**Pin Name**	**Type**	**Bit Name**
E13	P92	Output	263
		Output enable	262
		Input	261
E14	P93	Output	260
		Output enable	259
		Input	258
D13	P94	Output	257
		Output enable	256
		Input	255
C14	P95	Output	254
		Output enable	253
		Input	252
C13	P96	Output	251
		Output enable	250
		Input	249
B14	P97	Output	248
		Output enable	247
		Input	246
B13	P16	Output	230
		Output enable	229
		Input	228
C12	P17	Output	224
		Output enable	223
		Input	222
D9	P50	Output	215
		Output enable	214
		Input	213
A11	P51	Output	212
		Output enable	211
		Input	210
D8	P52	Output	209
		Output enable	208
		Input	207
B10	P53	Output	206
		Output enable	205
		Input	204
C9	P54	Output	203
		Output enable	202
		Input	201
A10	P56	Output	197
		Output enable	196
		Input	195
B9	PD5	Output	194
		Output enable	193
		Input	192

Table 31.5 Boundary Scan Register (196BGA) (6 / 8)

From TDI			
Pin No	**Pin Name**	**Type**	**Bit Name**
A9	PD6	Output	191
		Output enable	190
		Input	189
C8	PD7	Output	188
		Output enable	187
		Input	186
D7	P86	Output	185
		Output enable	184
		Input	183
B8	P87	Output	182
		Output enable	181
		Input	180
A8	PF5	Output	179
		Output enable	178
		Input	177
C7	PF6	Output	173
		Output enable	172
		Input	171
B7	PB7	Output	167
		Output enable	166
		Input	165
D6	PC0	Input	163
A7	PC1	Input	161
C6	PB0	Output	157
		Output enable	156
		Input	155
B6	PB1	Output	151
		Output enable	150
		Input	149
A6	PB2	Output	145
		Output enable	144
		Input	143
D5	PB3	Output	139
		Output enable	138
		Input	137
C5	PB4	Output	133
		Output enable	132
		Input	131
A5	PB5	Output	127
		Output enable	126
		Input	125
B5	PB6	Output	121
		Output enable	120
		Input	119
A4	PF7	Output	115
		Output enable	114
		Input	113

Table 31.5 Boundary Scan Register (196BGA) (7 / 8)

From TDI			
Pin No	**Pin Name**	**Type**	**Bit Name**
C4	PJ0	Output	112
		Output enable	111
		Input	110
B4	PJ1	Output	109
		Output enable	108
		Input	107
A3	PJ2	Output	106
		Output enable	105
		Input	104
B3	PJ3	Output	103
		Output enable	102
		Input	101
A2	PC2	Input	99
B2	PJ4	Output	98
		Output enable	97
		Input	96
C3	PC3	Input	91
C2	PJ5	Output	87
		Output enable	86
		Input	85
D3	PJ6	Output	72
		Output enable	71
		Input	70
B1	PJ7	Output	69
		Output enable	68
		Input	67
E4	P80	Output	66
		Output enable	65
		Input	64
E3	P82	Output	63
		Output enable	62
		Input	61
E2	P85	Output	57
		Output enable	56
		Input	55
D2	P81	Output	54
		Output enable	53
		Input	52
F3	ERROROUT#	Output	51
		Output enable	50
C1	P83	Output	48
		Output enable	47
		Input	46
F4	P35	Output	45
		Output enable	44
		Input	43

Table 31.5 Boundary Scan Register (196BGA) (8 / 8)

From TDI			
Pin No	**Pin Name**	**Type**	**Bit Name**
D1	P84	Output	42
		Output enable	41
		Input	40
G1	PC4	Input	32
G2	PC5	Input	30
H3	PU7	Output	26
		Output enable	25
		Input	24
J3	PM1	Output	23
		Output enable	22
		Input	21
M1	PM2	Output	20
		Output enable	19
		Input	18
L2	PM3	Output	17
		Output enable	16
		Input	15
M2	PM6	Output	14
		Output enable	13
		Input	12
K3	PM4	Output	11
		Output enable	10
		Input	9
N2	PM5	Output	8
		Output enable	7
		Input	6
N1	PM7	Output	5
		Output enable	4
		Input	3
K4	RSTOUT#	Output	2
		Output enable	1
		To TD0	

31.3 Operation

The boundary scan function is valid when the RES# and BSCANP pins are driven high.

31.3.1 TAP Controller

Figure 31.2 shows the state transition diagram of the TAP controller. Table 31.6 describes each state.

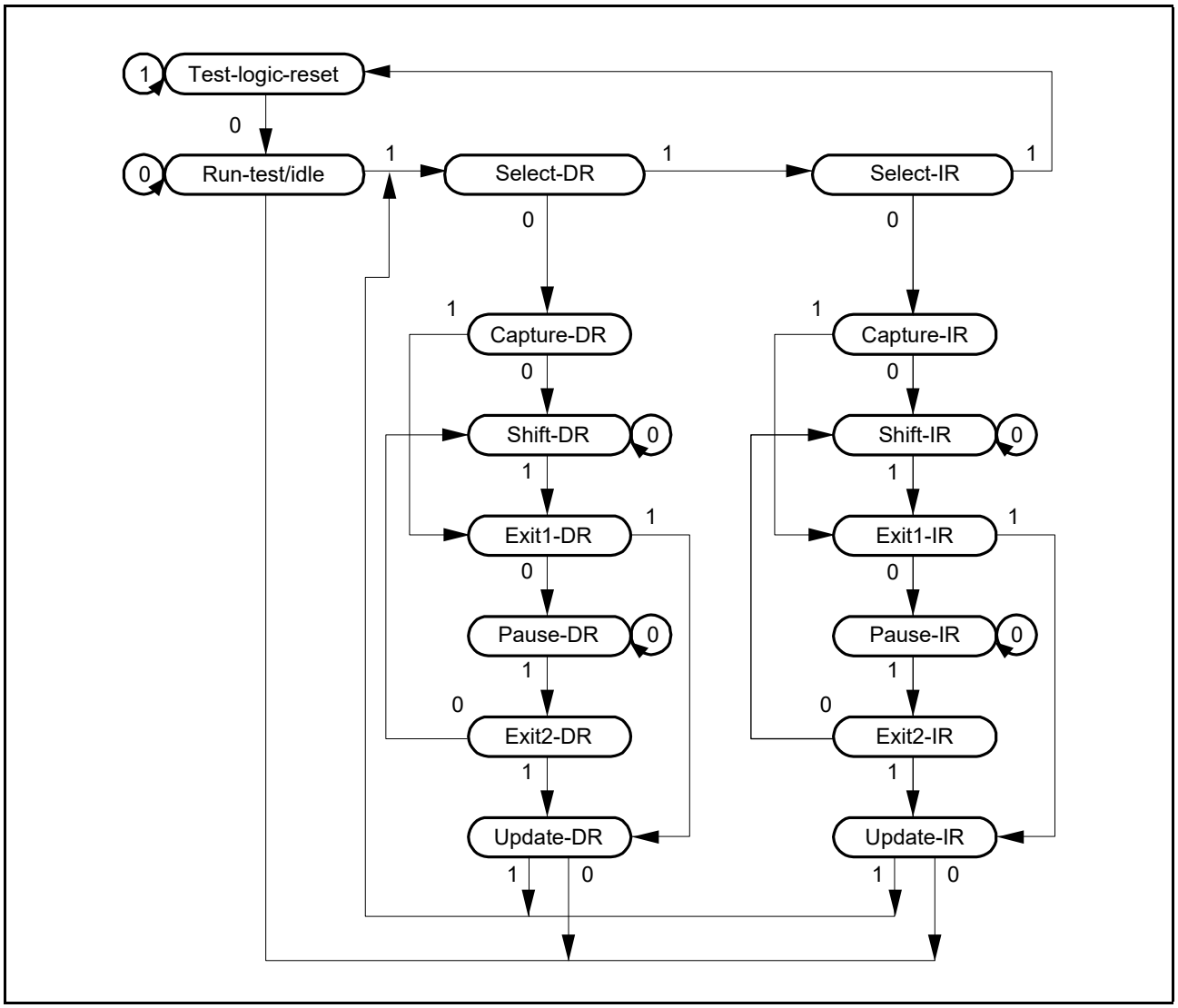


Figure 31.2 State Transition Diagram of TAP Controller

Table 31.6 Explanation of States

State	Explanation
Test Logic Reset	Reset state of the TAP controller. The LSI is in this state during normal operation.
Run Test/Idle	Test execution state
Select DR Scan	Temporary state used for selecting a data register.
Select IR Scan	Temporary state used for selecting an instruction register.
Capture DR	Data of the test data register corresponding to the current instruction is captured in parallel.
Shift DR	The test data register corresponding to the current instruction is connected between the TDI and TDO pins, and data is transferred serially.
Exit DR	Temporary state
Pause DR	Clocks are applied while the value input in the Shift DR state is retained.
Exit2 DR	Temporary state
Update DR	Output latches of the test data register corresponding to the current instruction are updated.
Capture IR	A fixed value is input to the instruction register.
Shift IR	The instruction register is connected between the TDI and TDO pins, and data is transferred serially.
Exit IR	Temporary state
Pause IR	Clocks are applied while the value input in the Shift IR state is retained.
Exit2 IR	Temporary state
Update IR	The current instruction is updated to the instruction input in the Shift IR state.

31.3.2 List of Commands

(1) BYPASS [Instruction Code: 1111 1111b]

The BYPASS instruction drives the bypass register. This instruction shortens the shift path, facilitating the transfer of serial data to other LSIs on a printed-circuit board at higher speeds. While this instruction is being executed, the test circuit does not affect the system circuit.
The bypass register is connected between the TDI and TDO pins. Bypass operation is initiated from shift-DR operation. The TDO is set to 0 in the first clock cycle in the Shift-DR state. In the subsequent clock cycles, the TDI signal is output on the TDO pin.

(2) EXTEST [Instruction Code: 0000 0000b]

The EXTEST instruction is used to test external circuits when this LSI is installed on the printed circuit board. If this instruction is executed, output pins are used to output test data (specified by the SAMPLE/PRELOAD instruction) from the boundary scan register to the print circuit board, and input pins are used to input test result to the boundary scan register from the print circuit board.

(3) SAMPLE/PRELOAD [Instruction Code: 0100 0000b]

The SAMPLE/PRELOAD instruction is used to input data from this LSI's internal circuits to the boundary scan register, output data from the scan path, and reload the data to the scan path. While this instruction is executed, input signals are directly input to this LSI and output signals are also directly output to the external circuits. The system circuit of this LSI is not affected by this instruction.
In SAMPLE operation, the boundary scan register latches the snap shot of data transferred from input pins to internal circuit or data transferred from internal circuit to output pins. The latched data is read from the scan path. The scan register latches the snap shot at the rising edge of the TCK in Capture-DR state. The scan register latches snap shots without affecting the LSI normal operation.
In PRELOAD operation, the value after reset is written from the scan path to the parallel output latch of the boundary scan register prior to the EXTEST instruction execution. If the EXTEST instruction is executed without executing this PRELOAD operation, undefined values are output over the period until the first scan sequence is completed (transfer to the output latch). (In the EXTEST instruction, output parallel latches are always output to the output pins.)

(4) IDCODE [Instruction Code: 0101 0101b]

When the IDCODE instruction is selected, the ID code register value is output from the TDO in LSB-first order in Shift-DR state of the TAP controller. While this instruction is being executed, the test circuit does not affect the system circuit.
The instruction register is initialized by the IDCODE instruction in Test-Logic-Reset state of the TAP controller.

(5) CLAMP [Instruction Code: 1101 0000b]

When the CLAMP instruction is selected, output pins output the boundary scan register value which was specified by the SAMPLE/PRELOAD instruction in advance. While the CLAMP instruction is selected, the status of the boundary scan register is maintained regardless of the TAP controller state.
The bypass register is connected between the TDI and TDO pins, leading to the same operation as when BYPASS mode has been selected

(6) HIGHZ [Instruction Code: 1000 0000b]

When the HIGHZ instruction is selected, all output pins enter high-impedance state. While the HIGHZ instruction is selected, the status of the boundary scan register is maintained regardless of the TAP controller state. The bypass register is connected between the TDI and TDO pins, leading to the same operation as when the BYPASS instruction has been selected.

31.4 Usage Notes

1. For serial transfer, data are input or output in LSB-first order. See Figure 31.3.

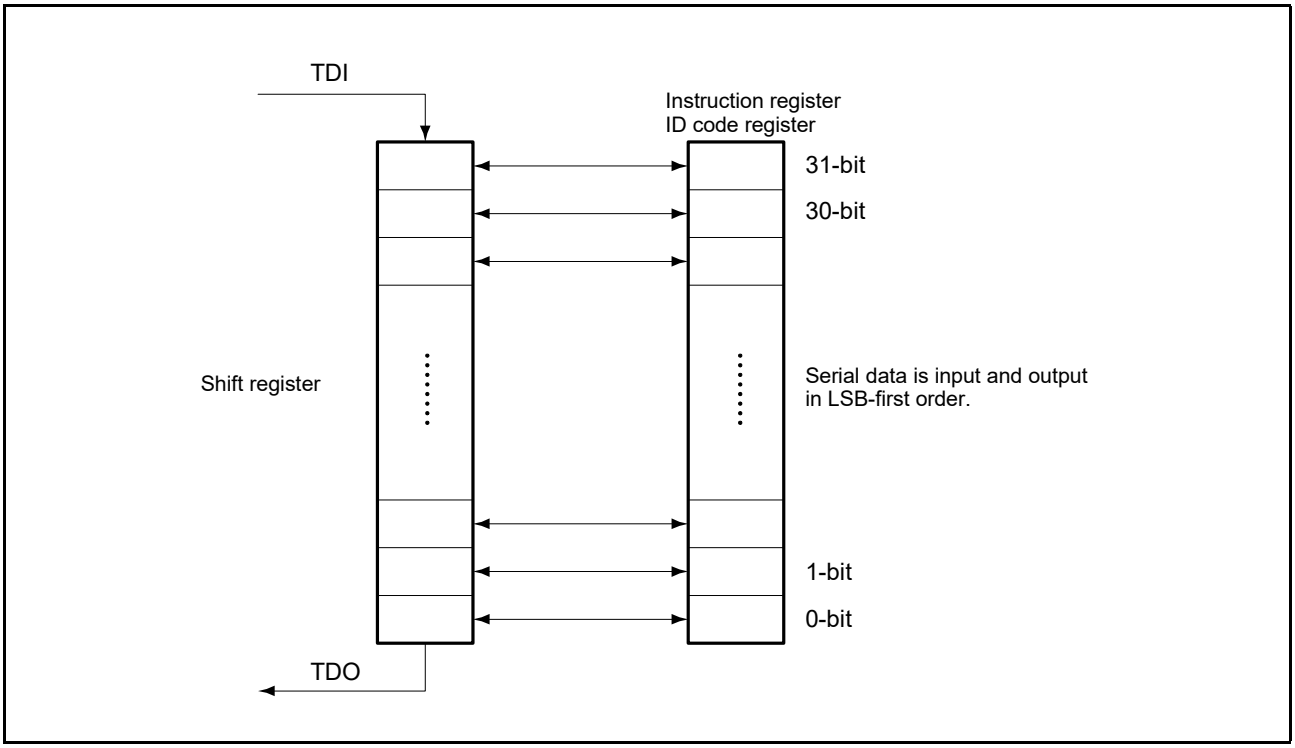


Figure 31.3 Serial Data Input/Output

2. Pins of the boundary scan (TDI, TMS, and TRST#) have to be pulled up by pull-up resistors. However, if an on-chip emulator is in use, handle the #TRST pin according to the manual for the given on-chip emulator.
3. Power supply pins (VDD, VSS, VCCQ33, PLLVDD0, PLLVSS0, PLLVDD1, PLLVSS1, VDD33_USB, VSS_USB, and DVDD_USB) cannot be boundary-scanned.
4. The USB reference current source input pin (USB_RREF) cannot be boundary-scanned.
5. Clock pins (EXTAL and XTAL) cannot be boundary-scanned.
6. The reset pin (RES#) cannot be boundary-scanned.
7. USB dedicated pins (USB_DP and USB_DM) cannot be boundary-scanned.
8. The boundary-scan pin (BSCANP) cannot be boundary-scanned.
9. The boundary-scan pins (TCK, TMS, TRST#, TDI, and TDO) cannot be boundary-scanned.
10. The output registers for I/O port pins (PC0 to PC7) cannot be boundary-scanned.
11. The boundary scan function is not available when the chip is in the reset state.
12. For a pin that incorporates open-drain functionality and for which the open-drain function is enabled, if the boundary-scan function sets the corresponding bit in the output scan register and output enable register to 1, executing an EXTEST, CLAMP, or SAMPLE/PRELOAD instruction makes the pin output the high level rather than placing it in the high-impedance state.

32. Error Control Module (ECM)

This section describes an error control module (ECM).

32.1 Overview

The error control module (ECM) collects error output signals coming from individual peripheral modules. It also outputs error signals from the error output pin (ERROROUT#) and generates error interrupts and internal reset signals. Table 32.1 describes the ECM specifications and Figure 32.1 shows the block diagram of ECM.

Table 32.1 Specifications of ECM

Item	Description
Safety processing	ECM can handle the following processing in response to error signal inputs from individual modules. • Error flag set ECM has flags that indicate the state of error occurrence for individual error sources. • ECM maskable interrupt generation Maskable interrupt generation can be controlled (enabled or disabled) for individual error sources. • ECM non-maskable interrupt generation Non-maskable interrupt generation can be controlled (enabled or disabled) for individual error sources. • ECM reset (Internal reset) Internal reset generation can be controlled (enabled or disabled) for individual error sources. • Error signal output from the ERROROUT# pin Error signal output and mask control (enable or disable) are possible for individual error sources. Output can be toggled in response to a CMTW timer input or made at a fixed level.
Error status	ECM incorporates error status registers, whose error flag values can be used to check whether the corresponding error sources have been generated.
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 error pin output, interrupt, or internal reset apply in the same way. In addition, extended pseudo errors can be used for error detection for functional safely. • ECM incorporates a loop-back function of the error pin output that is used to diagnose the path to the error output pin. The status of the error output pin is reflected to an internal register and can be confirmed by reading the register.
Delay timer timeout	ECM has a function of outputting an error signal or resetting ECM when the delay timer starts at the same time with generation of an ECM maskable interrupt or an ECM non-maskable interrupt and a delay timer overflow occurs because the delay timer cannot be stopped during the interrupt processing.
Others	ECM has a duplexed structure (master and checker) for redundancy.

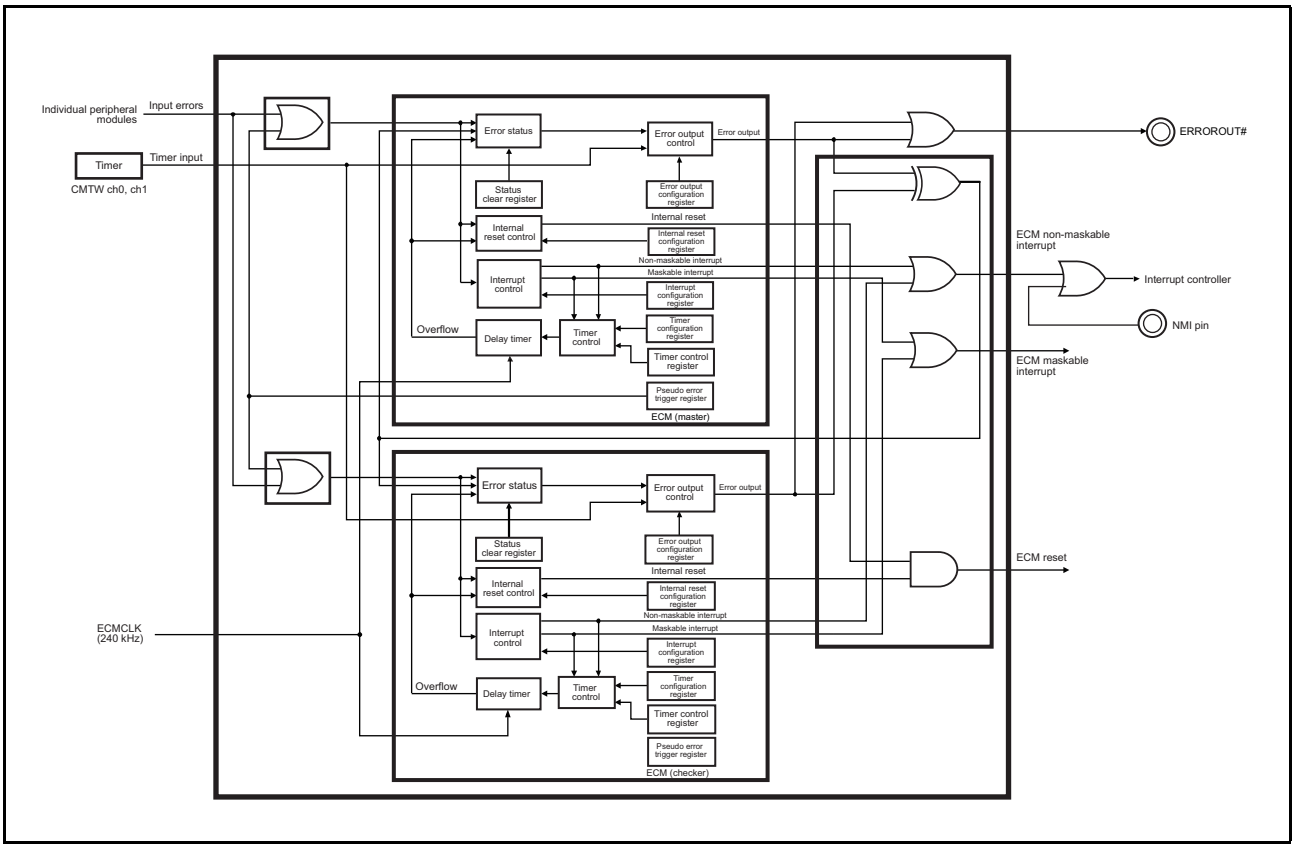


Figure 32.1 Block Diagram of ECM

Table 32.2 shows the error input to ECM.

Table 32.2 ECM Error Input

Error Source Number	Module	Function
1	WDTA	WDTA underflow/refresh error (for Cortex-R4)
2	—	Reserved
3	IWDTa	IWDTa underflow/refresh error
4	—	Reserved
5	Cortex-R4 cache	1-bit or 2-bit ECC error in the instruction cache (Tag RAM)
6		1-bit or 2-bit ECC error in the instruction cache (Data RAM)
7		1-bit ECC error in the data cache (Tag/Dirty RAM)
8		2-bit ECC error in the data cache (Tag/Dirty RAM)
9		1-bit ECC error in the data cache (Data RAM)
10		2-bit ECC error in the data cache (Data RAM)
11	Cortex-R4 RAM	1-bit ECC error in the ATCM
12		2-bit ECC error in the ATCM
13		1-bit ECC error in the BTCM
14		2-bit ECC error in the BTCM
15, 16	—	Reserved
17	RSCAN	1-bit ECC error in the RSCAN RAM
18		2-bit ECC error in the RSCAN RAM
19		RSCAN overflow error
20	Clock monitor circuit (CLMA)	Main clock oscillation stop detection
21		CLMA0 oscillation stop detection (PLL0)
22		CLMA1 oscillation stop detection (PLL1)
23		CLMA2 oscillation stop detection (LOCO)
24 to 32	—	Reserved
33	Internal bus	Bus error
34	—	Reserved
35	—	Extended pseudo error 35*1
36	—	Extended pseudo error 36*1
37	—	Extended pseudo error 37*1
38	—	Extended pseudo error 38*1
39	—	Extended pseudo error 39*1
40	—	Extended pseudo error 40*1
41	—	Extended pseudo error 41*1
42 to 92	—	Reserved
93	Error control module (ECM)	Compare error
94		Delay timer overflow error
95		Error set by the ECMmESET register
96		Loopback error

Note 1. A pseudo error can be generated if the corresponding bit in the ECM pseudo error trigger register (ECMPEn) is set by software. For details, see section 32.3.3, Pseudo Error Generation.

32.2 Registers

32.2.1 ECM Master/Checker Error Set Trigger Register (ECMmESET (m = M or C))

The ECMmSET (m = M or C) register controls the output of the ERROROUT# pin. Setting the ECMmEST bit to 1 can set the output of the ERROROUT# pin to the active level (low level). The ERROROUT# pin output cannot be masked. Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers. This register is always read as 00h.

Address(es): ECMMESET: A007 D000h
ECMCESET: A007 D040h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMmEST
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMmEST	Error Set Trigger	0: Writing 0 is invalid. 1: Sets the output level of the ERROROUT# pin to the active level (low level).	W
b7 to b1	—	Reserved	The write value should be 0.	W

Note 1. Setting the ECMmESET register will set the ECMmSSE228 bit of the ECMmESSTR2 register (ECM compare error). Therefore, to use the ECMmESET register to control the ERROROUT# pin, follow the procedure below.
1. Set the ECMEMK228 bit of the ECMEMK2 register to “masked”.
2. Prevent the generation of interrupts by setting the ECMMIE228 bit of the ECMMICFG2 register to “prohibited” and the ECMNMIE228 bit of the ECMNMICFG2 register to “prohibited”.
3. Prevent generation of an internal reset by setting the ECMIRE228 bit of the ECMIRCFG2 register to “prohibited”.
4. Set the error output bit in the ECMmESET register.
5. Clear error flags by setting the ECMCLSSE228 bit of the ECMESSTC2 register.
6. Make the following settings, as necessary, to change back the ECM compare error settings.
   - To enable the error output from the ERROROUT# pin, set the ECMEMK228 bit in the ECMEMK2 register to “not masked”.
   - To enable an error interrupt, set the ECMMIE228 bit in the ECMMICFG2 register or the ECMNMIE228 bit in the ECMNMICFG2 register to “enabled”.
   - To enable an ECM reset, set the ECMIRE228 bit in the ECMIRCFG2 register to “enabled”.

Note 2. When the error output is set by the ECMmESET register, an error state is retained even after all error sources that have not been masked in the ECMESSTCn register are cleared, and the low level is output from the ERROROUT# pin while the pin stays active. To clear an error state, perform clear processing on the ECMmECLR register.

32.2.2 ECM Master/Checker Error Clear Trigger Register (ECMmECLR (m = M or C))

The ECMmECLR (m = M or C) register controls the output from the ERROROUT# pin. When the ECMmECT bit is set to 1, the output of the ERROROUT# pin can be set to the inactive level (high level) if there are no other error sources that set the ERROROUT# pin to the active level (low level).
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers. This register is always read as 00h.

Address(es): ECMMECLR: A007 D004h
ECMCECLR: A007 D044h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMmECT
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMmECT	Error Clear Trigger	0: Writing 0 has no effect. 1: Sets the output level of the ERROROUT# pin to the inactive level (high level). Note that, however, this bit cannot be used to clear an error while ECM error output clear is disabled. For details, see section 32.3.6, Setting of Disabling Error Output Clear.	W
b7 to b1	—	Reserved	The write value shoud be 0.	W

Note: Clearing the output of the error pin is only possible if all error sources, not masked by ECMEMK0, ECMEMK1, or ECMEMK2, are cleared beforehand. Setting the ECMmECT bit in this register will set the output of the ERROROUT# pin to the inactive level (high level).
If an error state is generated due to the error output set by ECMmESET, use this register to clear the error state.
Follow the procedure below to “clear” error states.

1. Set all bits corresponding to an error in the ECMESSTC0, ECMESSTC1, and ECMESSTC2 registers to clear the error state indicator.
2. To mask the output of error signals due to any error sources, set all bits corresponding to an error in the ECMEMK0, ECMEMK1, and ECMEMK2 registers to “masked”.
3. To prevent unintended processing in response to an ECM comparison error (error source 93), set the ECMMIE228, ECMNMIE228, and ECMIRE228 bits in the ECMMICFG2, ECMNMICFG2, and ECMIRCFG2 registers to “prohibited”.
4. Clear the output on the ERROROUT# pin by using the ECMMECT and ECMCECT bits in the ECMMECLR and ECMCECLR registers. At this time, an ECM comparison error will be generated.
5. Use software to set up the following interval of waiting until generation of the ECM compare error is reflected in the ECMESSTR2 register.
   (3 × ECMCLK) + (5 × PCLKD)
   In this product, the waiting time is about 12.6 µs because ECMCLK = 240 kHz and PCLKD = 75 MHz.
6. Use the ECMCLSSE228 bit in the ECMESSTC2 register to “clear” the ECM comparison error.
7. As required, restore the settings of the ECMEMK0, ECMEMK1, and ECMEMK2 registers which were changed in step 2 above.
8. As required, restore the settings of the ECMMICFG2, ECMNMICFG2, and ECMIRCFG2 registers which were changed in step 3 above.

32.2.3 ECM Master/Checker Error Source Status Register 0 (ECMmESSTR0 (m = M or C))

The ECMmESSTR0 (m = M or C) register is a flag register that indicates whether individual error sources occurred. This register is cleared with the corresponding bits in the ECM error source status clear trigger register 0 (ECMESSTC0) set, or with the RES# pin reset. This register is not cleared by occurrence of other reset sources.

Address(es): ECMMESSTR0: A007 D008h
ECMCESSTR0: A007 D048h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	ECMm SSE022	ECMm SSE021	ECMm SSE020	ECMm SSE019	ECMm SSE018	ECMm SSE017	ECMm SSE016
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMm SSE013	ECMm SSE012	ECMm SSE011	ECMm SSE010	ECMm SSE009	ECMm SSE008	ECMm SSE007	ECMm SSE006	ECMm SSE005	ECMm SSE004	—	ECMm SSE002	—	ECMm SSE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMmSSE000	Error Source Status 1	Indicates occurrence of a WDTA underflow/refresh error (for Cortex-R4) (error source 1). 0: Error not occurred 1: Error occurred	R
b1	—	Reserved	This bit is read as 0.	R
b2	ECMmSSE002	Error Source Status 3	Indicates occurrence of an IWDTa underflow/refresh error (error source 3). 0: Error not occurred 1: Error occurred	R
b3	—	Reserved	This bit is read as 0.	R
b4	ECMmSSE004	Error Source Status 5	Indicates occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Error not occurred 1: Error occurred	R
b5	ECMmSSE005	Error Source Status 6	Indicates occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Error not occurred 1: Error occurred	R
b6	ECMmSSE006	Error Source Status 7	Indicates occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Error not occurred 1: Error occurred	R
b7	ECMmSSE007	Error Source Status 8	Indicates occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Error not occurred 1: Error occurred	R
b8	ECMmSSE008	Error Source Status 9	Indicates occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9) 0: Error not occurred 1: Error occurred	R
b9	ECMmSSE009	Error Source Status 10	Indicates occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Error not occurred 1: Error occurred	R

Bit	Symbol	Bit Name	Description	R/W
b10	ECMmSSE010	Error Source Status 11	Indicates occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: Error not occurred 1: Error occurred	R
b11	ECMmSSE011	Error Source Status 12	Indicates occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: Error not occurred 1: Error occurred	R
b12	ECMmSSE012	Error Source Status 13	Indicates occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: Error not occurred 1: Error occurred	R
b13	ECMmSSE013	Error Source Status 14	Indicates occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: Error not occurred 1: Error occurred	R
b15, b14	—	Reserved	These bits are read as 0.	R
b16	ECMmSSE016	Error Source Status 17	Indicates occurrence of a 1-bit ECC error in the RSCAN RAM (error source 17). 0: Error not occurred 1: Error occurred	R
b17	ECMmSSE017	Error Source Status 18	Indicates occurrence of a 2-bit ECC error in the RSCAN RAM (error source 18). 0: Error not occurred 1: Error occurred	R
b18	ECMmSSE018	Error Source Status 19	Indicates occurrence of an RSCAN overflow error (error source 19). 0: Error not occurred 1: Error occurred	R
b19	ECMmSSE019	Error Source Status 20	Indicates occurrence of an error of main clock oscillation stop detection (error source 20). 0: Error not occurred 1: Error occurred	R
b20	ECMmSSE020	Error Source Status 21	Indicates occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Error not occurred 1: Error occurred	R
b21	ECMmSSE021	Error Source Status 22	Indicates occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Error not occurred 1: Error occurred	R
b22	ECMmSSE022	Error Source Status 23	Indicates occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Error not occurred 1: Error occurred	R
b31 to b23	—	Reserved	These bits are read as 0.	R

32.2.4 ECM Master/Checker Error Source Status Register 1 (ECMmESSTR1 (m = M or C))

The ECMmESSTR1 (m = M or C) register is a flag register that indicates whether individual error sources occurred. This register is cleared with the corresponding bits in the ECM error source status clear trigger register 1 (ECMESSTC1) set, or with the RES# pin reset. This register is not cleared by occurrence of other reset sources.

Address(es): ECMMESSTR1: A007 D00Ch
ECMCESSTR1: A007 D04Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMm SSE108	ECMm SSE107	ECMm SSE106	ECMm SSE105	ECMm SSE104	ECMm SSE103	ECMm SSE102	—	ECMm SSE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMmSSE100	Error Source Status 33	Indicates occurrence of a bus error (error source 33). 0: Error not occurred 1: Error occurred	R
b1	—	Reserved	This bit is read as 0.	R
b2	ECMmSSE102	Error Source Status 35	Indicates occurrence of extended pseudo error 35 (error source 35)*1. 0: Error not occurred 1: Error occurred	R
b3	ECMmSSE103	Error Source Status 36	Indicates occurrence of extended pseudo error 36 (error source 36)*1. 0: Error not occurred 1: Error occurred	R
b4	ECMmSSE104	Error Source Status 37	Indicates occurrence of extended pseudo error 37 (error source 37)*1. 0: Error not occurred 1: Error occurred	R
b5	ECMmSSE105	Error Source Status 38	Indicates occurrence of extended pseudo error 38 (error source 38)*1. 0: Error not occurred 1: Error occurred	R
b6	ECMmSSE106	Error Source Status 39	Indicates occurrence of extended pseudo error 39 (error source 39)*1. 0: Error not occurred 1: Error occurred	R
b7	ECMmSSE107	Error Source Status 40	Indicates occurrence of extended pseudo error 40 (error source 40)*1. 0: Error not occurred 1: Error occurred	R
b8	ECMmSSE108	Error Source Status 41	Indicates occurrence of extended pseudo error 41 (error source 41)*1. 0: Error not occurred 1: Error occurred	R
b31 to b9	—	Reserved	These bits are read as 0.	R

Note 1. An error occurs when the ECMPE1.ECMPE102 to ECMPE1.ECMPE108 bit is set by software.

32.2.5 ECM Master/Checker Error Source Status Register 2 (ECMmESSTR2 (m = M or C))

The ECMmESSTR2 (m = M or C) register is a flag register that indicates whether individual error sources occurred. Bits 30 to 28 of this register are cleared with the corresponding bits in the ECM error source status clear trigger register 2 (ECMESSTC2) set, or with the RES# pin reset. This register is not cleared by occurrence of other reset sources. Note that bit 31 of this register is not reset.

Address(es): ECMMESSTR2: A007 D010h
ECMCESSTR2: A007 D050h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	ECMm SSE231	ECMm SSE230	ECMm SSE229	ECMm SSE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0.	R
b28	ECMmSSE228	Error Source Status 93	Indicates occurrence of an ECM compare error (error source 93). 0: ECM compare error not occurred 1: ECM compare error occurred	R
b29	ECMmSSE229	Error Source Status 94	Indicates occurrence of a delay timer overflow (error source 94). 0: Delay timer overflow not occurred 1: Delay timer overflow occurred	R
b30	ECMmSSE230	Error Source Status 95	Indicates the status written in the ECMmESET register (error source 95). 0: Error not occurred 1: An error was set by the ECMmEST bit in the ECMmESET register.	R
b31	ECMmSSE231	Error Source Status 96	Indicates occurrence of a loopback error (error source 96). This bit is not initialized by reset. 0: Error output (master/checker) is the low level. 1: Error output (master/checker) is the high level.	R

32.2.6 ECM Master/Checker Protection Command Register (ECMmPCMD0 (m = M or C))

The ECMmPCMD0 (m = M or C) register controls writing to the protected registers. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): ECMMPCMD0: A007 D014h
ECMCPCMD0: A007 D054h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	Undefined															

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ECMm0REG[7:0]							
Value after reset:	Undefined															

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ECMm0REG7 to ECMm0REG0	Specific Instruction Sequence Write	Writes the specific instruction sequence to enable writing to the ECMm register (m = M or C).	W
b31 to b8	—	Reserved	The write value should be 0.	W

32.2.7 ECM Error Pulse Configuration Register (ECMEPCFG)

The ECMEPCFG register controls output to the ERROROUT# pin. For details, see section 32.3.1, Operations for Error Output.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D080h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMSL 0
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMSL0	ERROROUT# Pin Operation Set	Sets the operation on the ERROROUT# pin for outputting an error. 0: Non-dynamic mode 1: Dynamic mode	R/W
b7 to b1	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.8 ECM Maskable Interrupt Configuration Register 0 (ECMMICFG0)

The ECMMICFG0 register controls generation of ECM maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D084h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMMIE013	ECMMIE012	ECMMIE011	ECMMIE010	ECMMIE009	ECMMIE008	ECMMIE007	ECMMIE006	ECMMIE005	ECMMIE004	—	ECMMIE002	—	ECMMIE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMMIE000	ECM Maskable Interrupt Generation Control 1	Enables or disables a maskable interrupt due to occurrence of a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMMIE002	ECM Maskable Interrupt Generation Control 3	Enables or disables a maskable interrupt due to occurrence of an IWDTa overflow/refresh error (error source 3). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMMIE004	ECM Maskable Interrupt Generation Control 5	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b5	ECMMIE005	ECM Maskable Interrupt Generation Control 6	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b6	ECMMIE006	ECM Maskable Interrupt Generation Control 7	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b7	ECMMIE007	ECM Maskable Interrupt Generation Control 8	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMMIE008	ECM Maskable Interrupt Generation Control 9	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b9	ECMMIE009	ECM Maskable Interrupt Generation Control 10	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b10	ECMMIE010	ECM Maskable Interrupt Generation Control 11	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b11	ECMMIE011	ECM Maskable Interrupt Generation Control 12	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b12	ECMMIE012	ECM Maskable Interrupt Generation Control 13	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b13	ECMMIE013	ECM Maskable Interrupt Generation Control 14	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	ECMMIE016	ECM Maskable Interrupt Generation Control 17	Enables or disables a maskable interrupt due to occurrence of a 1-bit ECC error in the RSCAN RAM (error source 17). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b17	ECMMIE017	ECM Maskable Interrupt Generation Control 18	Enables or disables a maskable interrupt due to occurrence of a 2-bit ECC error in the RSCAN RAM (error source 18). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b18	ECMMIE018	ECM Maskable Interrupt Generation Control 19	Enables or disables a maskable interrupt due to occurrence of an RSCAN overflow error (error source 19). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b19	ECMMIE019	ECM Maskable Interrupt Generation Control 20	Enables or disables a maskable interrupt due to occurrence of an error of main clock oscillation stop detection (error source 20). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b20	ECMMIE020	ECM Maskable Interrupt Generation Control 21	Enables or disables a maskable interrupt due to occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b21	ECMMIE021	ECM Maskable Interrupt Generation Control 22	Enables or disables a maskable interrupt due to occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b22	ECMMIE022	ECM Maskable Interrupt Generation Control 23	Enables or disables a maskable interrupt due to occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b23	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.9 ECM Maskable Interrupt Configuration Register 1 (ECMMICFG1)

The ECMMICFG1 register controls generation of ECM maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D088h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMMIE108	ECMMIE107	ECMMIE106	ECMMIE105	ECMMIE104	ECMMIE103	ECMMIE102	—	ECMMIE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMMIE100	ECM Maskable Interrupt Generation Control 33	Enables or disables a maskable interrupt due to occurrence of a bus error (error source 33). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMMIE102	ECM Maskable Interrupt Generation Control 35	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 35 (error source 35). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b3	ECMMIE103	ECM Maskable Interrupt Generation Control 36	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 36 (error source 36). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b4	ECMMIE104	ECM Maskable Interrupt Generation Control 37	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 37 (error source 37). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b5	ECMMIE105	ECM Maskable Interrupt Generation Control 38	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 38 (error source 38). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b6	ECMMIE106	ECM Maskable Interrupt Generation Control 39	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 39 (error source 39). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b7	ECMMIE107	ECM Maskable Interrupt Generation Control 40	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 40 (error source 40). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b8	ECMMIE108	ECM Maskable Interrupt Generation Control 41	Enables or disables a maskable interrupt due to occurrence of extended pseudo error 41 (error source 41). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.10 ECM Maskable Interrupt Configuration Register 2 (ECMMICFG2)

The ECMMICFG2 register controls generation of ECM maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D08Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	ECMMIE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMMIE228	ECM Maskable Interrupt Generation Control 93	Enables or disables a maskable interrupt due to occurrence of an ECM compare error (error source 93). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.11 ECM Non-maskable Interrupt Configuration Register 0 (ECMNMICFG0)

The ECMNMICFG0 register controls generation of ECM non-maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D090h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMNMIE013	ECMNMIE012	ECMNMIE011	ECMNMIE010	ECMNMIE009	ECMNMIE008	ECMNMIE007	ECMNMIE006	ECMNMIE005	ECMNMIE004	—	ECMNMIE002	—	ECMNMIE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMNMIE000	ECM Non-maskable Interrupt Generation Control 1	Enables or disables a non-maskable interrupt due to occurrence of a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMNMIE002	ECM Non-maskable Interrupt Generation Control 3	Enables or disables a non-maskable interrupt due to occurrence of an IWDTa overflow/refresh error (error source 3). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMNMIE004	ECM Non-maskable Interrupt Generation Control 5	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error (with correction) and 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b5	ECMNMIE005	ECM Non-maskable Interrupt Generation Control 6	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error (with correction) and 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b6	ECMNMIE006	ECM Non-maskable Interrupt Generation Control 7	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b7	ECMNMIE007	ECM Non-maskable Interrupt Generation Control 8	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMNMIE008	ECM Non-maskable Interrupt Generation Control 9	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b9	ECMNMIE009	ECM Non-maskable Interrupt Generation Control 10	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b10	ECMNMIE010	ECM Non-maskable Interrupt Generation Control 11	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b11	ECMNMIE011	ECM Non-maskable Interrupt Generation Control 12	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b12	ECMNMIE012	ECM Non-maskable Interrupt Generation Control 13	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b13	ECMNMIE013	ECM Non-maskable Interrupt Generation Control 14	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	ECMNMIE016	ECM Non-maskable Interrupt Generation Control 17	Enables or disables a non-maskable interrupt due to occurrence of a 1-bit ECC error in the RSCAN RAM (error source 17). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b17	ECMNMIE017	ECM Non-maskable Interrupt Generation Control 18	Enables or disables a non-maskable interrupt due to occurrence of a 2-bit ECC error in the RSCAN RAM (error source 18). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b18	ECMNMIE018	ECM Non-maskable Interrupt Generation Control 19	Enables or disables a non-maskable interrupt due to occurrence of an RSCAN overflow error (error source 19). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b19	ECMNMIE019	ECM Non-maskable Interrupt Generation Control 20	Enables or disables a non-maskable interrupt due to occurrence of an error of main clock oscillation stop detection (error source 20). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b20	ECMNMIE020	ECM Non-maskable Interrupt Generation Control 21	Enables or disables a non-maskable interrupt due to occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b21	ECMNMIE021	ECM Non-maskable Interrupt Generation Control 22	Enables or disables a non-maskable interrupt due to occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b22	ECMNMIE022	ECM Non-maskable Interrupt Generation Control 23	Enables or disables a non-maskable interrupt due to occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b23	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.12 ECM Non-maskable Interrupt Configuration Register 1 (ECMNMICFG1)

The ECMNMICFG1 register controls generation of ECM non-maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D094h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMNMIE108	ECMNMIE107	ECMNMIE106	ECMNMIE105	ECMNMIE104	ECMNMIE103	ECMNMIE102	—	ECMNMIE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMNMIE100	ECM Non-maskable Interrupt Generation Control 33	Enables or disables a non-maskable interrupt due to occurrence of a bus error (error source 33). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMNMIE102	ECM Non-maskable Interrupt Generation Control 35	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 35 (error source 35). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b3	ECMNMIE103	ECM Non-maskable Interrupt Generation Control 36	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 36 (error source 36). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b4	ECMNMIE104	ECM Non-maskable Interrupt Generation Control 37	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 37 (error source 37). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b5	ECMNMIE105	ECM Non-maskable Interrupt Generation Control 38	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 38 (error source 38). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b6	ECMNMIE106	ECM Non-maskable Interrupt Generation Control 39	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 39 (error source 39). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b7	ECMNMIE107	ECM Non-maskable Interrupt Generation Control 40	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 40 (error source 40). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b8	ECMNMIE108	ECM Non-maskable Interrupt Generation Control 41	Enables or disables a non-maskable interrupt due to occurrence of extended pseudo error 41 (error source 41). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.13 ECM Non-maskable Interrupt Configuration Register 2 (ECMNMICFG2)

The ECMNMICFG2 register controls generation of ECM non-maskable interrupts due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D098h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	ECMNMIE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMNMIE228	ECM Non-maskable Interrupt Generation Control 93	Enables or disables a non-maskable interrupt due to occurrence of an ECM compare error (error source 93). 0: Interrupt generation disabled 1: Interrupt generation enabled	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.14 ECM Internal Reset Configuration Register 0 (ECMIRCFG0)

The ECMIRCFG0 register controls generation of internal resets (ECM resets) due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D09Ch

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMIRE013	ECMIRE012	ECMIRE011	ECMIRE010	ECMIRE009	ECMIRE008	ECMIRE007	ECMIRE006	ECMIRE005	ECMIRE004	—	ECMIRE002	—	ECMIRE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Bit	Symbol	Bit Name	Description	R/W
b0	ECMIRE000	ECM Internal Reset Generation Control 1	Enables or disables generation of an ECM reset due to occurrence of a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMIRE002	ECM Internal Reset Generation Control 3	Enables or disables generation of an ECM reset due to occurrence of an IWDTa overflow/refresh error (error source 3). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMIRE004	ECM Internal Reset Generation Control 5	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b5	ECMIRE005	ECM Internal Reset Generation Control 6	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b6	ECMIRE006	ECM Internal Reset Generation Control 7	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b7	ECMIRE007	ECM Internal Reset Generation Control 8	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMIRE008	ECM Internal Reset Generation Control 9	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b9	ECMIRE009	ECM Internal Reset Generation Control 10	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b10	ECMIRE010	ECM Internal Reset Generation Control 11	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b11	ECMIRE011	ECM Internal Reset Generation Control 12	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b12	ECMIRE012	ECM Internal Reset Generation Control 13	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b13	ECMIRE013	ECM Internal Reset Generation Control 14	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	ECMIRE016	ECM Internal Reset Generation Control 17	Enables or disables generation of an ECM reset due to occurrence of a 1-bit ECC error in the RSCAN RAM (error source 17). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b17	ECMIRE017	ECM Internal Reset Generation Control 18	Enables or disables generation of an ECM reset due to occurrence of a 2-bit ECC error in the RSCAN RAM (error source 18). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b18	ECMIRE018	ECM Internal Reset Generation Control 19	Enables or disables generation of an ECM reset due to occurrence of an RSCAN overflow error (error source 19). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b19	ECMIRE019	ECM Internal Reset Generation Control 20	Enables or disables generation of an ECM reset due to occurrence of an error of main clock oscillation stop detection (error source 20). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b20	ECMIRE020	ECM Internal Reset Generation Control 21	Enables or disables generation of an ECM reset due to occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b21	ECMIRE021	ECM Internal Reset Generation Control 22	Enables or disables generation of an ECM reset due to occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b22	ECMIRE022	ECM Internal Reset Generation Control 23	Enables or disables generation of an ECM reset due to occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b31 to b23	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.15 ECM Internal Reset Configuration Register 1 (ECMIRCFG1)

The ECMIRCFG1 register controls generation of internal resets (ECM resets) due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0A0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMIRE108	ECMIRE107	ECMIRE106	ECMIRE105	ECMIRE104	ECMIRE103	ECMIRE102	—	ECMIRE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMIRE100	ECM Internal Reset Generation Control 33	Enables or disables generation of an ECM reset due to occurrence of a bus error (error source 33). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMIRE102	ECM Internal Reset Generation Control 35	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 35 (error source 35). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b3	ECMIRE103	ECM Internal Reset Generation Control 36	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 36 (error source 36). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b4	ECMIRE104	ECM Internal Reset Generation Control 37	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 37 (error source 37). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b5	ECMIRE105	ECM Internal Reset Generation Control 38	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 38 (error source 38). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b6	ECMIRE106	ECM Internal Reset Generation Control 39	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 39 (error source 39). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b7	ECMIRE107	ECM Internal Reset Generation Control 40	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 40 (error source 40). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b8	ECMIRE108	ECM Internal Reset Generation Control 41	Enables or disables generation of an ECM reset due to occurrence of extended pseudo error 41 (error source 41). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.16 ECM Internal Reset Configuration Register 2 (ECMIRCFG2)

The ECMIRCFG2 register controls generation of internal resets (ECM resets) due to occurrence of individual error sources.

This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0A4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ECMIRE229	ECMIRE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMIRE228	ECM Internal Reset Generation Control 93	Enables or disables generation of an ECM reset due to occurrence of an ECM compare error (error source 93). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b29	ECMIRE229	ECM Internal Reset Generation Control 94	Enables or disables generation of an ECM reset due to occurrence of a delay timer overflow (error source 94). 0: ECM reset generation disabled 1: ECM reset generation enabled	R/W
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.17 ECM Error Mask Register 0 (ECMEMK0)

The ECMEMK0 register controls mask settings for error output signals due to occurrence of individual error sources. If an unmasked error output signal is generated, the output of the ERROROUT# pin is set to the active level (low level). This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0A8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMEMK013	ECMEMK012	ECMEMK011	ECMEMK010	ECMEMK009	ECMEMK008	ECMEMK007	ECMEMK006	ECMEMK005	ECMEMK004	—	ECMEMK002	—	ECMEMK000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMEMK000	ECM Error Output Signal Mask Control 1	Controls whether to mask an error output signal due to occurrence of a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Error signal output not masked 1: Error signal output masked	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMEMK002	ECM Error Output Signal Mask Control 3	Controls whether to mask an error output signal due to occurrence of an IWDTa overflow/refresh error (error source 3). 0: Error signal output not masked 1: Error signal output masked	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMEMK004	ECM Error Output Signal Mask Control 5	Controls whether to mask an error output signal due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Error signal output not masked 1: Error signal output masked	R/W
b5	ECMEMK005	ECM Error Output Signal Mask Control 6	Controls whether to mask an error output signal due to occurrence of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Error signal output not masked 1: Error signal output masked	R/W
b6	ECMEMK006	ECM Error Output Signal Mask Control 7	Controls whether to mask an error output signal due to occurrence of a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Error signal output not masked 1: Error signal output masked	R/W
b7	ECMEMK007	ECM Error Output Signal Mask Control 8	Controls whether to mask an error output signal due to occurrence of a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Error signal output not masked 1: Error signal output masked	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMEMK008	ECM Error Output Signal Mask Control 9	Controls whether to mask an error output signal due to occurrence of a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Error signal output not masked 1: Error signal output masked	R/W
b9	ECMEMK009	ECM Error Output Signal Mask Control 10	Controls whether to mask an error output signal due to occurrence of a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Error signal output not masked 1: Error signal output masked	R/W
b10	ECMEMK010	ECM Error Output Signal Mask Control 11	Controls whether to mask an error output signal due to occurrence of a 1-bit ECC error in the ATCM (error source 11). 0: Error signal output not masked 1: Error signal output masked	R/W
b11	ECMEMK011	ECM Error Output Signal Mask Control 12	Controls whether to mask an error output signal due to occurrence of a 2-bit ECC error in the ATCM (error source 12). 0: Error signal output not masked 1: Error signal output masked	R/W
b12	ECMEMK012	ECM Error Output Signal Mask Control 13	Controls whether to mask an error output signal due to occurrence of a 1-bit ECC error in the BTCM (error source 13). 0: Error signal output not masked 1: Error signal output masked	R/W
b13	ECMEMK013	ECM Error Output Signal Mask Control 14	Controls whether to mask an error output signal due to occurrence of a 2-bit ECC error in the BTCM (error source 14). 0: Error signal output not masked 1: Error signal output masked	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	ECMEMK016	ECM Error Output Signal Mask Control 17	Controls whether to mask an error output signal due to occurrence of a 1-bit ECC error in the RSCAN RAM (error source 17). 0: Error signal output not masked 1: Error signal output masked	R/W
b17	ECMEMK017	ECM Error Output Signal Mask Control 18	Controls whether to mask an error output signal due to occurrence of a 2-bit ECC error in the RSCAN RAM (error source 18). 0: Error signal output not masked 1: Error signal output masked	R/W
b18	ECMEMK018	ECM Error Output Signal Mask Control 19	Controls whether to mask an error output signal due to occurrence of an RSCAN overflow error (error source 19). 0: Error signal output not masked 1: Error signal output masked	R/W
b19	ECMEMK019	ECM Error Output Signal Mask Control 20	Controls whether to mask an error output signal due to occurrence of an error of main clock oscillation stop detection (error source 20). 0: Error signal output not masked 1: Error signal output masked	R/W
b20	ECMEMK020	ECM Error Output Signal Mask Control 21	Controls whether to mask an error output signal due to occurrence of an error of CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Error signal output not masked 1: Error signal output masked	R/W
b21	ECMEMK021	ECM Error Output Signal Mask Control 22	Controls whether to mask an error output signal due to occurrence of an error of CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Error signal output not masked 1: Error signal output masked	R/W

Bit	Symbol	Bit Name	Description	R/W
b22	ECMEMK022	ECM Error Output Signal Mask Control 23	Controls whether to mask an error output signal due to occurrence of an error of CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Error signal output not masked 1: Error signal output masked	R/W
b31 to b23	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.18 ECM Error Mask Register 1 (ECMEMK1)

The ECMEMK1 register controls mask settings for error output signals due to occurrence of individual error sources. If an unmasked error output signal is generated, the output of the ERROROUT# pin is set to the active level (low level). This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0ACh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMEMK 108	ECMEMK 107	ECMEMK 106	ECMEMK 105	ECMEMK 104	ECMEMK 103	ECMEMK 102	—	ECMEMK 100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMEMK100	ECM Error Output Signal Mask Control 33	Controls whether to mask an error output signal due to occurrence of a bus error (error source 33). 0: Error signal output not masked 1: Error signal output masked	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMEMK102	ECM Error Output Signal Mask Control 35	Controls whether to mask an error output signal due to occurrence of extended pseudo error 35 (error source 35). 0: Error signal output not masked 1: Error signal output masked	R/W
b3	ECMEMK103	ECM Error Output Signal Mask Control 36	Controls whether to mask an error output signal due to occurrence of extended pseudo error 36 (error source 36). 0: Error signal output not masked 1: Error signal output masked	R/W
b4	ECMEMK104	ECM Error Output Signal Mask Control 37	Controls whether to mask an error output signal due to occurrence of extended pseudo error 37 (error source 37). 0: Error signal output not masked 1: Error signal output masked	R/W
b5	ECMEMK105	ECM Error Output Signal Mask Control 38	Controls whether to mask an error output signal due to occurrence of extended pseudo error 38 (error source 38). 0: Error signal output not masked 1: Error signal output masked	R/W
b6	ECMEMK106	ECM Error Output Signal Mask Control 39	Controls whether to mask an error output signal due to occurrence of extended pseudo error 39 (error source 39). 0: Error signal output not masked 1: Error signal output masked	R/W
b7	ECMEMK107	ECM Error Output Signal Mask Control 40	Controls whether to mask an error output signal due to occurrence of extended pseudo error 40 (error source 40). 0: Error signal output not masked 1: Error signal output masked	R/W
b8	ECMEMK108	ECM Error Output Signal Mask Control 41	Controls whether to mask an error output signal due to occurrence of extended pseudo error 41 (error source 41). 0: Error signal output not masked 1: Error signal output masked	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.19 ECM Error Mask Register 2 (ECMEMK2)

The ECMEMK2 register controls mask settings for error output signals due to occurrence of individual error sources. If an unmasked error output signal is generated, the output of the ERROROUT# pin is set to the active level (low level). This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.

Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0B0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ECMEMK229	ECMEMK228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0.	R
b28	ECMEMK228	ECM Internal Reset Generation Control 93	ECM Error Output Signal Mask Control Controls whether to mask an error output signal due to occurrence of an ECM compare error (error source 93). 0: Error signal output not masked 1: Error signal output masked	R/W
b29	ECMEMK229	ECM Internal Reset Generation Control 94	ECM Error Output Signal Mask Control Controls whether to mask an error output signal due to occurrence of a delay timer overflow (error source 94). 0: Error signal output not masked 1: Error signal output masked	R/W
b31, b30	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.20 ECM Error Source Status Clear Trigger Register 0 (ECMESSTC0)

The ECMESSTC0 register controls clearing of individual error statuses. Setting individual bits in this register to 1 can clear the corresponding error statuses held in the ECMmESSTR0 (m = M or C) register.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0B4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMCLSSE013	ECMCLSSE012	ECMCLSSE011	ECMCLSSE010	ECMCLSSE009	ECMCLSSE008	ECMCLSSE007	ECMCLSSE006	ECMCLSSE005	ECMCLSSE004	—	ECMCLSSE002	—	ECMCLSSE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMCLSSE000	ECM Error Status Clear 1	Clears the error status of WDT overflow/refresh error (for Cortex-R4) (error source 1) and the ECMmESSTR0.ECMmSSE000 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b1	—	Reserved	The write value should be 0.	W
b2	ECMCLSSE002	ECM Error Status Clear 3	Clears the error status of IWDTa overflow/refresh error (error source 3) and the ECMmESSTR0.ECMmSSE002 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b3	—	Reserved	The write value should be 0.	W
b4	ECMCLSSE004	ECM Error Status Clear 5	Clears the error status of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5) and the ECMmESSTR0.ECMmSSE004 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b5	ECMCLSSE005	ECM Error Status Clear 6	Clears the error status of a 1-bit ECC error (with correction) or a 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6) and the ECMmESSTR0.ECMmSSE005 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b6	ECMCLSSE006	ECM Error Status Clear 7	Clears the error status of 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7) and the ECMmESSTR0.ECMmSSE006 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b7	ECMCLSSE007	ECM Error Status Clear 8	Clears the error status of 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8) and the ECMmESSTR0.ECMmSSE007 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMCLSSE008	ECM Error Status Clear 9	Clears the error status of 1-bit ECC error in the data cache (Data RAM) (error source 9) and the ECMmESSTR0.ECMmSSE008 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b9	ECMCLSSE009	ECM Error Status Clear 10	Clears the error status of 2-bit ECC error in the data cache (Data RAM) (error source 10) and the ECMmESSTR0.ECMmSSE009 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b10	ECMCLSSE010	ECM Error Status Clear 11	Clears the error status of 1-bit ECC error in the ATCM (error source 11) and the ECMmESSTR0.ECMmSSE010 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b11	ECMCLSSE011	ECM Error Status Clear 12	Clears the error status of 2-bit ECC error in the ATCM (error source 12) and the ECMmESSTR0.ECMmSSE011 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b12	ECMCLSSE012	ECM Error Status Clear 13	Clears the error status of 1-bit ECC error in the BTCM (error source 13) and the ECMmESSTR0.ECMmSSE012 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b13	ECMCLSSE013	ECM Error Status Clear 14	Clears the error status of 2-bit ECC error in the BTCM (error source 14) and the ECMmESSTR0.ECMmSSE013 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b15, b14	—	Reserved	The write value should be 0.	W
b16	ECMCLSSE016	ECM Error Status Clear 17	Clears the error status of 1-bit ECC error in the RSCAN RAM (error source 17) and the ECMmESSTR0.ECMmSSE016 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b17	ECMCLSSE017	ECM Error Status Clear 18	Clears the error status of 2-bit ECC error in the RSCAN RAM (error source 18) and the ECMmESSTR0.ECMmSSE017 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b18	ECMCLSSE018	ECM Error Status Clear 19	Clears the error status of RSCAN overflow error (error source 19) and the ECMmESSTR0.ECMmSSE018 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b19	ECMCLSSE019	ECM Error Status Clear 20	Clears the error status of main clock oscillation stop detection (error source 20) and the ECMmESSTR0.ECMmSSE019 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b20	ECMCLSSE020	ECM Error Status Clear 21	Clears the error status of CLMA0 oscillation stop detection (PLL0) (error source 21) and the ECMmESSTR0.ECMmSSE020 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b21	ECMCLSSE021	ECM Error Status Clear 22	Clears the error status of CLMA1 oscillation stop detection (PLL1) (error source 22) and the ECMmESSTR0.ECMmSSE021 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W

Bit	Symbol	Bit Name	Description	R/W
b22	ECMCLSSE022	ECM Error Status Clear 23	Clears the error status of CLMA2 oscillation stop detection (LOCO) (error source 23) and the ECMmESSTR0.ECMmSSE022 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b31 to b23	—	Reserved	The write value should be 0.	W

32.2.21 ECM Error Source Status Clear Trigger Register 1 (ECMESSTC1)

The ECMESSTC1 register controls clearing of individual error statuses. Setting individual bits in this register to 1 can clear the corresponding error statuses held in the ECMmESSTR1 (m = M or C) register.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0B8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMCLSSE108	ECMCLSSE107	ECMCLSSE106	ECMCLSSE105	ECMCLSSE104	ECMCLSSE103	ECMCLSSE102	—	ECMCLSSE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMCLSSE100	ECM Error Status Clear 33	Clears the error status of bus error (error source 33) and the ECMmESSTR1.ECMmSSE100 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b1	—	Reserved	The write value should be 0.	W
b2	ECMCLSSE102	ECM Error Status Clear 35	Clears the error status of extended pseudo error 35 (error source 35) and the ECMmESSTR1.ECMmSSE102 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b3	ECMCLSSE103	ECM Error Status Clear 36	Clears the error status of extended pseudo error 36 (error source 36) and the ECMmESSTR1.ECMmSSE103 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b4	ECMCLSSE104	ECM Error Status Clear 37	Clears the error status of extended pseudo error 37 (error source 37) and the ECMmESSTR1.ECMmSSE104 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b5	ECMCLSSE105	ECM Error Status Clear 38	Clears the error status of extended pseudo error 38 (error source 38) and the ECMmESSTR1.ECMmSSE105 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b6	ECMCLSSE106	ECM Error Status Clear 39	Clears the error status of extended pseudo error 39 (error source 39) and the ECMmESSTR1.ECMmSSE106 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b7	ECMCLSSE107	ECM Error Status Clear 40	Clears the error status of extended pseudo error 40 (error source 40) and the ECMmESSTR1.ECMmSSE107 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b8	ECMCLSSE108	ECM Error Status Clear 41	Clears the error status of extended pseudo error 41 (error source 41) and the ECMmESSTR1.ECMmSSE108 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b31 to b9	—	Reserved	The write value should be 0.	W

32.2.22 ECM Error Source Status Clear Trigger Register 2 (ECMESSTC2)

The ECMESSTC2 register controls clearing of individual error statuses. Setting individual bits in this register to 1 can clear the corresponding error statuses held in the ECMmESSTR2 (m = M or C) register.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0BCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	ECMCLSSE230	ECMCLSSE229	ECMCLSSE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	The write value should be 0.	W
b28	ECMCLSSE228	ECM Error Status Clear 93	Clears the error status of ECM compare error (error source 93) and the ECMmESSTR2.ECMmSSE228 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b29	ECMCLSSE229	ECM Error Status Clear 94	Clears the error status of a delay timer overflow (error source 94) and the ECMmESSTR2.ECMmSSE229 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b30	ECMCLSSE230	ECM Error Status Clear 95	Clears the error status due to writing to ECMmESET (error source 95) and the ECMmESSTR2.ECMmSSE230 bit. 0: Corresponding error status unchanged 1: Corresponding error status cleared	W
b31	—	Reserved	The write value should be 0.	W

32.2.23 ECM Protection Command Register (ECMPCMD1)

The ECMPCMD1 register controls writing to protected common registers. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0C0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	Undefined															

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	ECM1REG[7:0]							
Value after reset:	Undefined															

Bit	Symbol	Bit Name	Description	R/W
b7 to b0	ECM1REG[7:0]	Specific Instruction Sequence Write	Write the specific instruction sequence to enable writing to common registers.	W
b31 to b8	—	Reserved	The write value should be 0.	W

32.2.24 ECM Protection Status Register (ECMPS)

The ECMPS register indicates the status of whether writing to the protected register has been correctly done. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0C4h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMPRERR
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMPRERR	ECM Protection Status	Indicates whether writing to the write protected register has been correctly done. 0: Writing was successful. 1: Writing failed.	R
b7 to b1	—	Reserved	These bits are read as 0.	R

32.2.25 ECM Pseudo Error Trigger Register 0 (ECMPE0)

The ECMPE0 register is a control register that issues a pseudo error for self-diagnosis. When a pseudo error is issued, an interrupt set by ECM or an ECM reset occurs in the same way as when an actual error source occurs.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0C8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMPE013	ECMPE012	ECMPE011	ECMPE010	ECMPE009	ECMPE008	ECMPE007	ECMPE006	ECMPE005	ECMPE004	—	ECMPE002	—	ECMPE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMPE000	ECM Pseudo Error Trigger 1	Generates a pseudo WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b1	—	Reserved	The write value should be 0.	W
b2	ECMPE002	ECM Pseudo Error Trigger 3	Generates a pseudo IWDTa overflow/refresh error (error source 3). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b3	—	Reserved	The write value should be 0.	W
b4	ECMPE004	ECM Pseudo Error Trigger 5	Generates a pseudo 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b5	ECMPE005	ECM Pseudo Error Trigger 6	Generates a pseudo 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b6	ECMPE006	ECM Pseudo Error Trigger 7	Generates a pseudo 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b7	ECMPE007	ECM Pseudo Error Trigger 8	Generates a pseudo 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b8	ECMPE008	ECM Pseudo Error Trigger 9	Generates a pseudo 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b9	ECMPE009	ECM Pseudo Error Trigger 10	Generates a pseudo 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W

Bit	Symbol	Bit Name	Description	R/W
b10	ECMPE010	ECM Pseudo Error Trigger 11	Generates a pseudo 1-bit ECC error in the ATCM (error source 11). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b11	ECMPE011	ECM Pseudo Error Trigger 12	Generates a pseudo 2-bit ECC error in the ATCM (error source 12). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b12	ECMPE012	ECM Pseudo Error Trigger 13	Generates a pseudo 1-bit ECC error in the BTCM (error source 13). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b13	ECMPE013	ECM Pseudo Error Trigger 14	Generates a pseudo 2-bit ECC error in the BTCM (error source 14). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b15, b14	—	Reserved	The write value should be 0.	W
b16	ECMPE016	ECM Pseudo Error Trigger 17	Generates a pseudo 1-bit ECC error in the RSCAN RAM (error source 17). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b17	ECMPE017	ECM Pseudo Error Trigger 18	Generates a pseudo 2-bit ECC error in the RSCAN RAM (error source 18). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b18	ECMPE018	ECM Pseudo Error Trigger 19	Generates a pseudo RSCAN overflow error (error source 19). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b19	ECMPE019	ECM Pseudo Error Trigger 20	Generates pseudo detection of main clock oscillation stoppage (error source 20). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b20	ECMPE020	ECM Pseudo Error Trigger 21	Generates pseudo detection of CLMA0 oscillation stoppage (PLL0) (error source 21). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b21	ECMPE021	ECM Pseudo Error Trigger 22	Generates pseudo detection of CLMA1 oscillation stoppage (PLL1) (error source 22). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b22	ECMPE022	ECM Pseudo Error Trigger 23	Generates pseudo detection of CLMA2 oscillation stoppage (LOCO) (error source 23). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b31 to b23	—	Reserved	The write value should be 0.	W

32.2.26 ECM Pseudo Error Trigger Register 1 (ECMPE1)

The ECMPE1 register is a control register that issues a pseudo error for self-diagnosis. When a pseudo error is issued, an interrupt set by ECM or an ECM reset occurs in the same way as when an actual error source occurs.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.
Bits 2 to 8 are trigger bits for extended pseudo errors. For details, see section 32.3.3, Pseudo Error Generation.

Address(es): A007 D0CCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMPE108	ECMPE107	ECMPE106	ECMPE105	ECMPE104	ECMPE103	ECMPE102	—	ECMPE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMPE100	ECM Pseudo Error Trigger 33	Generates a pseudo bus error (error source 33). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b1	—	Reserved	The write value should be 0.	W
b2	ECMPE102	ECM Pseudo Error Trigger 35	Generates a pseudo “extended pseudo error 35” (error source 35). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b3	ECMPE103	ECM Pseudo Error Trigger 36	Generates a pseudo “extended pseudo error 36” (error source 36). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b4	ECMPE104	ECM Pseudo Error Trigger 37	Generates a pseudo “extended pseudo error 37” (error source 37). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b5	ECMPE105	ECM Pseudo Error Trigger 38	Generates a pseudo “extended pseudo error 38” (error source 38). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b6	ECMPE106	ECM Pseudo Error Trigger 39	Generates a pseudo “extended pseudo error 39” (error source 39). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b7	ECMPE107	ECM Pseudo Error Trigger 40	Generates a pseudo “extended pseudo error 40” (error source 40). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b8	ECMPE108	ECM Pseudo Error Trigger 41	Generates a pseudo “extended pseudo error 41” (error source 41). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b31 to b9	—	Reserved	The write value should be 0.	W

32.2.27 ECM Pseudo Error Trigger Register 2 (ECMPE2)

The ECMPE2 register is a control register that issues a pseudo error for self-diagnosis. When a pseudo error is issued, an interrupt set by ECM or an ECM reset occurs in the same way as when an actual error source occurs.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0D0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	ECMPE229	ECMPE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	The write value should be 0.	W
b28	ECMPE228	ECM Pseudo Error Trigger 93	Generates a pseudo EMC error (error source 93). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b29	ECMPE229	ECM Pseudo Error Trigger 94	Generates a pseudo delay timer overflow (error source 94). 0: Pseudo error not generated 1: Generates corresponding pseudo error	W
b31, b30	—	Reserved	The write value should be 0.	W

32.2.28 ECM Delay Timer Control Register (ECMDTMCTL)

The ECMDTMCTL register controls the delay timer.
This register is a common register. Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0D4h

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	DTMSTACNTCLK	—	—	DTMSTP	DTMSTA
Value after reset:	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	DTMSTA	Delay Timer Start	Sets the operation of the delay timer. 0: Delay timer operation disabled 1: Delay timer operation enabled	R/W
b1	DTMSTP	Delay Timer Stop	Writing 1 to this bit initializes the delay timer counter, causing the delay timer to stop. Simultaneously, the DTMSTA bit is set to 0.	W
b3, b2	—	Reserved	These bits are read as 0.	R
b4	DTMSTACNTCLK	Delay Timer Status	The value of the DTMSTA bit is applied to the operating status of the delay timer. If the DTMSTA bit is modified once, modifying the DTMSTA bit is disabled until the setting value of the DTMSTA bit is applied to the DTMSTACNTCLK bit.	R
b7 to b5	—	Reserved	These bits are read as 0.	R

32.2.29 ECM Delay Timer Register (ECMDTMR)

The ECMDTMR register is a 16-bit counter register for the delay timer. The 16-bit counter counts up using ECMCLK (240 kHz). Changing the DTMSTA bit in the ECM delay timer control register from 1 (delay timer operation enabled) to 0 (delay timer operation disabled) initializes the 16-bit counter. This register can only be read.

Address(es): A007 D0D8h

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMTDMR[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

32.2.30 ECM Delay Timer Compare Register (ECMDTMCMP)

The ECMDTMCMP register is a compare register used to set the overflow cycle of the delay timer. A delay timer overflow signal is generated to set the ECMmSSE229 bit when the value of this register matches with the value of the delay timer counter. Writing data to this register has to be conducted while the delay timer is stopped.
This register is a common register. Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.
Write this register after confirming that the read value of the CMPW bit is 0.

Address(es): A007 D0DCh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPW
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMDTMCMP[15:0]															
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	ECMDTMCMP [15:0]	Delay Timer Compare	Sets the overflow cycle of the delay timer. (overflow cycle) = (N + 1) × (ECMCLK cycle (240 kHz)) N: Set value For details, see section 32.3.5, Timeout Function for Interrupt Processing by Using the Delay Timer.	R/W
b16	CMPW	Compare Write	Indicates whether writing to the delay timer compare bit is enabled. Writing the compare value is possible when this bit is 0. 0: Writing to ECMDTMCMP[15:0] enabled 1: Writing to ECMDTMCMP[15:0] disabled	R
b31 to b17	—	Reserved	These bits are read as 0.	R

32.2.31 ECM Delay Timer Configuration Register 0 (ECMDTMCFG0)

The ECMDTMCFG0 register controls delay timer operation in response to an ECM maskable interrupt. Whether to enable delay timer operation when an ECM maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0E0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	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

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMTE013	ECMTE012	ECMTE011	ECMTE010	ECMTE009	ECMTE008	ECMTE007	ECMTE006	ECMTE005	ECMTE004	—	ECMTE002	—	ECMTE000
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMTE000	ECM Delay Timer Start Control 1	Enables delay timer operation in response to an ECM maskable interrupt caused by a WDT overflow/refresh error (for Cortex-R4) (error source 1). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMTE002	ECM Delay Timer Start Control 3	Enables delay timer operation in response to an ECM maskable interrupt caused by an IWDTa overflow/refresh error (error source 3). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMTE004	ECM Delay Timer Start Control 5	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b5	ECMTE005	ECM Delay Timer Start Control 6	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b6	ECMTE006	ECM Delay Timer Start Control 7	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b7	ECMTE007	ECM Delay Timer Start Control 8	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMTE008	ECM Delay Timer Start Control 9	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b9	ECMTE009	ECM Delay Timer Start Control 10	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b10	ECMTE010	ECM Delay Timer Start Control 11	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the ATCM (error source 11). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b11	ECMTE011	ECM Delay Timer Start Control 12	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the ATCM (error source 12). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b12	ECMTE012	ECM Delay Timer Start Control 13	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the BTCM (error source 13). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b13	ECMTE013	ECM Delay Timer Start Control 14	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the BTCM (error source 14). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	ECMTE016	ECM Delay Timer Start Control 17	Enables delay timer operation in response to an ECM maskable interrupt caused by a 1-bit ECC error in the RSCAN RAM (error source 17). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b17	ECMTE017	ECM Delay Timer Start Control 18	Enables delay timer operation in response to an ECM maskable interrupt caused by a 2-bit ECC error in the RSCAN RAM (error source 18). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b18	ECMTE018	ECM Delay Timer Start Control 19	Enables delay timer operation in response to an ECM maskable interrupt caused by an RSCAN overflow error (error source 19). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b19	ECMTE019	ECM Delay Timer Start Control 20	Enables delay timer operation in response to an ECM maskable interrupt caused by main clock oscillation stop detection (error source 20). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b20	ECMTE020	ECM Delay Timer Start Control 21	Enables delay timer operation in response to an ECM maskable interrupt caused by CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b21	ECMTE021	ECM Delay Timer Start Control 22	Enables delay timer operation in response to an ECM maskable interrupt caused by CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b22	ECMTE022	ECM Delay Timer Start Control 23	Enables delay timer operation in response to an ECM maskable interrupt caused by CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b23	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.32 ECM Delay Timer Configuration Register 1 (ECMDTMCFG1)

The ECMDTMCFG1 register controls delay timer operation in response to an ECM maskable interrupt. Whether to enable delay timer operation when an ECM maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0E4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMTE108	ECMTE107	ECMTE106	ECMTE105	ECMTE104	ECMTE103	ECMTE102	—	ECMTE100
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMTE100	ECM Delay Timer Start Control 33	Enables delay timer operation in response to an ECM maskable interrupt caused by a bus error (error source 33). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMTE102	ECM Delay Timer Start Control 35	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 35 (error source 35). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b3	ECMTE103	ECM Delay Timer Start Control 36	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 36 (error source 36). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b4	ECMTE104	ECM Delay Timer Start Control 37	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 37 (error source 37). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b5	ECMTE105	ECM Delay Timer Start Control 38	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 38 (error source 38). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b6	ECMTE106	ECM Delay Timer Start Control 39	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 39 (error source 39). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b7	ECMTE107	ECM Delay Timer Start Control 40	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 40 (error source 40). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMTE108	ECM Delay Timer Start Control 41	Enables delay timer operation in response to an ECM maskable interrupt caused by extended pseudo error 41 (error source 41). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.33 ECM Delay Timer Configuration Register 2 (ECMDTMCFG2)

The ECMDTMCFG2 register controls delay timer operation in response to an ECM maskable interrupt. Whether to enable delay timer operation when an ECM maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0E8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	ECMTE228	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMTE228	ECM Delay Timer Start Control 93	Enables delay timer operation in response to an ECM maskable interrupt caused by an ECM compare error (error source 93). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.34 ECM Delay Timer Configuration Register 3 (ECMDTMCFG3)

The ECMDTMCFG3 register controls delay timer operation in response to an ECM non-maskable interrupt. Whether to enable delay timer operation when an ECM non-maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0ECh

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	ECMTE322	ECMTE321	ECMTE320	ECMTE319	ECMTE318	ECMTE317	ECMTE316
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ECMTE313	ECMTE312	ECMTE311	ECMTE310	ECMTE309	ECMTE308	ECMTE307	ECMTE306	ECMTE305	ECMTE304	—	ECMTE302	—	ECMTE300
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMTE300	ECM Delay Timer Start Control 1	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a WDT overflow/ refresh error (for Cortex-R4) (error source 1). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMTE302	ECM Delay Timer Start Control 3	Enables delay timer operation in response to an ECM non-maskable interrupt caused by an IWDTa overflow/ refresh error (error source 3). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b3	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b4	ECMTE304	ECM Delay Timer Start Control 5	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Tag RAM) (error source 5). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b5	ECMTE305	ECM Delay Timer Start Control 6	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error (with correction) or 2-bit ECC error (without correction) in the instruction cache (Data RAM) (error source 6). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b6	ECMTE306	ECM Delay Timer Start Control 7	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the data cache (Tag/Dirty RAM) (error source 7). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b7	ECMTE307	ECM Delay Timer Start Control 8	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the data cache (Tag/Dirty RAM) (error source 8). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMTE308	ECM Delay Timer Start Control 9	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the data cache (Data RAM) (error source 9). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b9	ECMTE309	ECM Delay Timer Start Control 10	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the data cache (Data RAM) (error source 10). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b10	ECMTE310	ECM Delay Timer Start Control 11	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the ATCM (error source 11). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b11	ECMTE311	ECM Delay Timer Start Control 12	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the ATCM (error source 12). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b12	ECMTE312	ECM Delay Timer Start Control 13	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the BTCM (error source 13). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b13	ECMTE313	ECM Delay Timer Start Control 14	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the BTCM (error source 14). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b15, b14	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b16	ECMTE316	ECM Delay Timer Start Control 17	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 1-bit ECC error in the RSCAN RAM (error source 17). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b17	ECMTE317	ECM Delay Timer Start Control 18	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a 2-bit ECC error in the RSCAN RAM (error source 18). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b18	ECMTE318	ECM Delay Timer Start Control 19	Enables delay timer operation in response to an ECM non-maskable interrupt caused by an RSCAN overflow error (error source 19). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b19	ECMTE319	ECM Delay Timer Start Control 20	Enables delay timer operation in response to an ECM non-maskable interrupt caused by main clock oscillation stop detection (error source 20). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b20	ECMTE320	ECM Delay Timer Start Control 21	Enables delay timer operation in response to an ECM non-maskable interrupt caused by CLMA0 oscillation stop detection (PLL0) (error source 21). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b21	ECMTE321	ECM Delay Timer Start Control 22	Enables delay timer operation in response to an ECM non-maskable interrupt caused by CLMA1 oscillation stop detection (PLL1) (error source 22). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b22	ECMTE322	ECM Delay Timer Start Control 23	Enables delay timer operation in response to an ECM non-maskable interrupt caused by CLMA2 oscillation stop detection (LOCO) (error source 23). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b23	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.35 ECM Delay Timer Configuration Register 4 (ECMDTMCFG4)

The ECMDTMCFG4 register controls delay timer operation in response to an ECM non-maskable interrupt. Whether to enable delay timer operation when an ECM non-maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0F0h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	ECMTE408	ECMTE407	ECMTE406	ECMTE405	ECMTE404	ECMTE403	ECMTE402	—	ECMTE400
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	ECMTE400	ECM Delay Timer Start Control 33	Enables delay timer operation in response to an ECM non-maskable interrupt caused by a bus error (error source 33). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b1	—	Reserved	This bit is read as 0. The write value should be 0.	R/W
b2	ECMTE402	ECM Delay Timer Start Control 35	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 35 (error source 35). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b3	ECMTE403	ECM Delay Timer Start Control 36	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 36 (error source 36). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b4	ECMTE404	ECM Delay Timer Start Control 37	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 37 (error source 37). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b5	ECMTE405	ECM Delay Timer Start Control 38	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 38 (error source 38). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b6	ECMTE406	ECM Delay Timer Start Control 39	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 39 (error source 39). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b7	ECMTE407	ECM Delay Timer Start Control 40	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 40 (error source 40). 0: Delay timer start disabled 1: Delay timer start enabled	R/W

Bit	Symbol	Bit Name	Description	R/W
b8	ECMTE408	ECM Delay Timer Start Control 41	Enables delay timer operation in response to an ECM non-maskable interrupt caused by extended pseudo error 41 (error source 41). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b9	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.36 ECM Delay Timer Configuration Register 5 (ECMDTMCFG5)

The ECMDTMCFG5 register controls delay timer operation in response to an ECM non-maskable interrupt. Whether to enable delay timer operation when an ECM non-maskable interrupt is generated can be set.
This register is a common register. Settings for both ECM master and ECM checker can be performed by writing to this register.
Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.

Address(es): A007 D0F4h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	ECMTE528	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b27 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R/W
b28	ECMTE528	ECM Delay Timer Start Control 93	Enables delay timer operation in response to an ECM non-maskable interrupt caused by an ECM compare error (error source 93). 0: Delay timer start disabled 1: Delay timer start enabled	R/W
b31 to b29	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.2.37 ECM Error Output Clear Disable Configuration Register (ECMEOCCFG)

The ECMEOCCFG register controls the setting to disable clearing of an error output signal.
Clearing an error output signal by the ECMmECLR register (m = M or C) can be disabled until the value of the error output clear disable counter exceeds the value set in this register.
Setting to this register is possible only when no error source occurs.
This register is reset by the RES# pin reset only, and is not reset by other reset sources.
This register is a common register. Writing to this register is protected by the specific instruction sequence. For details, see section 32.3.4, Writing to Protected Registers.
Write this register after confirming that the read value from the CMPW bit is 1.

Address(es): A007 D0F8h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CMPW
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	ECMEOUTCLRT[15:0]															
Value after reset:	000h															

Bit	Symbol	Bit Name	Description	R/W
b15 to b0	ECMEOUTCLRT[15:0]	Error Output Signal Clear Disabled Period Setting	Sets the clear disabled period for an error output signal. (clear disabled period) = (N + 1) × (ECMCLK cycle (240 kHz)) N: Set value For details, see section 32.3.6, Setting of Disabling Error Output Clear.	R/W
b16	CMPW	Compare Write	Indicates that ECM is waiting for the operation of the control circuit for clearing an ECM error output to be stable after reset is released. Write this register after confirming that the read value is 0. 0: Operating stably 1: Waiting for stable operation	R
b31 to b17	—	Reserved	These bits are read as 0.	R

32.2.38 ECM Mask Control Register (ECMMCNT)

The ECMMCNT register is used for mask control for ECM compare errors (error source 93).

Address(es): A00B 0A80h

	b31	b30	b29	b28	b27	b26	b25	b24	b23	b22	b21	b20	b19	b18	b17	b16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MSKM	MSKC
Value after reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b0	MSKC	ECM Compare Error Mask for Checker	Masks the ECM compare error source (error source 93) on the ECM checker. 0: ECM compare error not masked 1: ECM compare error masked	R/W
b1	MSKM	ECM Compare Error Mask for Master	Masks the ECM compare error source (error source 93) on the ECM master. 0: ECM compare error not masked 1: ECM compare error masked	R/W
b31 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R/W

32.3 Operations

32.3.1 Operations for Error Output

There are two output modes (non-dynamic and dynamic) for the ERROROUT# pin when an error source occurs.
In non-dynamic mode, the ERROROUT# pin outputs the high level when an error is not detected.
In dynamic mode, the ERROROUT# pin can output a toggle waveform according to the compare match timer W (CMTW) when an error is not detected.
In both modes, the ERROROUT# pin outputs the low level when an error is detected.

Operating Mode	Error Status	ERROROUT# Pin Output Level
Non-dynamic (ECMEPCFG.ECMSL0 bit = 0)	No error*1	High
	Error	Low
Dynamic (ECMEPCFG.ECMSL0 bit = 1)	No error*1	Toggles (according to CMTW input)
	Error	Low

Note 1. After reset is released, the ERROROUT# pin outputs the low level (error status).
Use this pin after clearing the error status according to the procedure described in the note in section **32.2.2**.

32.3.1.1 Dynamic Mode Enable

1. Initialize the compare match timer W (CMTW) for the input signal for toggle output. The output compare signal for toggle output can be selected by the ECDMESLR register.
   For details on the CMTW setting, see section 19, Compare Match Timer W (CMTW).
2. Set the output of the ERROROUT# pin to the high level (no error) by setting the ECMmECT bit (m = M or C) of the ECM master/checker error clear trigger register to 1.
3. Set the ECMSL0 bit in the ECM error pulse configuration register (ECMPCFG) to 1 to specify dynamic mode.
4. Start up the CMTW.

32.3.1.2 Dynamic Mode Disable

1. Set the output of the ERROROUT# pin to the low level by setting the ECMmEST bit (m = M or C) in the ECM master/checker error set trigger register to 1.
2. Stop the CMTW operation.
3. Clear the ECMSL0 bit in the ECM error pulse configuration register (ECMPCFG) to 0 to specify non-dynamic mode.

32.3.2 Loop-Back Function

ECM incorporates a loop-back function that is used to check the path for an error output signal from the ECM module to the ERROROUT# pin. The output level of the ERROROUT# pin can be checked with the ECMmSSE231 bit (m = M or C) in the ECM master/checker error source status register 1.

32.3.3 Pseudo Error Generation

ECM can generate individual pseudo error sources for self-diagnosis. ECM generates a pseudo error when the corresponding bit in the ECM pseudo error trigger register (ECMPEn) is set to 1. When a pseudo error is generated, ECM operates in the same way as when a real error occurs, and the settings for error source mask, ECM interrupt, ECM reset, and delay timer all apply in the same way.
Error sources 35 to 41 can be used as extended pseudo errors to detect errors for functional safety as shown below. When software detects an error, it can set a bit in the ECMPEn register to generate a corresponding pseudo error to use a function such as ECM interrupt, ECM reset, or delay timer.

Table 32.3 Example Assignment of Errors for Functional Safety to Error Sources

Example Functions to Be Assigned to Extended Pseudo Errors
Access violation to the Cortex-R4 protected area

32.3.4 Writing to Protected Registers

Write protected registers are protected from incorrect write access due to erroneous program execution, etc.

32.3.4.1 Protection Unlock Sequence

Write access to a write protected register is only possible within the following sequence.

1. Write the fixed value 0000 00A5h to the ECM protection command register (ECMPCMD1) or the ECM master/checker protection command register (ECMmPCMD0) (m = M or C). If the register to be written is the common register, write the fixed value to the ECM protection command register (ECMPCMD1).
   If the register to be written is not the common register, write the fixed value to the ECM master/checker protection command register (ECMmPCMD0).
   See descriptions of each register in section 32.2, Registers, to find out whether the target register is a common one or not.
2. Write the desired setting value to the protected registers of the ECM common, ECM master, and ECM checker according to the following sequence:
   - Write the desired setting value.
   - Write the inversed value of the desired setting value.
   - Write the desired setting value again.
3. Check the desired setting value has been successfully written to the protected register by checking that the ECMPRERR bit of the ECM protection status register (ECMPS) is 0.

In case of any access to another register between step 1 to step 3 of the above sequence, the protection mechanism behaves as follows.

- If that register belongs to the ECM, the write to the protected register fails (the ECMPRERR bit of the ECM protection status register becomes 1). The sequence has to be reexecuted from step 1.
- If that register does not belong to the ECM, the sequence is not disrupted and the write to the protected register is conducted successfully.

In case the protection unlock sequence is interrupted, the protection mechanism behaves as follows.

- Interrupts during protection sequence
  If an interrupt is acknowledged within the protection sequence and the interrupt process does not access any ECM register, the protection sequence is not disrupted and the write to the protected register can be successfully completed after returning from the interrupt process.

In case a break occurs in the protection unlock sequence, the protection mechanism behaves as follows.

- Breaks during protection sequence
  If the CPU goes into the break state during the protection sequence and any ECM register is not accessed, the protection sequence is not disrupted and the write to the protected register can be successfully completed after returning from the break.
  If the CPU goes into the break state during the protection sequence and any ECM register is accessed, the protection sequence is disrupted and the write to the protected register is not performed after returning from the break.
  Therefore, be careful not to let the CPU go into the break state during the protection sequence.

32.3.5 Timeout Function for Interrupt Processing by Using the Delay Timer

ECM can start the delay timer at the same time as an ECM maskable or non-maskable interrupt request due to occurrence of an error source is issued, to manage timeout of the interrupt processing time. If ECM cannot stop the delay timer (by setting the ECMDTMCTL.DTMSTP bit to 1) during interrupt processing and the count value of the delay timer matches with the value of the delay timer compare register, ECM can generate a delay timer overflow (error source 94), and an error signal output on the ERROROUT# pin or ECM reset. Specify the settings for the error signal output and ECM reset when the delay timer overflow (error source 94) occurs, in the ECMEMK2 and ECMIRCFG2 registers, respectively.
The delay timer always starts counting up from 0 by using ECMCLK (240 kHz).
Specify the setting of the overflow cycle of the delay timer in the ECMDTMCMP register.

(Overflow cycle) = (value set in ECMDTMCMP. ECMDTMCMP[15:0] + 1) × ECMCLK cycle (240 kHz)

Note: The delay timer continues count operation even when a break occurs.

32.3.6 Setting of Disabling Error Output Clear

ECM has a function that disables clearing (by the ECMmECLR register (m = M or C)) of an error signal output from the ERROROUT# pin when an error occurs. An error output signal is disabled during the specified period. This period can be specified in the ECMEOCCFG register.
When an error source occurs, the error output clear disable counter starts counting up by using the ECMCLK clock (240 kHz). Clear processing by the ECMmECLR register for an error output is disabled until the counter value matches with the value set in ECMEOCCFG.ECMEOUTCLRT[15:0]. After the counter value exceeds the set value, clearing of an error output is possible.

(Error output clear disabled period) = (ECMEOUTCLRT[15:0] + 1) × ECMCLK cycle (240 kHz)

If another error source occurs while the error output clear disable counter is in count operation, the counter is cleared and restarts counting up. If the same error source occurs again, the counter is not cleared and continues counting up.

Note: The error output disable counter continues count operation even when a break occurs.

32.4 Usage Notes

32.4.1 Notes Regarding ECMCLK

Counting by the delay timer and by the counter to disable clearing of the error output is of cycles of ECMCKL, a signal generated by the low-speed on-chip oscillator.
After release from the reset state, counting does not proceed or the error output is not cleared if the delay timer is started while the low-speed on-chip oscillator is stopped.
To enable the delay timer function and error output, enable the low-speed on-chip oscillator beforehand and wait for the LOCO oscillation stabilization time.
For details on the low-speed on-chip oscillator, see section 7.2.4, Low-Speed On-Chip Oscillator Control Register (LOCOCR).

33. Electrical Characteristics

33.1 Absolute Maximum Ratings

Table 33.1 Absolute Maximum Rating

Conditions: VSS = PLLVSS0 = PLLVSS1 = VSS_USB = 0 V

Item	Symbol	Value	Unit
Power supply voltage (I/O)	VCCQ33	–0.3 to +4.2	V
Power supply voltage (internal)	VDD	–0.3 to +1.6	V
PLL power supply voltage	PLLVDD0, PLLVDD1	–0.3 to +1.6	V
Input voltage (except for ports for 5-V tolerant*1)	V_{in1}	–0.3 to VCCQ33 + 0.3*5	V
Input voltage (ports for 5-V tolerant*1)	V_{in2}	–0.3 to +5.5*3	V
USB digital power supply voltage	DVDD_USB	–0.3 to +1.6	V
USB power supply voltage	VDD33_USB*2	–0.3 to +4.2	V
Operating temperature (junction temperature)	Tj*4	–40 to +125	°C
Storage temperature	T_{stg}	–55 to +125	°C

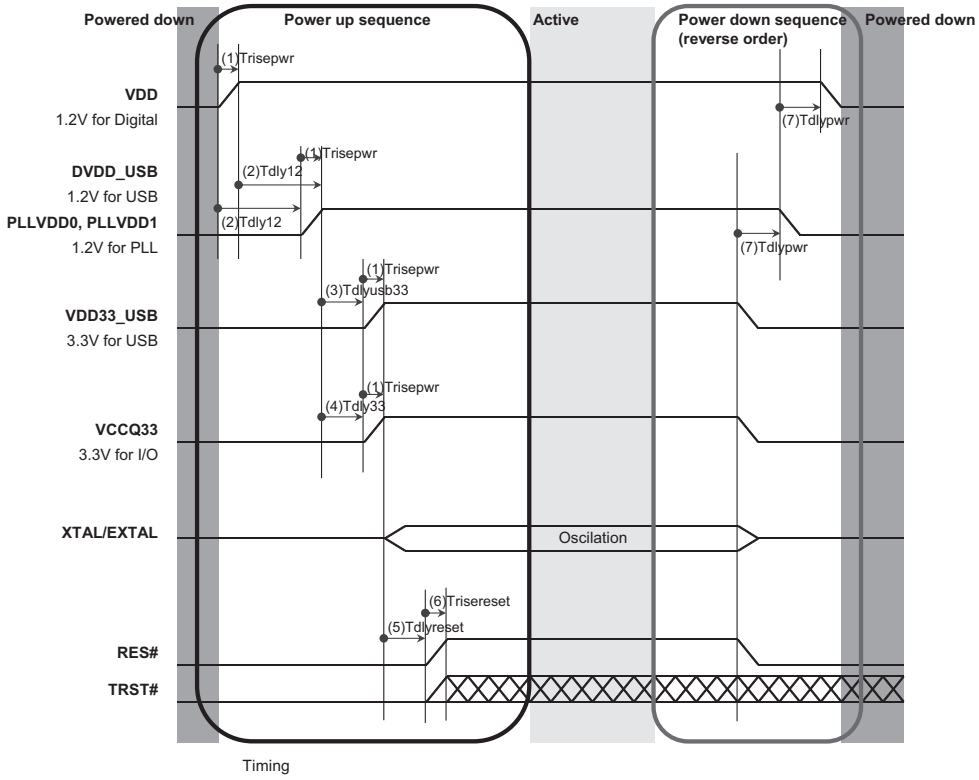
[Usage Notes]

1. Do not directly connect output pins (I/O pins in output state) of IC products to other output pins (including I/O pins in output state) , power pins, or GND pins. However, output pins are directly connectable in an external circuit where timing design is provided to avoid conflict of outputs of high-impedance pins such as I/O pins.
2. If even a single item exceeds the absolute maximum rating for even a moment, it may degrade the product's quality. In other words, the absolute maximum rating is a rated value that potentially causes physical damage to products. Use products with a margin of the absolute maximum rating.
   Specified values and conditions shown in DC characteristics and AC characteristics are the range of normal operation and quality assurance of products.

Note 1. Ports PC0 to PC7 are 5-V tolerant.
Note 2. When the USB is not to be used, connect the VDD33_USB pin to VCCQ33, the VSS_USB pin to VSS, and the DVDD_USB pin to VDD, respectively. Do not leave these pins open.
Note 3. When VCCQ33 is less than 3.0 V, the rated value of ports for 5-V tolerant is 3.6 V.
Note 4. For operations at the temperatures over 110 °C (junction temperature), refer to the “Precautions for High-Temperature Operations with the EC-1 Group (R01AN3998).
Note 5. Do not exceed the absolute maximum rating, 4.2 V.

33.2 Power On/Off Sequence

Turn on and off each power supply voltage according to the procedure shown in the figure below.
When turning on the power, be sure to fix TRST# pins and RES# pins to the low level. Otherwise, initialization is not performed successfully.



Timing

		Value		
No.	**Item**	**min**	**typ**	**max**
(1)	Trisepwr	100 us	—	50 ms
(2)	Tdly12	0 ms	—	100 ms
(3)	Tdlyusb33	0 ms	—	100 ms
(4)	Tdly33	0 ms	—	100 ms
(5)	Tdlyreset	10 ms	—	—
(6)	Trisereset	—	—	150 µs
(7)	Tdlypwr	0 ms	—	—

Note 1. Turn on all power supply voltages and reset signals with a monotonic increase and turn off with a monotonic decrease.
Note 2. Do not apply a negative voltage to power supply voltages.
Note 3. Before turning on the power, be sure to make reset pins (TRST# and RES#) active (low). Otherwise, inputs and outputs on the pins may be unstable. If this may also be a problem when turning off the power, make reset pins (TRST# and RES#) active (low.)

Note: If the power on/off sequence is not met (out of operation guarantee range), the pin input state and output state may be undefined.

Figure 33.1 Power On/Off Sequence

33.3 DC Characteristics

- Conditions: VDD = PLLVDD0 = PLLVDD1 = DVDD_USB = 1.14 to 1.26 V,
  VCCQ33 = VDD33_USB = 3.0 to 3.6 V
  VSS = PLLVSS0 = PLLVSS1 = VSS_USB = 0 V,
  Tj = –40 to 125°C

Table 33.2 DC Characteristics (1)

Item	Symbol	min	typ	max	Unit	Test Conditions
Power supply voltage (I/O)	VCCQ33	3.0	3.3	3.6	V	
Power supply voltage (internal)	VDD	1.14	1.2	1.26	V	
PLL power supply voltage	PLLVDD0, PLLVDD1	1.14	1.2	1.26	V	
USB digital power supply voltage	DVDD_USB	1.14	1.2	1.26	V	
USB power supply voltage	VDD33_USB	3.0	3.3	3.6	V	

Table 33.3 DC Characteristics (2) [Power Supply]

Item	Type	Symbol	typ	max	Unit	Test Conditions
Normal operation	VDD 150MHz	VIcc	107	555	mA	Tj = -40 to 125°C
	PLLVDD0 + PLLVDD1	PLLIcc	3.2	5	mA	
	VCCQ33	V33Icc	19*1, *2	—	mA	
	DVDD_USB	V12UIcc	5.1	9	mA	USB high-speed communication
			3.5	9	mA	USB full-speed communication
	VDD33_USB	V33UIcc	15*1	—	mA	USB high-speed communication
			10*1	—	mA	USB full-speed communication
Standby mode with all modules inactive (reference value)	VDD	VIcc	41	—	mA	
	PLLVDD0 + PLLVDD1	PLLIcc	3.2	—	mA	
	VCCQ33	V33Icc	0.35*1, *2	—	mA	
	DVDD_USB	V12UIcc	3.5	—	mA	UTMI suspend mode
	VDD33_USB	V33UIcc	9.6*1	—	mA	UTMI suspend mode

Note 1. These values are reference values. The actual operating current greatly depends on the system (such as unsharpened waveforms due to I/O load and toggle frequency). Be sure to measure these current values in the system.
Note 2. V33Icc must be 80 mA or less. (ΣI_{OH} in Table 33.9)

Table 33.4 DC Characteristics (3) [Except for USB2.0 Host/Function-Related Pins]

Item		Symbol	min	typ	max	Unit	Test Conditions
Schmitt trigger Input voltage	Other than 5-V tolerant pins	V_{IH1}	2.4	—	VCCQ33 + 0.3	V	
		V_{IL1}	–0.3	—	0.8	V	
		ΔV_{T1}	VCCQ33 × 0.05	—	—	V	
	5-V tolerant pins*1	V_{IH2}	VCCQ33 × 0.7	—	5.3*2	V	
		V_{IL2}	–0.3	—	VCCQ33 × 0.3	V	
		ΔV_{T2}	VCCQ33 × 0.05	—	—	V	
Input high level voltage (except for schmitt trigger input pins)		V_{IH3}	2.4	—	VCCQ33 + 0.3	V	
Input low level voltage (except for schmitt trigger input pins)		V_{IL3}	–0.3	—	0.8	V	
Output high level voltage	Other than 5-V tolerant pins	V_{OH}	VCCQ33 – 0.5	—	—	V	I_{OH} = –2 mA
Output low level voltage	Other than 5-V tolerant pins	V_{OL1}	—	—	0.4	V	I_{OL1} = 2 mA
	5-V tolerant pins*1	V_{OL2}	—	—	0.4	V	I_{OL2} = 3 mA
			—	—	0.6	V	I_{OL2} = 6 mA
Input leakage current		$\lvert I_{in} \rvert$	—	—	1.0	µA	$V_{in1} = V_{in2}$ = 0 V $V_{in1} = V_{in2}$ = VCCQ33
Three-state leakage current (off state)	Input/output and output pins excluding 5-V tolerant pins	$\lvert I_{TSI} \rvert$	—	—	1.0	µA	V_{in1} = 0 V V_{in1} = VCCQ33
	5-V tolerant pins*1		—	—	5.0	µA	V_{in2} = 0 V V_{in2} = VCCQ33
Input pull-up MOS current/ resistance	Ports P50 to P54, P56, P86, P87, P90 to P97, PD5 to PD7	I_{pu1}	–300	—	–30	µA	VCCQ33 = 3.0 to 3.6 V $V_{in1} = V_{in2}$ = 0 V
		R_{pu1}	10	—	120	kΩ	
	Pins other than the above*3	I_{pu2}	–120	—	–7	µA	VCCQ33 = 3.0 to 3.6 V $V_{in1} = V_{in2}$ = 0 V
		R_{pu2}	25	—	515	kΩ	
Input pull-down MOS current/ resistance	Ports P50 to P54, P56, P86, P87, P90 to P97, PD5 to PD7	I_{pd1}	30	—	300	µA	VCCQ33 = 3.0 to 3.6 V $V_{in1} = V_{in2}$ = VCCQ33
		R_{pd1}	10	—	120	kΩ	
	Pins other than the above*3	I_{pd2}	7	—	120	µA	VCCQ33 = 3.0 to 3.6 V $V_{in1} = V_{in2}$ = VCCQ33
		R_{pd2}	25	—	515	kΩ	
Pin capacity	All input/output and input pins	C_{in}	—	—	10	pF	

Note 1. Ports PC0 to PC7 are 5-V tolerant.
Note 2. When VCCQ33 is less than 3.0 V, do not apply voltage of 3.6 V or higher to 5-V tolerant pins.
Note 3. 5-V tolerant pins are not included.

Table 33.5 DC Characteristics (4) [USB2.0 USB_RREF Pin]

Item	Symbol	min	typ	max	Unit	Test Conditions
Reference resistor	R_{REF}	200 ±1%			Ω	

Table 33.6 DC Characteristics (5) [USB2.0 Host/Function-Related Pins (Items for Both Full Speed and High Speed)*1]

Item	Symbol	min	typ	max	Unit	Test Conditions
DP pull-up resistor (when the function controller operation is selected)	R_{PU}	0.900	—	1.575	kΩ	Idle
		1.425	—	3.090	kΩ	Transmission/ reception
DP/DM pull-down resistors (when the host function is selected)	R_{PD}	14.25	—	24.80	kΩ	

Note 1. USB_DP and USB_DM pins

Table 33.7 DC Characteristics (6) [USB2.0 Host/Function-Related Pins (Full Speed)*1]

Item	Symbol	min	typ	max	Unit	Measuring Condition
Input high level voltage	V_{FSIH}	2.0	—	—	V	
Input low level voltage	V_{FSIL}	—	—	0.8	V	
Differential input sensitivity	V_{FSDI}	0.2	—	—	V	| (USB_DP) – (USB_DM) |
Differential common mode range	V_{FSCM}	0.8	—	2.5	V	
Output high level voltage	V_{FSOH}	2.8	—	3.6	V	I_{FSOH} = –200 µA
Output low level voltage	V_{FSOL}	0.0	—	0.3	V	I_{FSOL} = 2 mA
Output signal crossover voltage	V_{FSCRS}	1.3	—	2.0	V	CL = 50 pF (full-speed)

Note 1. USB_DP and USB_DM pins

Table 33.8 DC Characteristics (7) [USB2.0 Host/Function-Related Pins (High Speed)*1]

Item	Symbol	min	typ	max	Unit	Test Conditions
Squelch detection threshold voltage (differential voltage)	V_{HSSQ}	100	—	150	mV	
Common mode voltage range	V_{HSCM}	–50	—	500	mV	
Idle state	V_{HSOI}	–10.0	—	10.0	mV	
Output high level voltage	V_{HSOH}	360	—	440	mV	
Output low level voltage	V_{HSOL}	–10.0	—	10.0	mV	
Chirp J output voltage (differential)	V_{CHIRPJ}	700	—	1100	mV	
Chirp K output voltage (differential)	V_{CHIRPK}	–900	—	–500	mV	

Note 1. USB_DP and USB_DM pins

Table 33.9 Permissible Output Currents

Item		Symbol	min	typ	max	Unit
Permissible output low current (average value per pin)	Other than 5-V tolerant pins	I_{OL1}	—	—	2.0	mA
	5-V tolerant pins	I_{OL2}	—	—	3.0	mA
Permissible output low current (maximum value per pin)	Other than 5-V tolerant pins	I_{OL1}	—	—	4.0	mA
	5-V tolerant pins	I_{OL2}	—	—	6.0	mA
Permissible output low current (total)	Total of all output pins	ΣI_{OL}	—	—	80	mA
Permissible output high current (average value per pin)	All output pins	I_{OH}	—	—	–2.0	mA
Permissible output high current (maximum value per pin)	All output pins	I_{OH}	—	—	–4.0	mA
Permissible output high current (total)	Total of all output pins	ΣI_{OH}	—	—	–80	mA

[Usage Note] All output current values shall be within the values in Table 33.9 to ensure the reliability of this LSI.

33.4 AC Characteristics

- Conditions: VDD = PLLVDD0 = PLLVDD1 = DVDD_USB = 1.14 to 1.26 V,
  VCCQ33 = VDD33_USB = 3.0 to 3.6 V
  VSS = PLLVSS0 = PLLVSS1 = VSS_USB = 0 V,
  Tj = –40 to 125 °C

Table 33.10 Operating Frequency

Item		Symbol	min	max	Unit
Operating frequency	CPU clock (CPUCLK)	f	150		MHz
	System clock (ICLK)		150		
	Peripheral module clock (PCLKA)		150		
	Peripheral module clock (PCLKB)		75		
	Peripheral module clock (PCLKD)		75		
	Peripheral module clock (PCLKE)		18.75	75	
	High-speed serial clock (SERICLK)		120	150	
	External clock output for EtherCAT PHY (CLKOUT25Mn)		25		

n = 0, 1

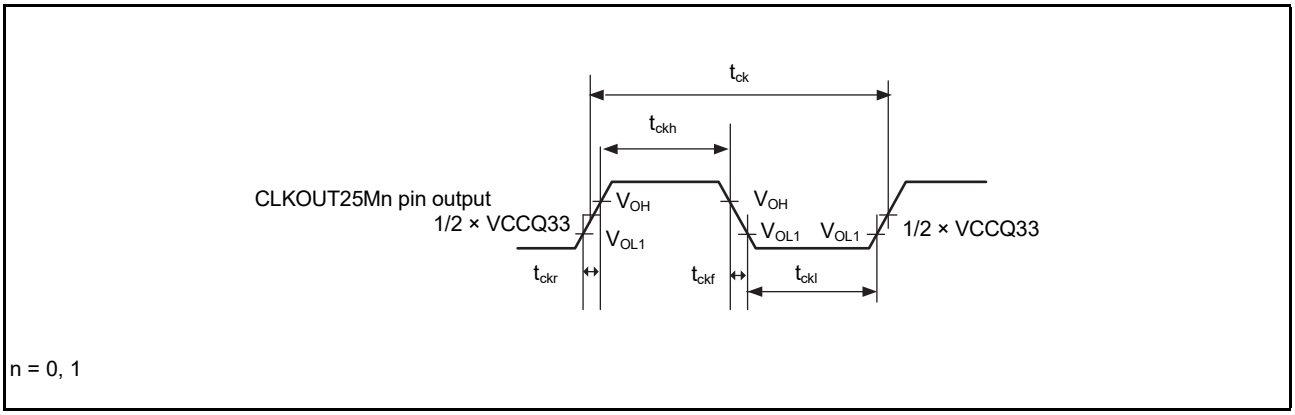
33.4.1 Clock Timing

Table 33.11 CLKOUT25Mn Timing

Output load conditions: V_{OH} = VCCQ33 – 0.5 V, V_{OL1} = 0.4 V, C = 30 pF

Item		Symbol	min	max	Unit	Test Conditions
CLKOUT25Mn (MII)	CLKOUT25Mn cycle time	T_{ck}	40	—	ns	Figure 33.2
	CLKOUT25Mn frequency Typ. 25 MHz	—	25 ± 50 ppm		MHz	
	CLKOUT25Mn duty ratio	—	35	65	%	
	CLKOUT25Mn output low pulse width	T_{ckl}	$T_{ck}/2 - T_{ckf}$	$T_{ck}/2 + T_{ckf}$	ns	
	CLKOUT25Mn output high pulse width	T_{ckh}	$T_{ck}/2 - T_{ckr}$	$T_{ck}/2 + T_{ckr}$	ns	
	CLKOUT25Mn rising/falling time	$T_{ckr2/ckf}$	0.5	9	ns	

n = 0, 1



n = 0, 1

Figure 33.2 CLKOUT25Mn Pin Output Timing 2

Table 33.12 XTAL Clock Timing

Item	Symbol	min	typ	max	Unit
XTAL clock oscillator output cycle*1	$t_{XTALcyc}$	40.00 ± 50 ppm*2			ns

Note 1. When using the XTAL clock, ask the oscillator manufacturer to evaluate oscillation of the oscillator. For the oscillation stabilization time, see the evaluation result provided by the oscillator manufacturer.
Note 2. When using the EtherCAT slave controller, make sure that the clock timing satisfies 25.00 MHz ± 25 ppm.

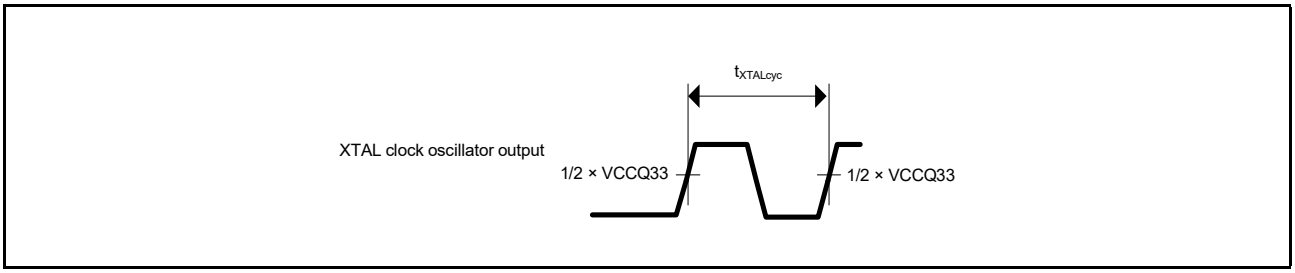


Figure 33.3 XTAL Clock Oscillator Output Timing

Table 33.13 LOCO Clock Timing

Item	Symbol	min	typ	max	Unit	Test Conditions
LOCO clock cycle time	t_{Lcyc}	4.62	4.17	3.79	µs	
LOCO clock oscillation frequency	f_{LOCO}	216	240	264	kHz	
LOCO clock oscillation stabilization wait time	t_{LOCOWT}	—	—	40	µs	Figure 33.4

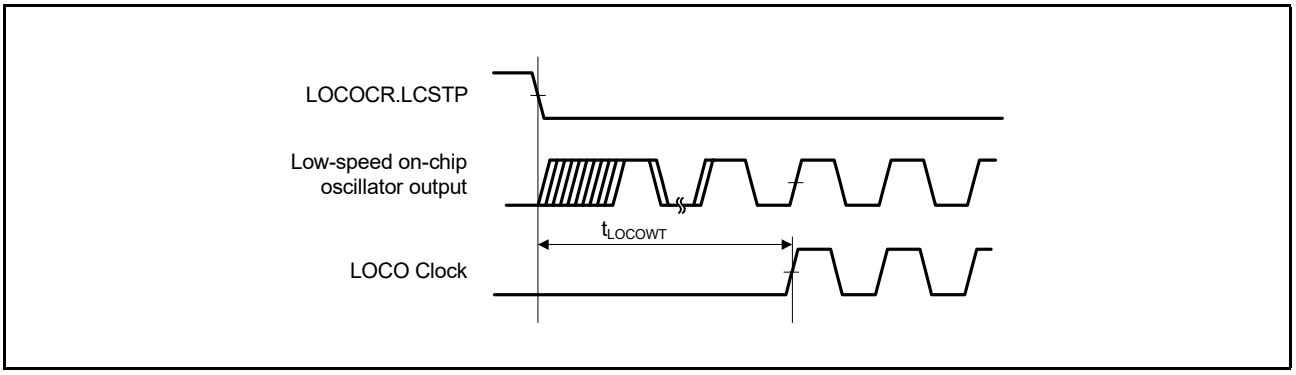


Figure 33.4 LOCO Clock Oscillation Start Timing

33.4.2 Reset Timing and Interrupt Timing

Table 33.14 Reset Timing and Interrupt Timing

Item		Symbol	Min*1	typ	max	Unit	Test Conditions
RES# pulse width	At power on	$T_{dlyreset}$	10	—	—	ms	Figure 33.5
	Other than above	$T_{dlyreset2}$	1	—	—	ms	
RES# rising time		$T_{risereset}$	—	—	150	µs	
TRST# pulse width	At power on	$T_{dlyreset}$	10	—	—	ms	
	Other than above	$T_{dlyreset2}$	1	—	—	ms	
TRST# rising time		$T_{risereset}$	—	—	150	µs	
NMI pulse width		t_{NMIW}	t_{Icyc} × 2	—	—	ns	Figure 33.6
IRQ pulse width		t_{IRQW}	t_{Icyc} × 2	—	—	ns	Figure 33.7
ETH_INT pulse width		t_{EINTW}	t_{Icyc} × 2	—	—	ns	Figure 33.8

Note 1. t_{Icyc}: ICLK cycle

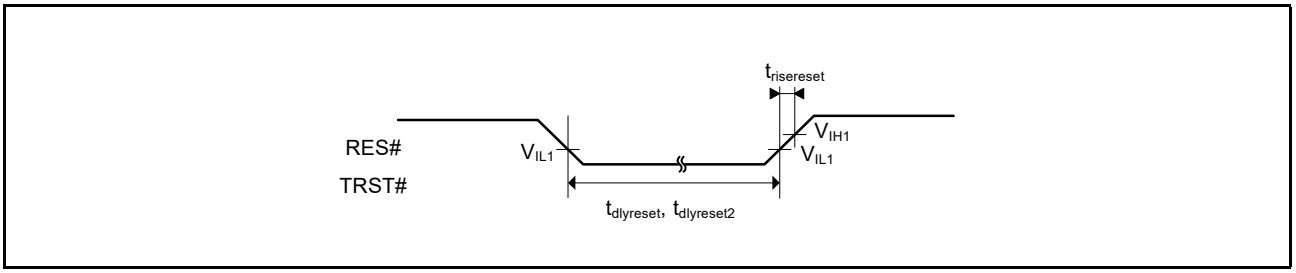


Figure 33.5 Reset Input Timing

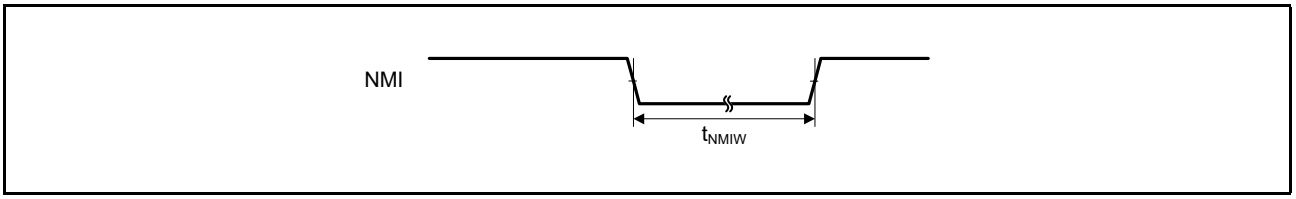


Figure 33.6 NMI Interrupt Input Timing

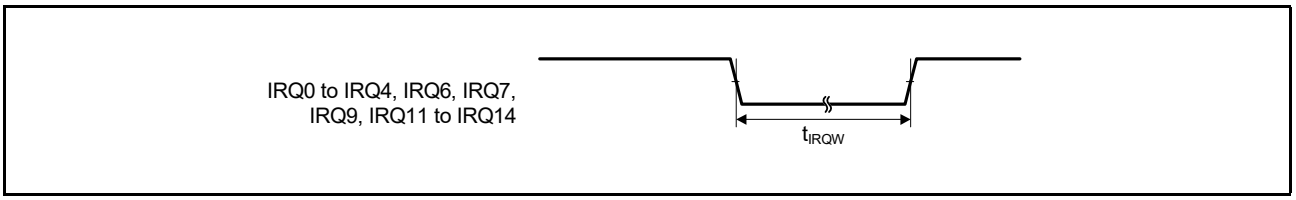


Figure 33.7 IRQ Interrupt Input Timing

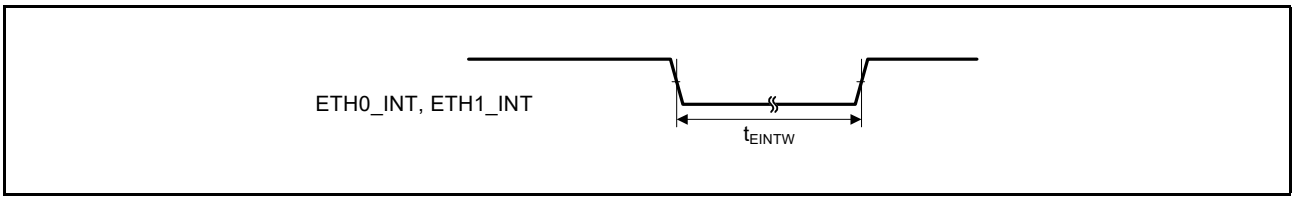


Figure 33.8 ETH_INT Interrupt Input Timing

33.4.3 On-Chip Peripheral Module Timing

33.4.3.1 I/O Port Timing

Table 33.15 I/O Port Timing

Item		Symbol	min	max	Unit*1	Test Conditions
I/O port	Input data pulse width	t_{PRW}	1.5	—	t_{PBcyc}	Figure 33.9

Note 1. t_{PBcyc}: PCLKB cycle

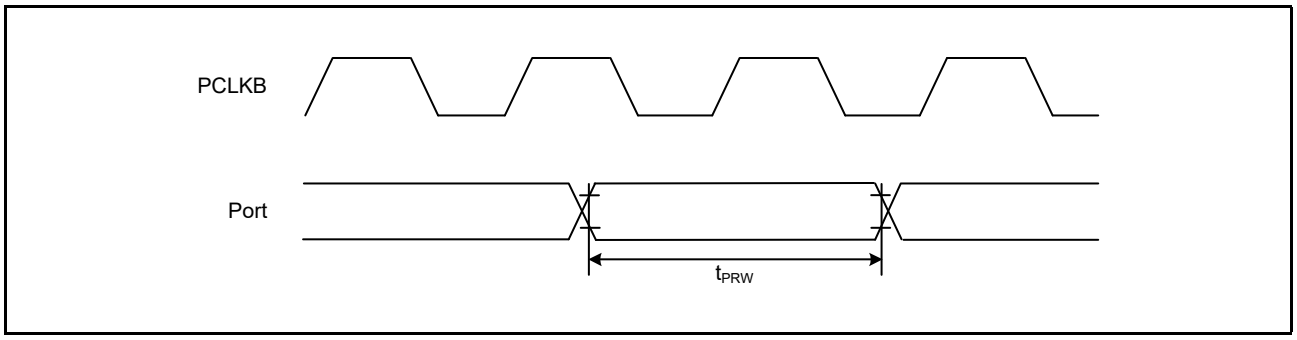


Figure 33.9 I/O Port Input Timing

33.4.3.2 CMTW Timing

Table 33.16 CMTW Timing

Item			Symbol	min	max	Unit*1	Test Conditions
CMTW	Input capture input pulse width	Single-edge setting	$t_{CMTWICW}$	1.5	—	t_{PDcyc}	Figure 33.10
		Both-edge setting		2.5	—		

Note 1. t_{PDcyc}: PCLKD cycle

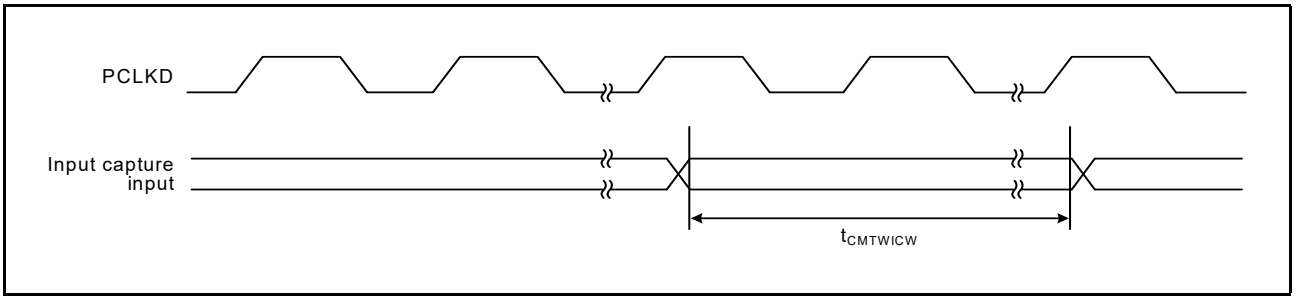


Figure 33.10 CMTW Input Capture Input Timing

33.4.3.3 SCIFA Timing

Table 33.17 SCIFA Timing

Output load conditions: V_{OH} = VCCQ33 × 0.5, V_{OL1} = VCCQ33 × 0.5, C = 30 pF

Item			Symbol	min*1	max*1	Unit*1	Test Conditions
SCIFA	Input clock cycle	Asynchronous	t_{Scyc}	4	—	t_{SEcyc}	Figure 33.11
		Clock synchronous		12	—		
	Input clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Input clock rising time		t_{SCKr}	—	5	ns	
	Input clock falling time		t_{SCKf}	—	5	ns	
	Output clock cycle	Asynchronous*2	t_{Scyc}	8	—	t_{SEcyc}	
		Clock synchronous		4	—		
	Output clock pulse width		t_{SCKW}	0.4	0.6	t_{Scyc}	
	Output clock rising time		t_{SCKr}	—	9	ns	
	Output clock falling time		t_{SCKf}	—	9	ns	
	Transmit data delay time	Internal clock	t_{TXD}	– 10	10	ns	Figure 33.12
		External clock		3 × t_{SEcyc}	4 × t_{SEcyc} + 20		
	Receive data setup time	Internal clock	t_{RXS}	3 × t_{SEcyc} + 20	—	ns	
		External clock		t_{SEcyc} + 10	—		
	Receive data hold time	Internal clock	t_{RXH}	– 3 × t_{SEcyc}	—	ns	
		External clock		2 × t_{SEcyc} + 10	—		

Note 1. t_{SEcyc}: SERICLK cycle
Note 2. When the SEMR.ABCS0 bit = 1 and the SEMR.BGDM bit = 1

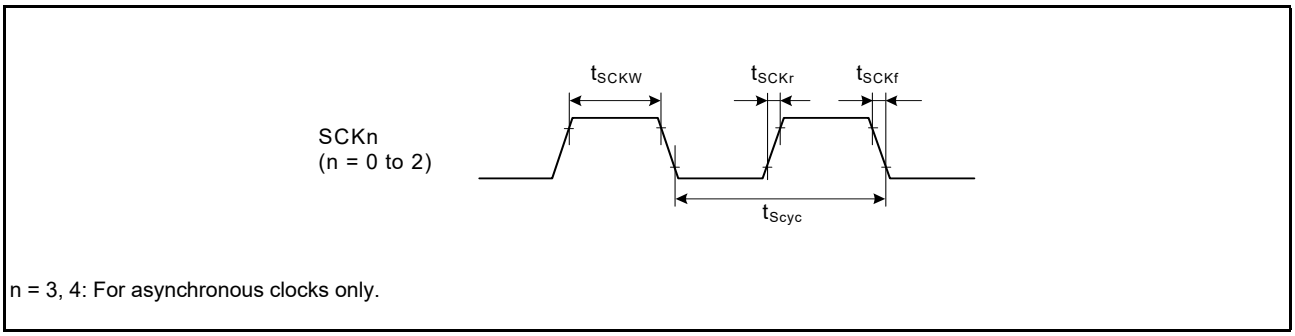


Figure 33.11 SCK Clock Input Timing

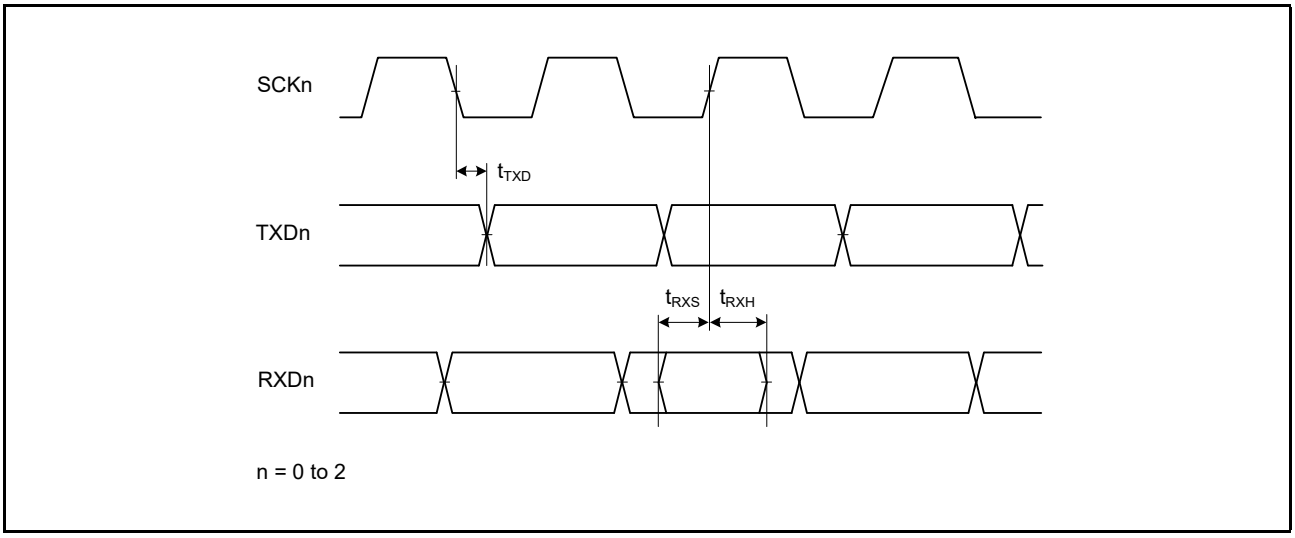


Figure 33.12 SCIFA Input/Output Timing/Clock Synchronous Mode

33.4.3.4 RSPIa Timing

Table 33.18 RSPIa Timing

Output load conditions: V_{OH} = VCCQ33 × 0.5, V_{OL1} = VCCQ33 × 0.5, C = 30 pF

Item			Symbol*1	Min*1	Max*1	Unit*1	Test Conditions
RSPIa	RSPCK clock cycle	Master	t_{SPcyc}	4	4096	t_{SEcyc}	Figure 33.13
		Slave*4		8	4096		
	RSPCK clock high level pulse width	Master	t_{SPCKWH}	$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF}) / 2 - 3$	—	ns	
		Slave		0.4	—	t_{SPcyc}	
	RSPCK clock low level pulse width	Master	t_{SPCKWL}	$(t_{SPcyc} - t_{SPCKR} - t_{SPCKF}) / 2 - 3$	—	ns	
		Slave		0.4	—	t_{SPcyc}	
	RSPCK clock rising/falling time	Output	t_{SPCKr}, t_{SPCKf}	—	9	ns	
		Input		—	10	ns	
	Data input setup time	Master	t_{SU}	6	—	ns	Figure 33.14 to Figure 33.17
		Slave		$8 - t_{SEcyc}$	—		
	Data input hold time	Master	t_H	t_{SEcyc}	—	ns	
		Slave		$8 + 2 \times t_{SEcyc}$	—		
	SSL setup time	Master	t_{LEAD}	$N \times t_{Spcyc} - 3$*2	$N \times t_{Spcyc} + 3$*2	ns	
		Slave		4	—	t_{SEcyc}	
	SSL hold time	Master	t_{LAG}	$N \times t_{Spcyc} - 3$*3	$N \times t_{Spcyc} + 3$*3	ns	
		Slave		4	—	t_{SEcyc}	
	Data output delay time	Master	t_{OD}	—	6	ns	
		Slave		—	$3 \times t_{SEcyc} + 20$*4		
	Data output hold time	Master	t_{OH}	0	—	ns	
		Slave		0	—		
	Continuous transmission delay	Master	t_{TD}	$t_{SPcyc} + 2 \times t_{SEcyc}$	$8 \times t_{SPcyc} + 2 \times t_{SEcyc}$	ns	
		Slave		$4 \times t_{SEcyc}$	—		
	MOSI, MISO rising/falling time	Output	t_{Dr}, t_{Df}	—	9	ns	
		Input		—	10	ns	
	SSL rising/falling time	Output	t_{SSLr}, t_{SSLf}	—	9	ns	
		Input		—	10	ns	
	Slave access time		t_{SA}	—	4	t_{SEcyc}	Figure 33.16 to Figure 33.17
	Slave output release time		t_{REL}	—	3	t_{SEcyc}	

Note 1. t_{SEcyc}: SERICLK cycle
Note 2. N = SPCKD set value + 1 (1 to 8)
Note 3. N = SSLND set value + 1 (1 to 8)
Note 4. The data output delay time may become longer than half a cycle of the RSPCK clock depending on the bit rate setting. Be sure to satisfy the conditions required for the electrical characteristics of the master device.

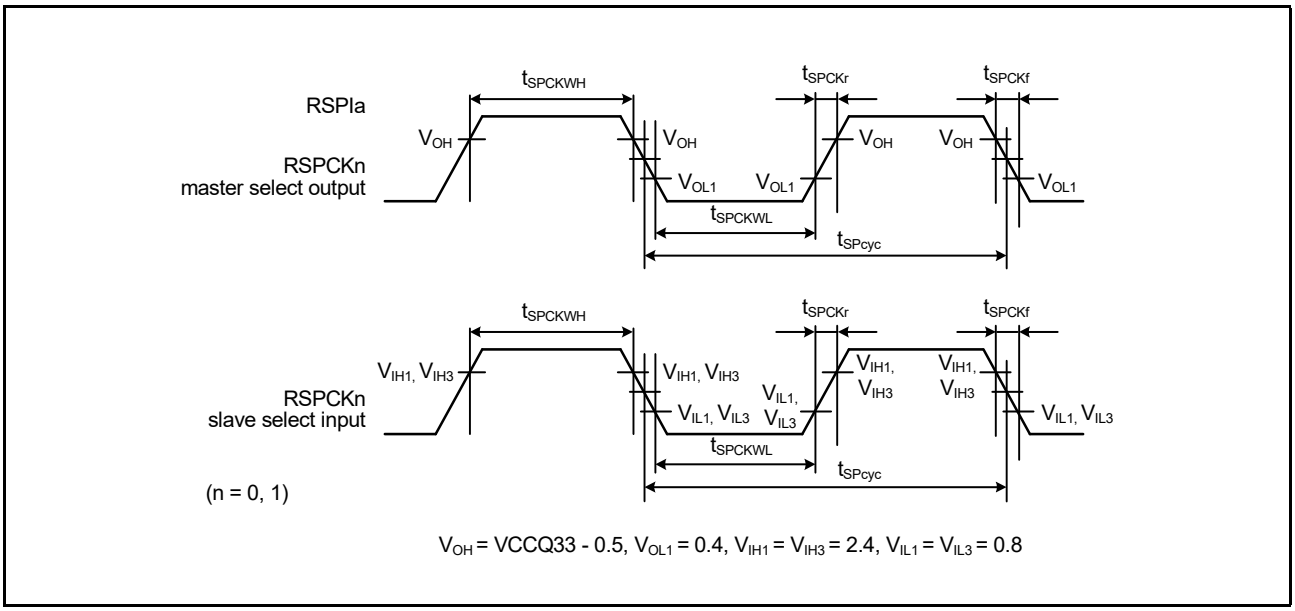


Figure 33.13 RSPIa Clock Timing

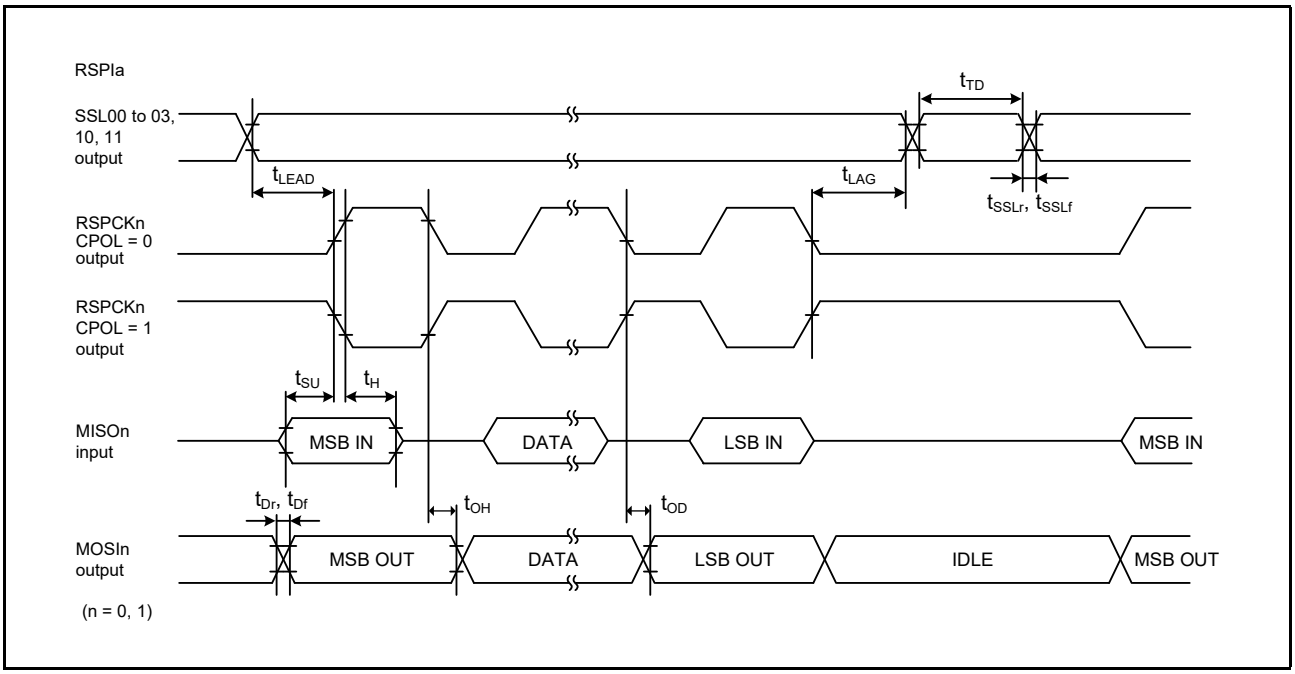


Figure 33.14 RSPIa Timing (Master, CPHA = 0)

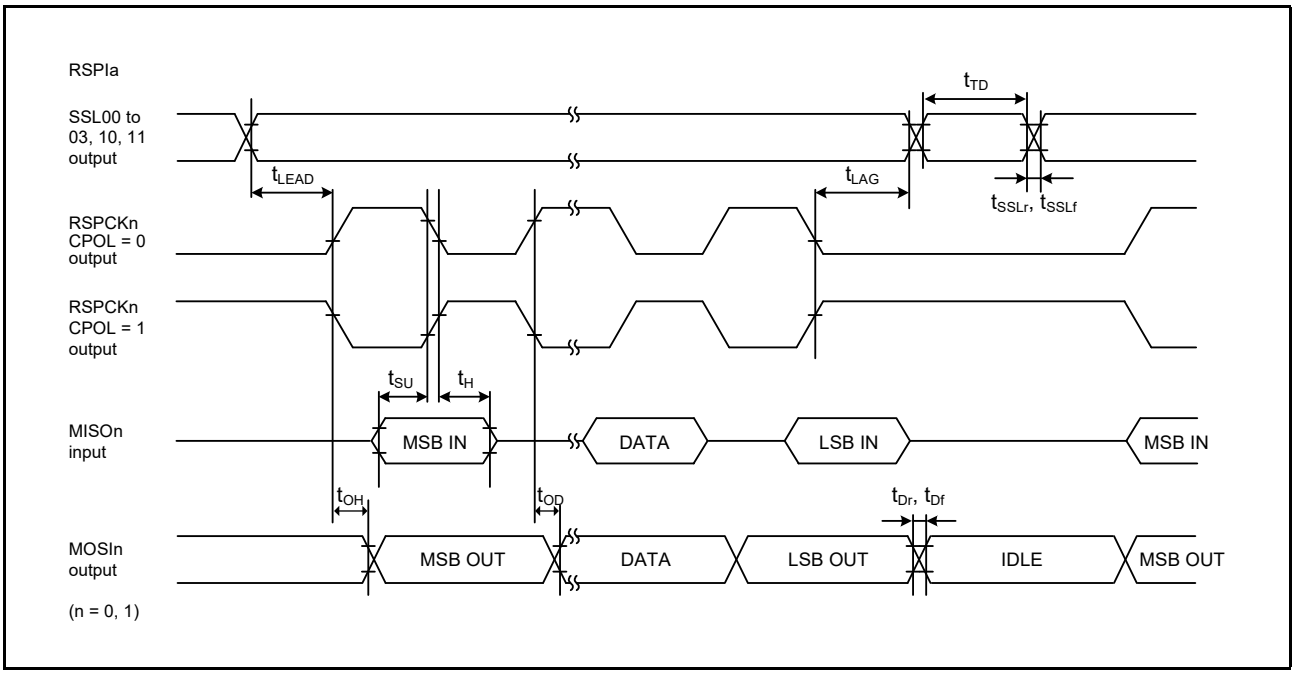


Figure 33.15 RSPIa Timing (Master, CPHA = 1)

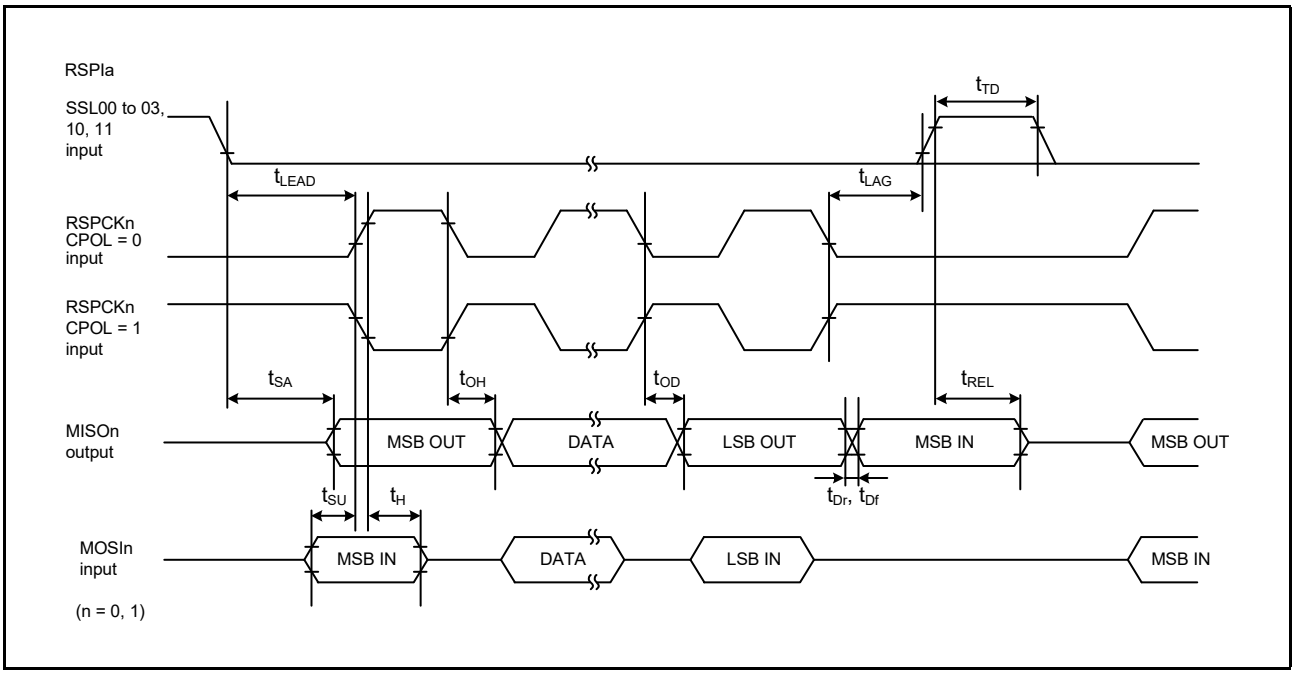


Figure 33.16 RSPI Timing (Slave, CPHA = 0)

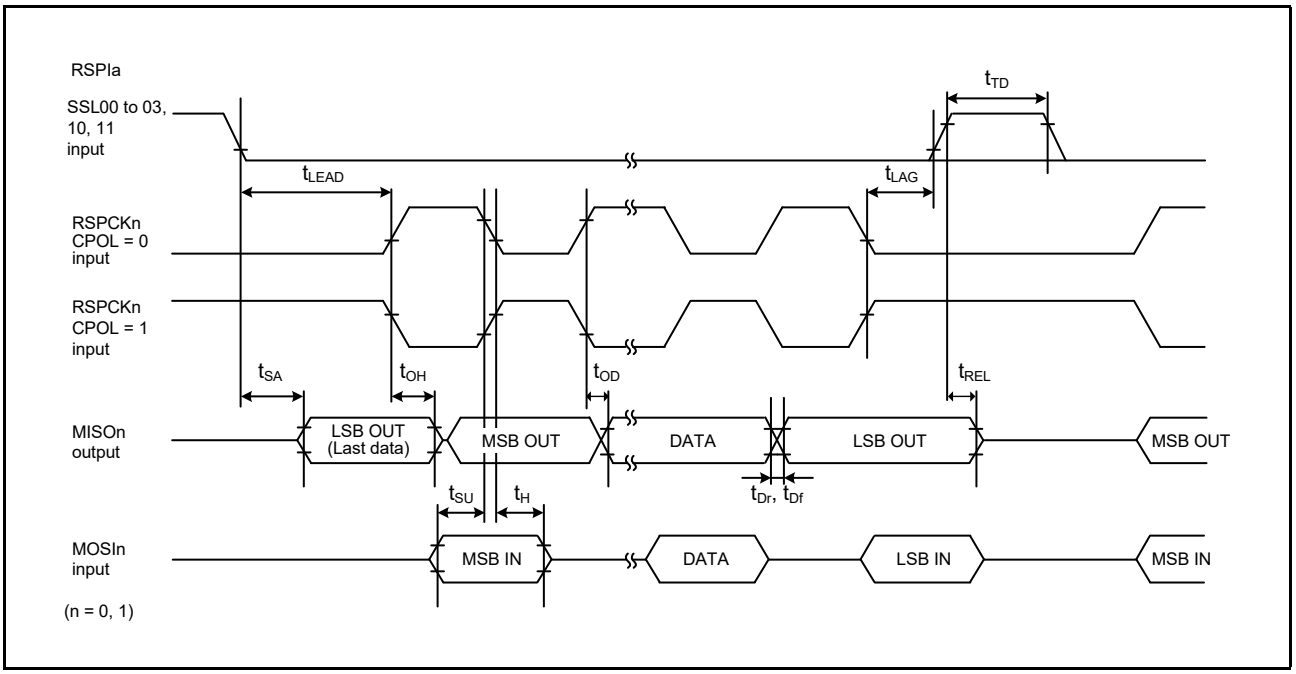


Figure 33.17 RSPI Timing (Slave, CPHA = 1)

33.4.3.5 SPIBSC Timing

Table 33.19 SPIBSC Timing

Output load conditions: V_{OH} = VCCQ33 × 0.5, V_{OL1} = VCCQ33 × 0.5, C = 30 pF

Item		Symbol	min	max	Unit*1	Test Conditions
SPIBSC	SPBCLK clock cycle	t_{SPBcyc}	2	4080	t_{PAcyc}	Figure 33.18
	SPBCLK high level pulse width	t_{SPBWH}	0.45	0.55	t_{SPBcyc}	
	SPBCLK low level pulse width	t_{SPBWL}	0.45	0.55	t_{SPBcyc}	
	Data input setup time	t_{SU}	3.5	—	ns	Figure 33.19, Figure 33.20, Figure 33.21
	Data input hold time	t_{H}	0.5	—	ns	
	SSL setup time	t_{LEAD}	1 × t_{SPBcyc} – 3	8 × t_{SPBcyc}	ns	
	SSL hold time	t_{LAG}	1.5 × t_{SPBcyc}	8.5 × t_{SPBcyc} + 3	ns	
	Continuous transfer delay time	t_{TD}	1	8	t_{SPBcyc}	
	Data output delay time	t_{OD}	—	3.6	ns	
	Data output hold time	t_{OH}	– 1	—	ns	
	Data output buffer on time	t_{BON}	—	3.6	ns	Figure 33.22, Figure 33.23, Figure 33.24
	Data output buffer off time	t_{BOFF}	– 7	0	ns	

Note 1. t_{PAcyc}: PCLKA cycle

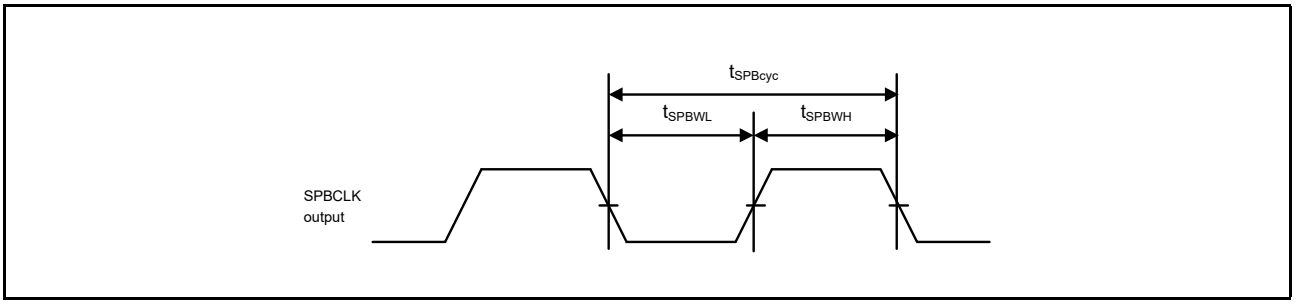


Figure 33.18 SPIBSC Clock Timing

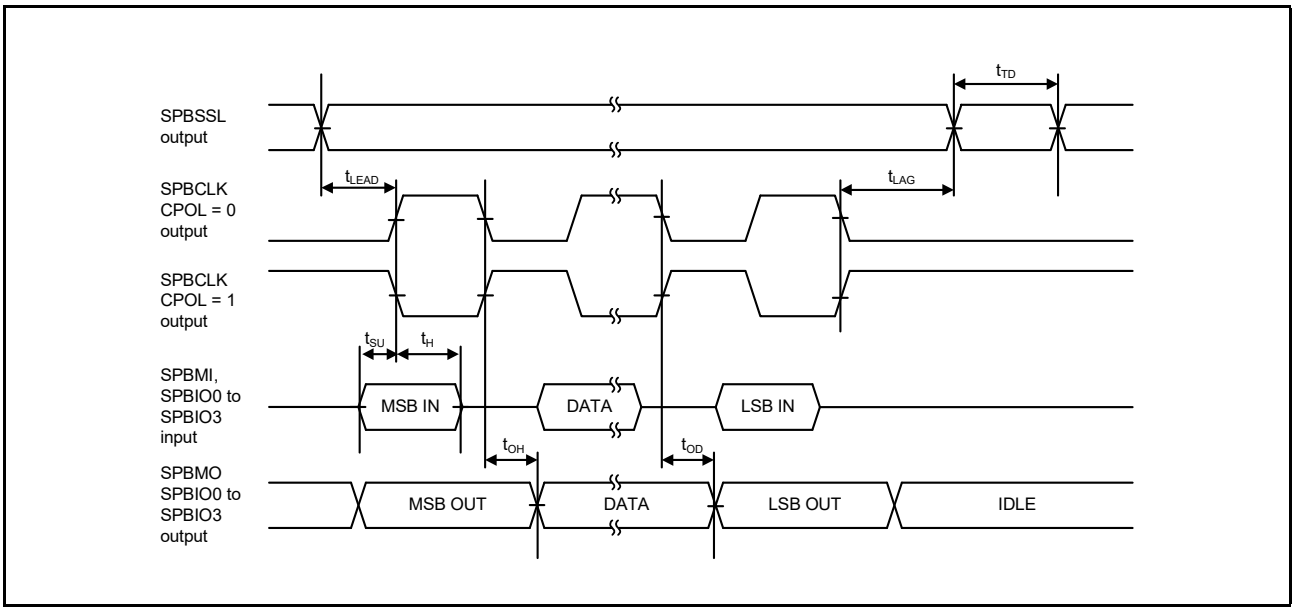


Figure 33.19 SPIBSC Transmit/Receive Timing (CPHAT = 0, CPHAR = 0)

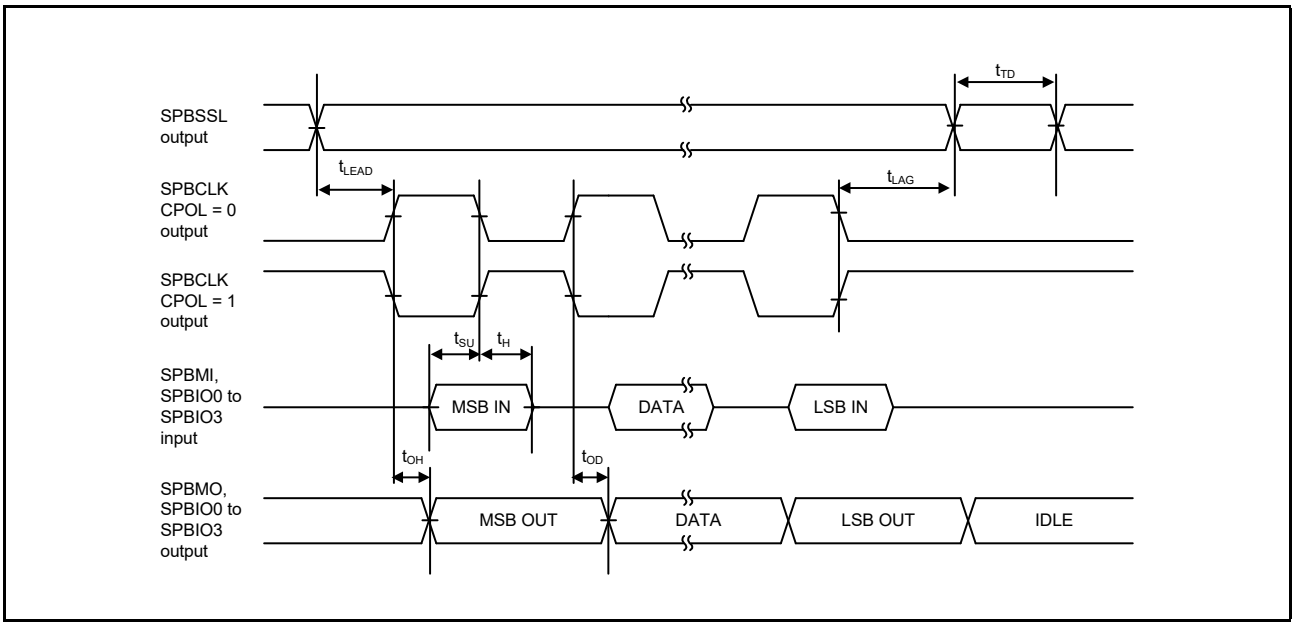


Figure 33.20 SPIBSC Transmit/Receive Timing (CPHAT = 1, CPHAR = 1)

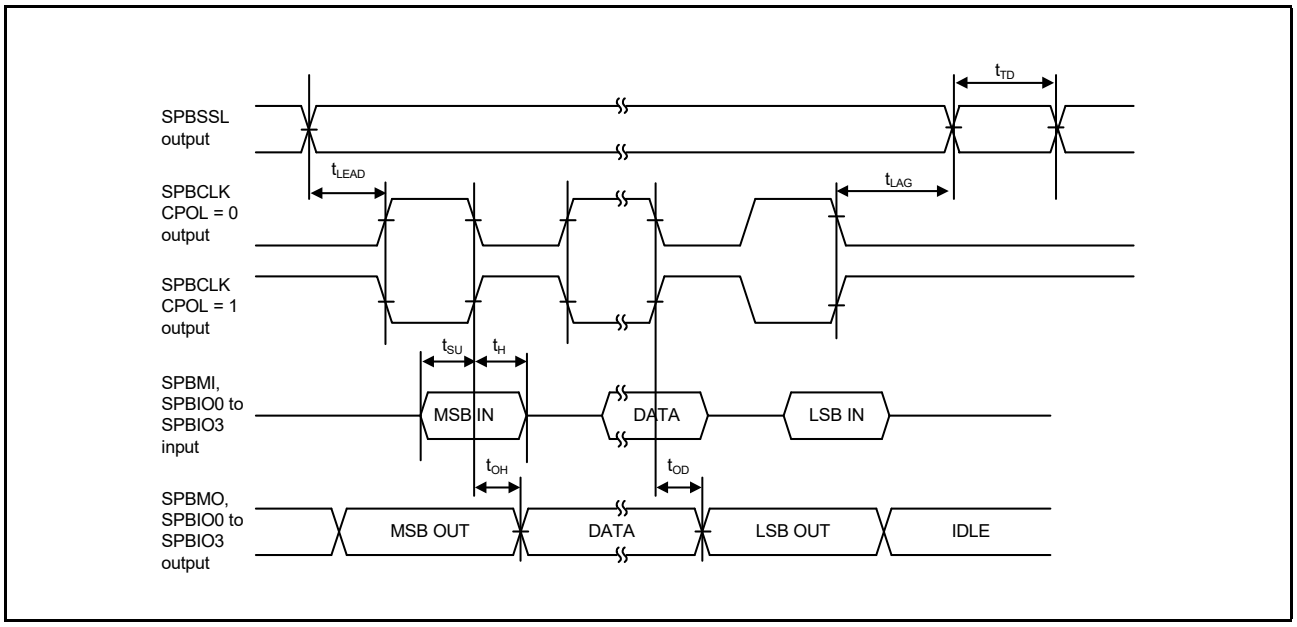


Figure 33.21 SPIBSC Transmit/Receive Timing (CPHAT = 0, CPHAR = 1)

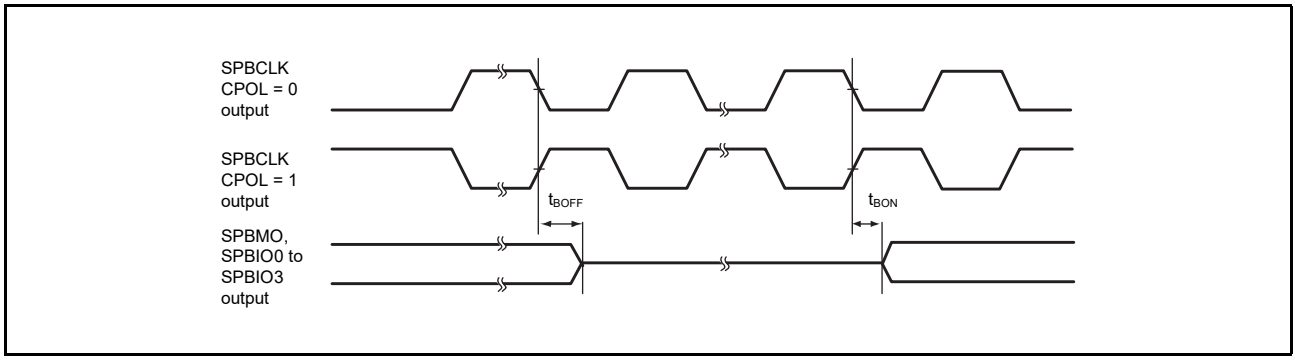


Figure 33.22 SPIBSC Buffer On/Off Timing (CPHAT = 0, CPHAR = 0)

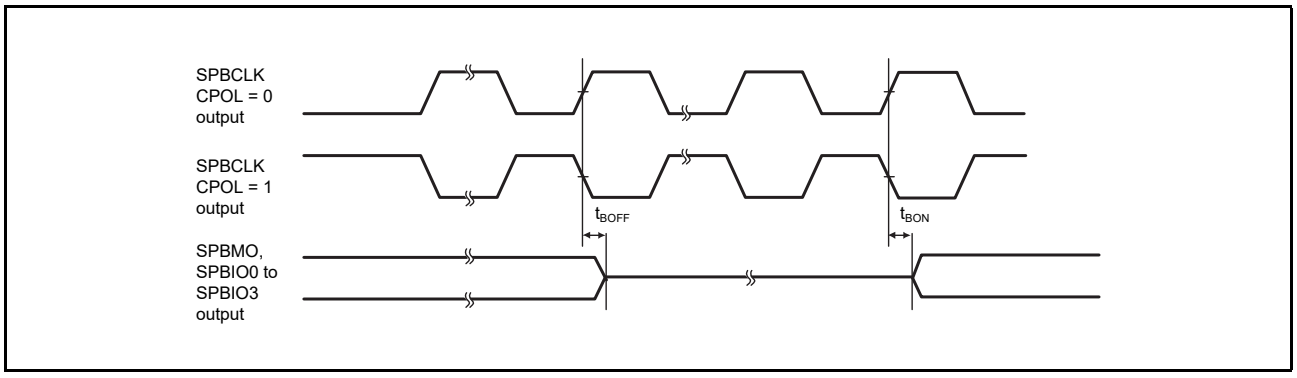


Figure 33.23 SPIBSC Buffer On/Off Timing (CPHAT = 1, CPHAR = 1)

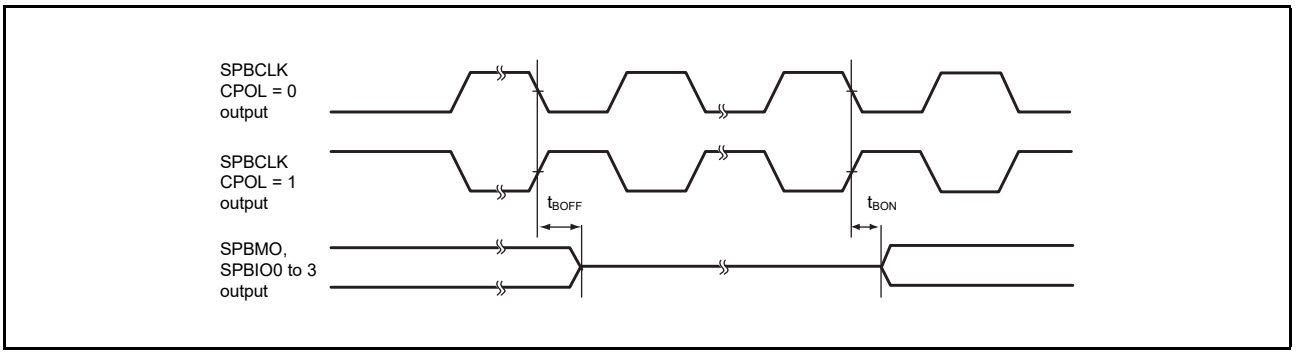


Figure 33.24 SPIBSC Buffer On/Off Timing (CPHAT = 0, CPHAR = 1)

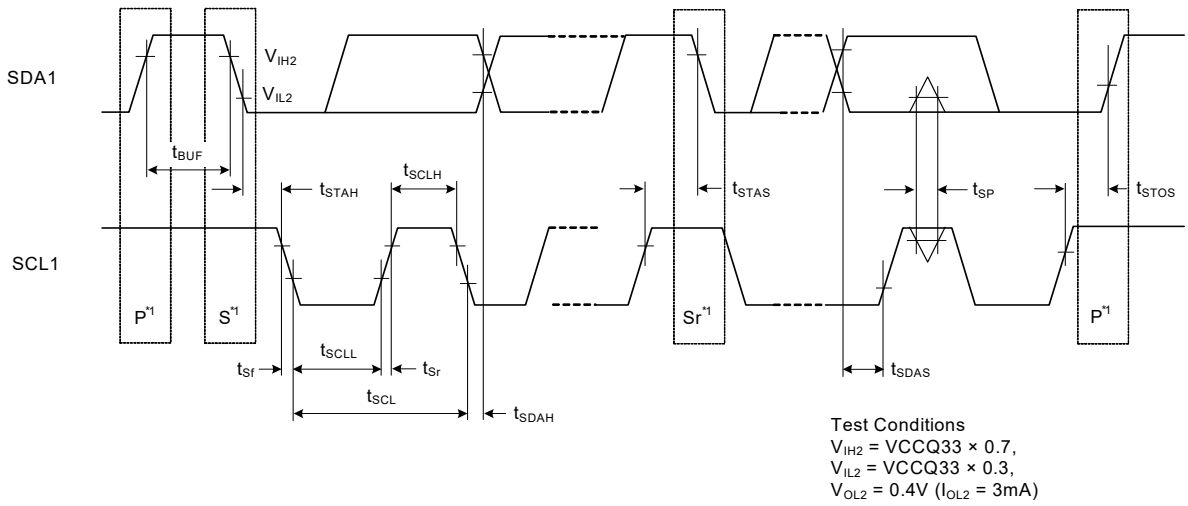
33.4.3.6 RIICa Timing

Table 33.20 RIICa Timing

Output load conditions: V_{OL2} = 0.4 V, I_{OL2} = 3 mA

Item		symbol	min*2	max*2	Unit*1	Test Conditions
RIICa (Standard-mode)	SCL input cycle time	t_{SCL}	6(12) × t_{IICcyc} + 1300	—	ns	Figure 33.25
	SCL input high pulse width	t_{SCLH}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL input low pulse width	t_{SCLL}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL, SDA input rising time	t_{sr}	—	1000	ns	
	SCL, SDA input falling time	t_{sf}	—	300	ns	
	SCL, SDA input spike pulse removal time	t_{SP}	0	1(4) × t_{IICcyc}	ns	
	SDA input bus free time	t_{BUF}	3(6) × t_{IICcyc} + 300	—	ns	
	Start condition input hold time	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Restart condition input setup time	t_{STAS}	1000	—	ns	
	Stop condition input setup time	t_{STOS}	1000	—	ns	
	Data input setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data input hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load	C_b	—	400	pF	
RIICa (Fast-mode)	SCL input cycle time	t_{SCL}	6(12) × t_{IICcyc} + 600	—	ns	
	SCL input high pulse width	t_{SCLH}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL input low pulse width	t_{SCLL}	3(6) × t_{IICcyc} + 300	—	ns	
	SCL, SDA input rising time	t_{sr}	—*4	300	ns	
	SCL, SDA input falling time	t_{sf}	—*4	300	ns	
	SCL, SDA input spike pulse removal time	t_{SP}	0	1(4) × t_{IICcyc}	ns	
	SDA input bus free time	t_{BUF}	3(6) × t_{IICcyc} + 300	—	ns	
	Start condition input hold time	t_{STAH}	t_{IICcyc} + 300	—	ns	
	Restart condition input setup time	t_{STAS}	300	—	ns	
	Stop condition input setup time	t_{STOS}	300	—	ns	
	Data input setup time	t_{SDAS}	t_{IICcyc} + 50	—	ns	
	Data input hold time	t_{SDAH}	0	—	ns	
	SCL, SDA capacitive load*3	C_b	—	400	pF	

Note 1. t_{IICcyc}: RIICa internal reference clock (IICφ) cycle
Note 2. The value out of parentheses is applicable when the value of the ICMR3.NF[1:0] bits is 00b while the digital filter is enabled by the setting ICFER.NFE = 1. The value within parentheses is applicable when the value of the ICMR3.NF[1:0] bits is 11b while the digital filter is enabled by the setting ICFER.NFE = 1.
Note 3. Cb is the total capacitance of the bus lines.
Note 4. The minimum values are not specified for trs and tsf in Fast-mode.



Note 1. S, P, and Sr indicate the following conditions.
S: Start condition
P: Stop condition
Sr: Restart condition

Figure 33.25 RIICa Bus Interface Input/Output Timing

33.4.3.7 CAN Interface Timing

Table 33.21 CAN Interface Timing

Item	Symbol	min	max	Unit	Test Conditions
Internal delay time	tnode	—	100	ns	Figure 33.26
Transmission rate		—	1	Mbps	

Internal delay time (tnode) = Internal transmission delay time (toutput) + Internal reception delay time (tinput)

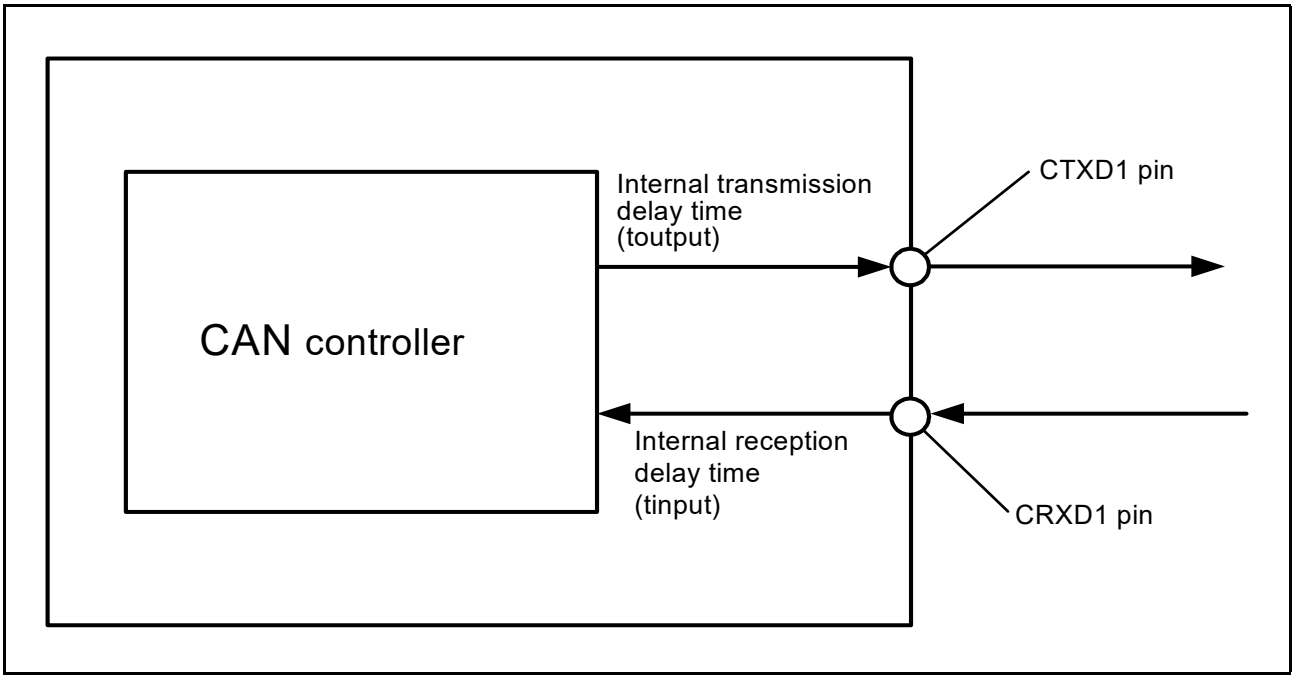


Figure 33.26 CAN Interface Conditions

33.4.3.8 ESC Timing

Table 33.22 ESC Timing

Output load conditions: V_{OH} = VCCQ33 × 0.5, V_{OL1} = VCCQ33 × 0.5, C = 30 pF

Item		Symbol	min	max	Unit	Test Conditions
ESC (MII)	ETHn_TXC cycle time	t_{Tcyc}	40	—	ns	—
	ETHn_TXEN output delay time	t_{TENd}	0	25	ns	Figure 33.27
	ETHn_TXD0 to ETHn_TXD3 output delay time	t_{MTDd}	0	25	ns	
	ETHn_RXC cycle time	t_{TRcyc}	40	—	ns	—
	ETHn_RXDV setup time	t_{RDVs}	10	—	ns	Figure 33.28
	ETHn_RXDV hold time	t_{RDVh}	10	—	ns	
	ETHn_RXD0 to ETHn_RXD3 setup time	t_{MRDs}	10	—	ns	
	ETHn_RXD0 to ETHn_RXD3 hold time	t_{MRDh}	10	—	ns	
	ETHn_RXER setup time	t_{RERs}	10	—	ns	Figure 33.29
	ETHn_RXER hold time	t_{RERh}	10	—	ns	

n = 0, 1

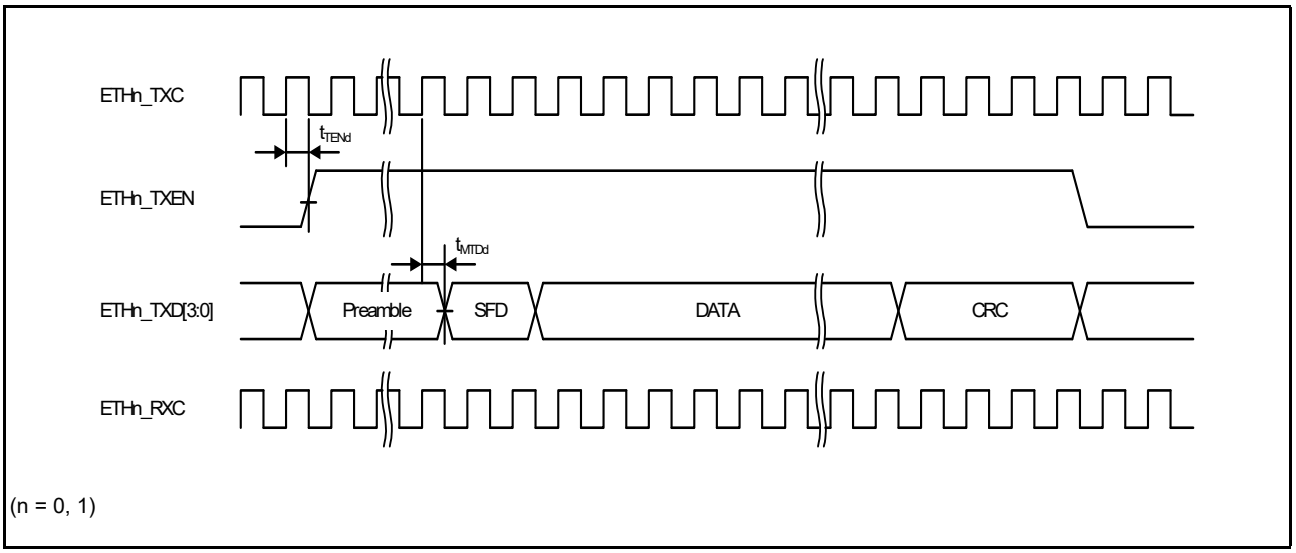


Figure 33.27 MII Transmission Timing

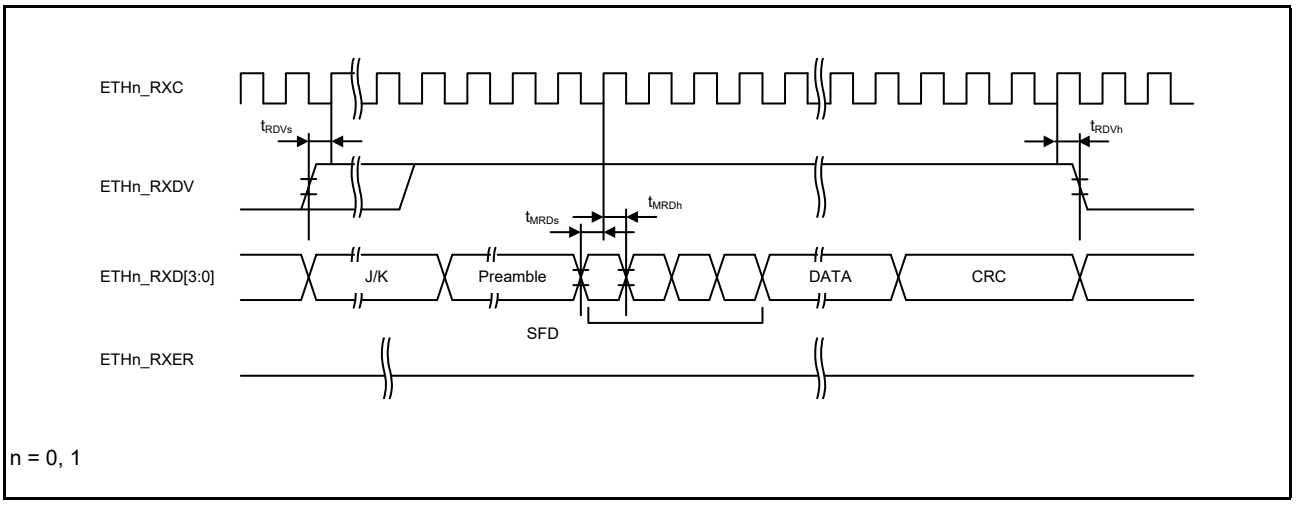


Figure 33.28 MII Reception Timing (Normal Operation)

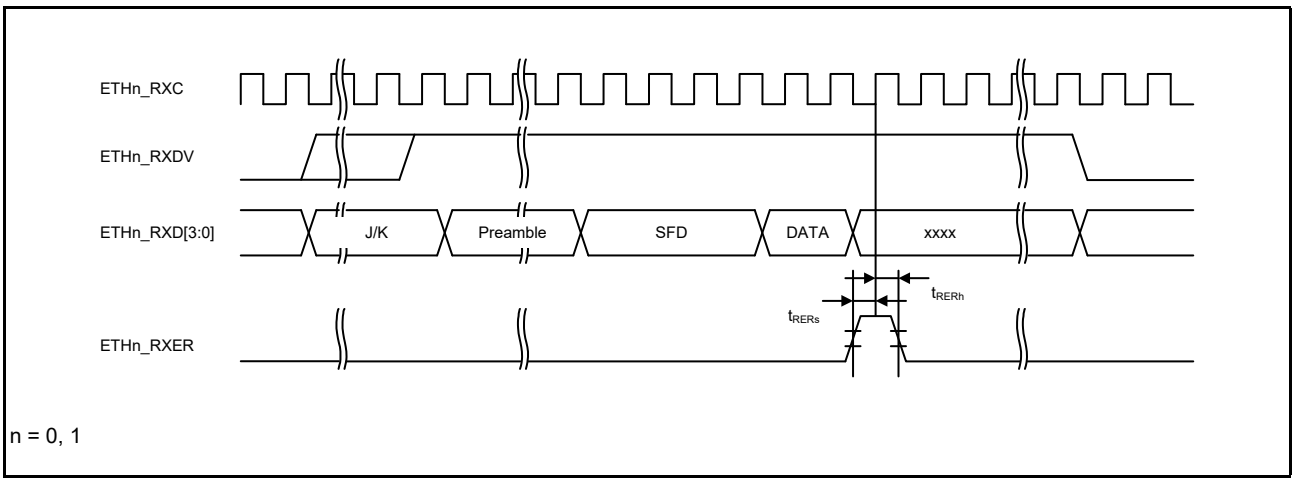


Figure 33.29 MII Reception Timing (Error Occurrence)

33.4.3.9 Serial Management Interface

Table 33.23 Serial Management Interface

Output load conditions: V_{OH} = VCCQ33 × 0.5, V_{OL1} = VCCQ33 × 0.5, C = 30 pF

Item		Symbol	min	max	Unit	Test Conditions
MDIO	ETH_MDC output cycle	t_{MDC}	80	—	ns	Figure 33.30
	ETH_MDIO input setting time (to ETH_MDC↑)	t_{SMDIO}	10	—	ns	
	ETH_MDIO input hold time (to ETH_MDC↑)	t_{HMDIO}	0	—	ns	
	ETH_MDIO output delay time (to ETH_MDC↓)	t_{DMDIO}	—	20	ns	

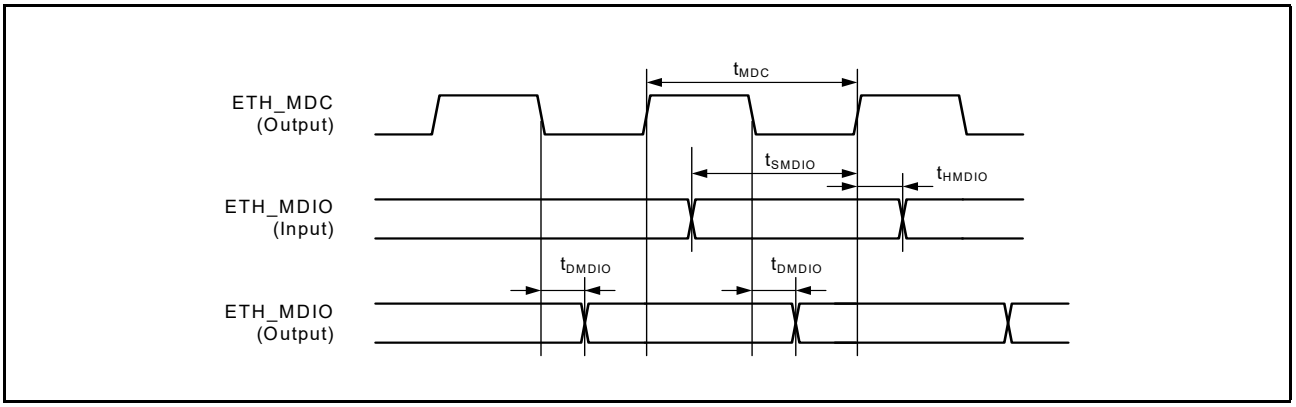


Figure 33.30 Serial Management Access Timing

33.5 USB Characteristics

- Conditions: VDD = PLLVDD0 = PLLVDD1 = DVDD_USB = 1.14 to 1.26 V,
  VCCQ33 = VDD33_USB = 3.0 to 3.6 V
  VSS = PLLVSS0 = PLLVSS1 = VSS_USB = 0 V,
  Tj = –40 to 125 °C

Table 33.24 On-chip USB Full-Speed Characteristics (USB_DP, USB_DM Pin Characteristics)

Item	Symbol	min	typ	max	Unit	Test Conditions
Rising time	t_{FR}	4	—	20	ns	Figure 33.31
Falling time	t_{FF}	4	—	20	ns	
Rising/falling time ratio	t_{FR} / t_{FF}	90	—	111.11	%	t_{FR} / t_{FF}

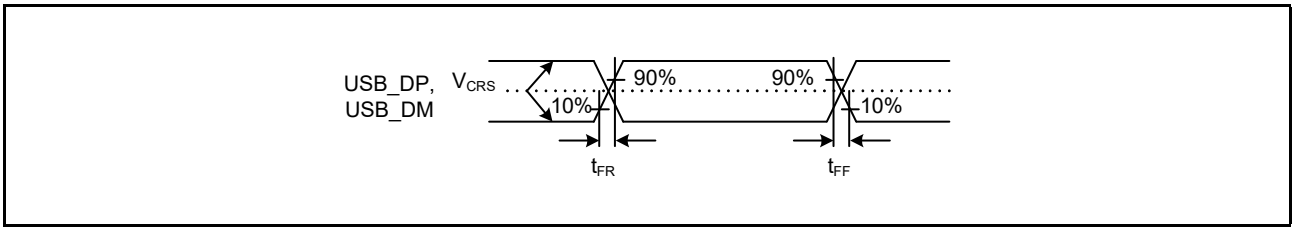


Figure 33.31 USB_DP, USB_DM Output Timing (Full Speed)

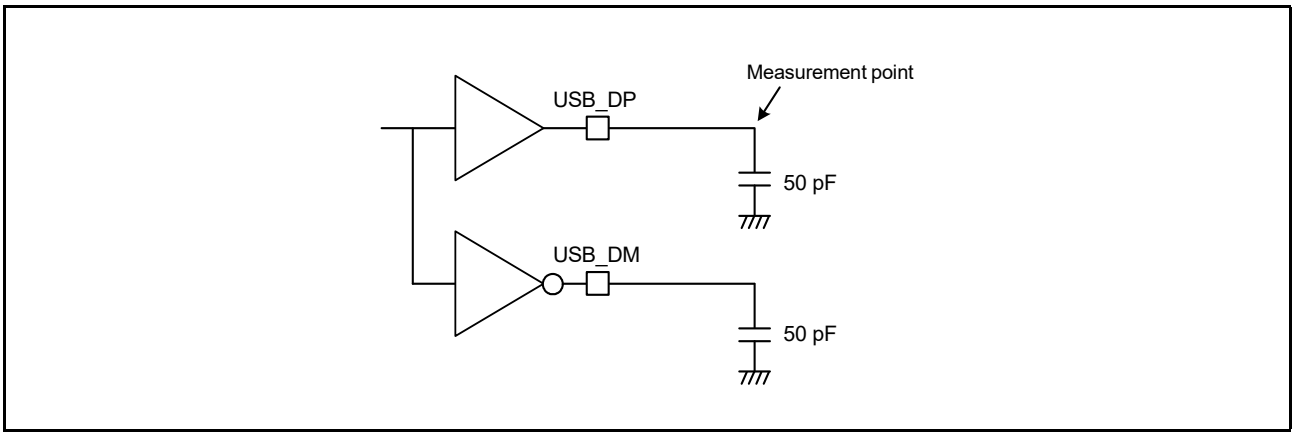


Figure 33.32 Measurement Circuit (Full Speed)

Table 33.25 On-chip USB High-Speed Characteristics (USB_DP, USB_DM Pin Characteristics)

Item		Symbol	min	typ	max	Unit	Test Conditions
AC characteristics	Rising time	t_{HSR}	500	—	—	ps	Figure 33.33
	Falling time	t_{HSF}	500	—	—	ps	
	Output resistance	Z_{HSDRV}	40.5	—	49.5	Ω	

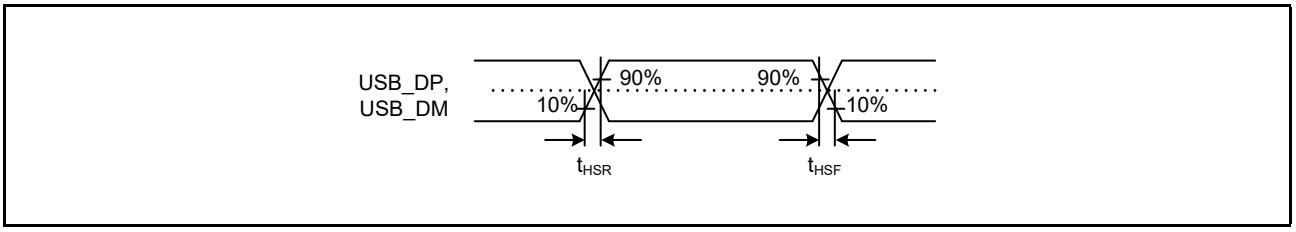


Figure 33.33 USB_DP, USB_DM Output Timing (High Speed)

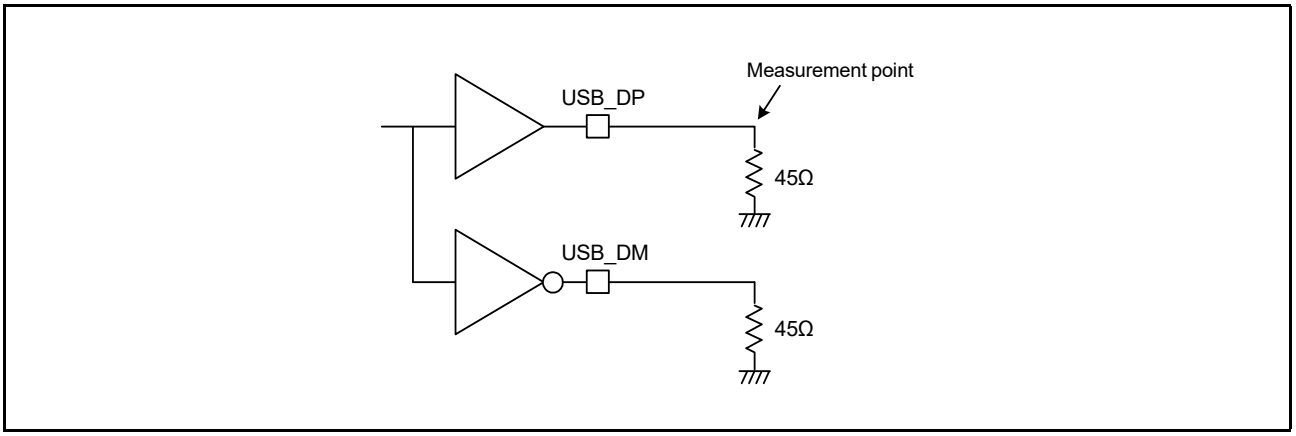


Figure 33.34 Measurement Circuit (High Speed)

33.6 Oscillation Stop Detection Timing

Table 33.26 Oscillation Stop Detection Circuit Characteristics

Item	Symbol	min	typ	max	Unit	Test Conditions
Clock switching time	t_{dr}	—	—	1	ms	Figure 33.35

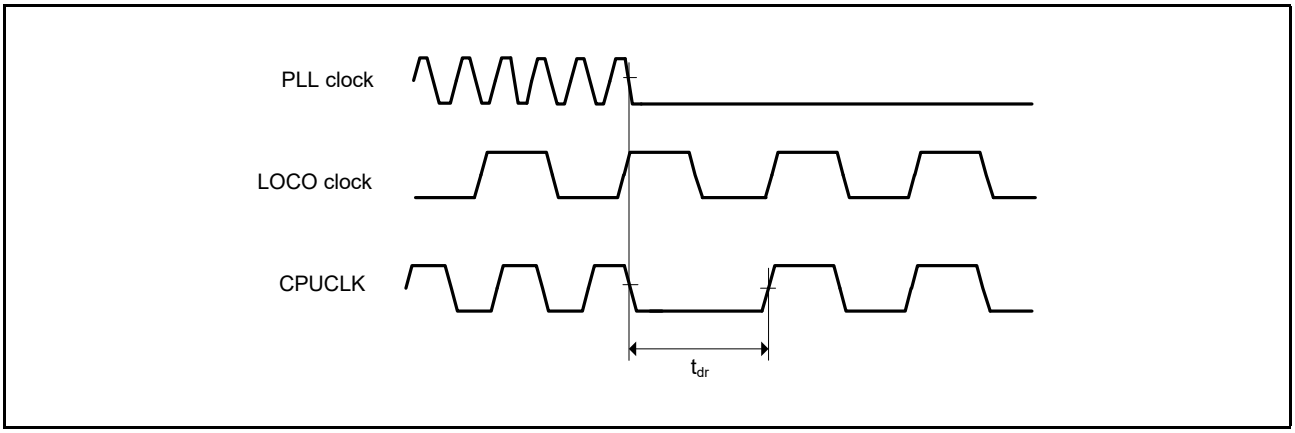


Figure 33.35 Oscillation Stop Detection Timing

33.7 Debug Interface Timing

Table 33.27 Debug Interface Timing

Output load conditions: V_{OH} = VCCQ33 – 0.5 V, V_{OL1} = 0.4 V

Item	Symbol	Min.	Max.	Unit	Reference Figure
TCK cycle time	t_{TCKcyc}	30	—	ns	Figure 33.36
TCK high pulse width	t_{TCKH}	0.4	0.6	t_{TCKcyc}	
TCK low pulse width	t_{TCKL}	0.4	0.6	t_{TCKcyc}	
TDI setup time	t_{TDIS}	5	—	ns	Figure 33.37 Output load: 30 pF
TDI hold time	t_{TDIH}	5	—	ns	
TMS/SWDIO setup time	t_{TMSS}	5	—	ns	
TMS/SWDIO hold time	t_{TMSH}	5	—	ns	
SWDIO delay time	t_{SWDO}	—	15	ns	
TDO delay time	t_{TDOD}	—	15	ns	
Capture register setup time	t_{CAPTS}	5	—	ns	Figure 33.38
Capture register hold time	t_{CAPTH}	5	—	ns	
Update register delay time	$t_{UPDATED}$	—	15	ns	
Trace clock cycle	t_{TCYC}	26.6	—	ns	Figure 33.39 Output load: 15 pF
Trace data delay time	t_{TDT}	0.25 × t_{TCYC} – 2	0.25 × t_{TCYC} + 2	ns	

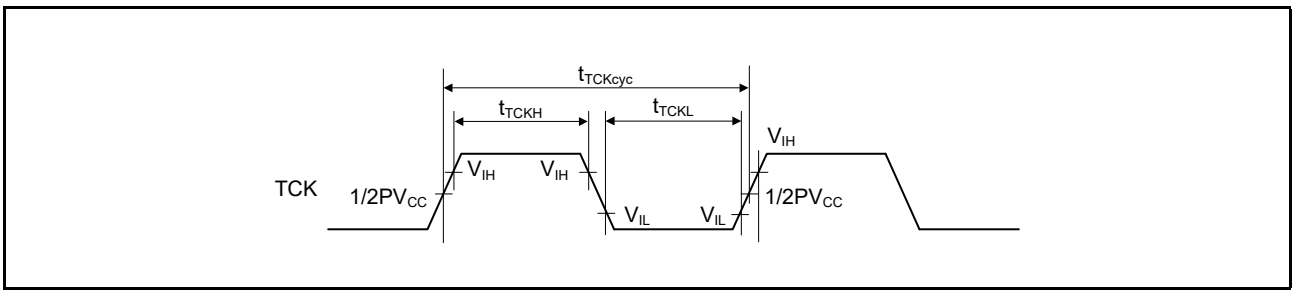


Figure 33.36 TCK Input Timing

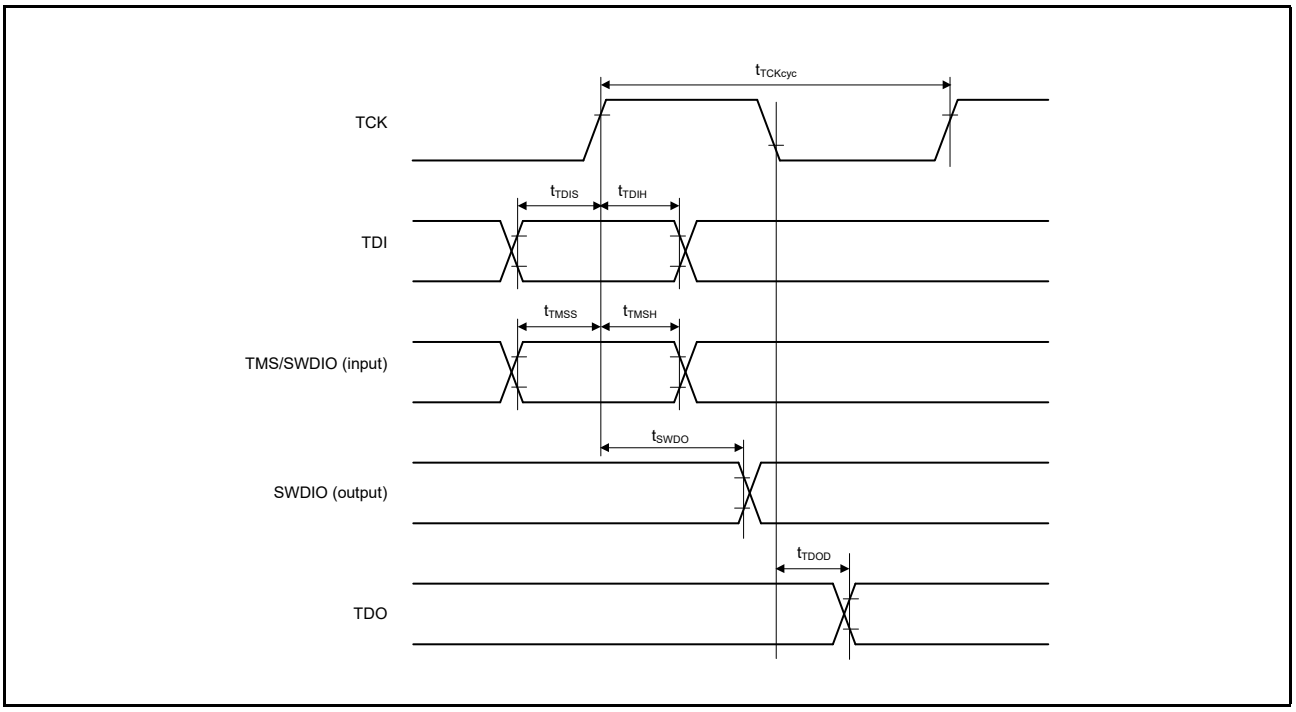


Figure 33.37 Data Transfer Timing

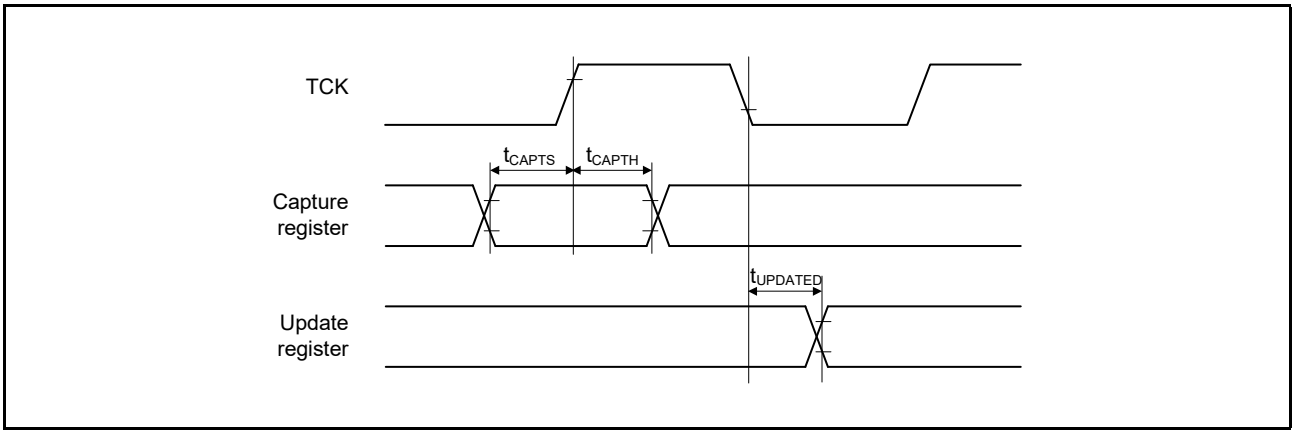


Figure 33.38 Boundary Scan Input/Output Timing

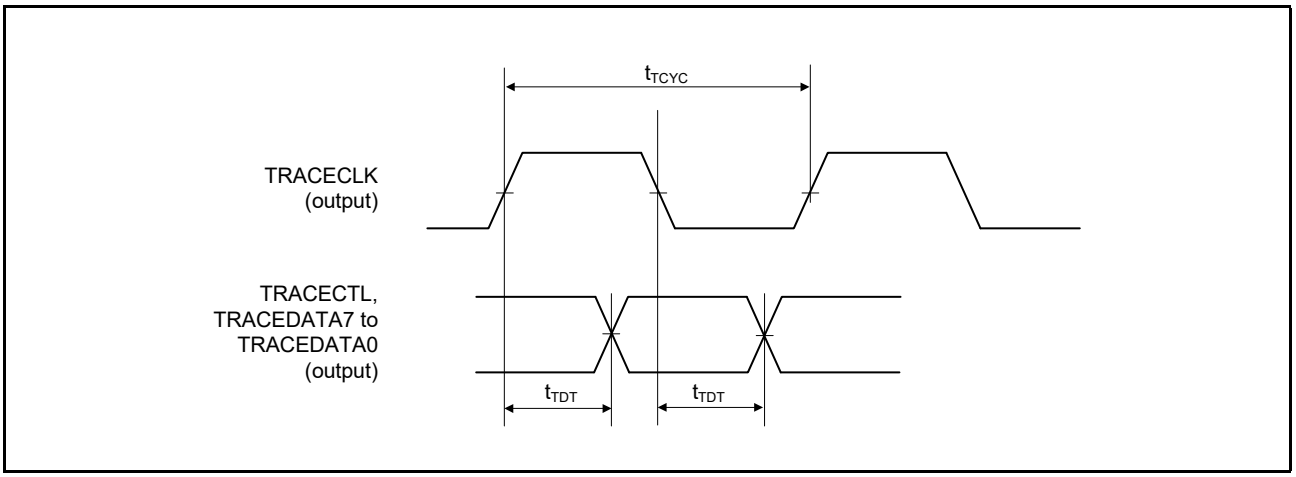
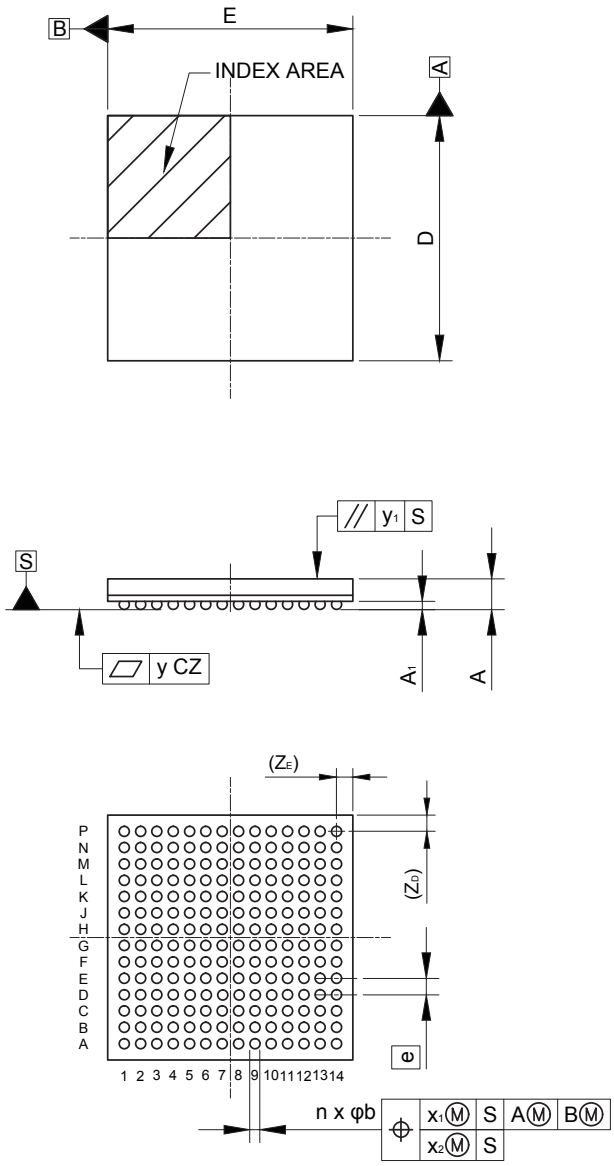


Figure 33.39 Trace Interface Timing

Appendix 1.Outer Dimensions Diagram

JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-LFBGA196-12x12-0.80	PLBG0196GA-B	0.42



Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
D	11.92	12.00	12.08
E	11.92	12.00	12.08
A	—	—	1.60
A_1	0.35	0.40	0.45
e	—	0.80	—
b	0.45	0.50	0.55
x_1	—	—	0.15
x_2	—	—	0.08
y	—	—	0.10
y_1	—	—	0.20
n	—	196	—
Z_D	—	0.80	—
Z_E	—	0.80	—

Figure A 196-Pin FBGA (PLBG0196GA-B)

REVISION HISTORY — EC-1 User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
0.50	Aug. 03, 2016	—	First edition, issued
1.00	Sep. 30, 2016	1. Overview	
		37	Table 1.1 Outline of Specifications (2 / 3), Communication function, CAN module (RSCAN), Message buffers: Number of channels changed, description partially deleted.
		3. Operating Modes	
		64	3.4.4.1 Operation Settings in SPI Boot Mode: Description added.
		12. Interrupt Controller (ICUA)	
		183	Figure 12.1 Block Diagram of Interrupt Controller: Description partially deleted.
		242	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISCi Bit (IRQ Interrupt Request Service Flag) (i = 1 to 63): Bit description partially amended.
		243	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISCi Bit (IRQ Interrupt Request Service Flag) (i = 64 to 127): Bit description partially amended.
		244	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISCi Bit (IRQ Interrupt Request Service Flag) (i = 128 to 191): Bit description partially amended.
		245	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISCi Bit (IRQ Interrupt Request Service Flag) (i = 192 to 255): Bit description partially amended.
		246	12.4.2.15 Interrupt Service Current Register n (ISCn) (n = 0 to 9), ISCi Bit (IRQ Interrupt Request Service Flag) (i = 256 to 300): Bit description partially amended.
		247	12.4.2.16 Interrupt Address Store Register 0 (VADn) (n = 1 to 255), Interrupt Address Store Register 1 (VADn) (n = 256 to 300), Bit symbol amended (VADn to VADi).
		14. DMA Controller (DMACAa)	
		308	14.3.1.1 Register Mode (1) Operation Flow in Register Mode <Register Mode Flow Description>, 1. Channel Setting: Description partially deleted.
		322	14.3.1.2 Link Mode (3) Descriptor settings, Notes on descriptors: DREQn pin input deleted.
		338	Figure 14.18 Stop Timing for Responding to a Bus Error: Waveforms of DREQ[0] and DACK[0] deleted.
		22. EtherCAT Slave Controller	
		504	Table 22.4 MAC Function Selection Method: One of the values for MAC[2:0] amended.
		509	22.3.2.1 MIIM Register (GMAC_MIIM): Description of b26 amended.
		27. CAN Interface (RSCAN)	
		925	27.1.1 Functional Overview: Buffer specification amended.
		28. Serial Peripheral Interface (RSPIa)	
		1087	28.2.5 RSPI Data Register (SPDR): Address RSPI0.SPDR A006 8004h and RSPI1.SPDR A006 8404h amended.
		29. SPI Multi I/O Bus Controller (SPIBSC)	
		1161, 1162	29.2.1 Common Control Register (CMNCR): Descriptions for b9-b8, b13-b12, b15-b14, b17-b16, b19-b18, b21-b20, and b23-b22 amended.
		32. Error Control Module (ECM)	
		1257	32.2.23 ECM Protection Command Register (ECMPCMD1): Bit symbol amended.
1.10	May 22, 2017	3. Operating Modes	
		59	3.1 Overview: The description, modified
		69	3.4.7.2 Serial Flash Memory in SPI Boot Mode, added

Rev.	Date	Description	
		Page	Summary
1.10	May 22, 2017	5. I/O Registers	
		94	Table 5.1 List of I/O Registers (Address Order) (22/56): A006 005Ch / DCP configuration register (DCPCFG), deleted
		7. Clock Generation Circuit	
		136	Table 7.2 Specifications of Clock Generation Circuit (Internal Clock): The clock source of Ethernet clock E (ETCLKE), modified
		9. Low-Power Consumption Function	
		161	9.2.1 Module Stop Control Register A (MSTPCRA): The description, added
		162	9.2.2 Module Stop Control Register B (MSTPCRB): The description, added
		164	9.2.3 Module Stop Control Register C (MSTPCRC): The description, added
		165	9.2.4 Module Stop Control Register E (MSTPCRE): The description, added
		165	9.2.5 Module Stop Control Register F (MSTPCRF): The description, added
		166, 167	9.3.1 Module-Stop Function: The description, modified. Procedures, Example code, and Table 9.3, added.
		10. Debugging Interface	
		179	Figure 10.7 Example of Connection Circuit of an Emulator That Can Drive the nTRST Output to High: Timing charts when the emulator is connected and not connected, modified
		12. Interrupt Controller (ICUA)	
		260	Table 12.3 Cortex-R4F/DMACAa Interrupt Vector Table: Note 4, added
		263	Figure 12.7 Register Rewrite Flow: The processing, modified
		13. Internal Buses	
		273	Table 13.1 Specifications of Internal Buses: The contents of external serial flash bus, modified (PCLKD → ICLK)
		273	Figure 13.1 Bus Configuration: Serial Flash (PCLKD) → Serial Flash, modified
		14. DMA Controller (DMACAa)	
		351	14.6 Usage Notes: The note, added
		16. I/O Ports	
		381	Table 16.3 Handling of Unused Pins: The handling of the TRST# pin, modified
		22. EtherCAT Slave Controller	
		588	22.4.3 Initial Settings, added
		24. USB 2.0 HS Function Module (USBf)	
		783	Table 24.23 Pipe Settings: The DCPCFG register, deleted
		785	24.5.3 Pipe Control Register Switching Procedures: The DCPCFG register, deleted
		25. Serial Communications Interface with FIFO (SCIFA)	
		All	The symbol of the serial extended mode register, corrected (SMER → SEMR)
		831	Figure 25.5 Example of Transmit Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected): ICU.IRn → ICU.RAISn
		834	Figure 25.9 Example of SCIFA Receive Operation in Asynchronous Mode (8-Bit Data with Parity and One Stop Bit when LSB-First Transfer is Selected): ICU.IRn → ICU.RAISn
		839	Figure 25.14 Example of SCIFA Transmit Operation in Clock Synchronous Mode (when LSB-First Transfer is Selected): ICU.IRn → ICU.RAISn
		841	Figure 25.16 Example of SCIFA Receive Operation (when LSB-First Transfer is Selected): ICU.IRn → ICU.RAISn
		26. I^2C Bus Interface (RIICa)	
		—	26.12.1 Buffer Operation for TXI and RXI Interrupts, deleted
		—	26.14.2 Notes on Starting Transfer, deleted
		29. SPI Multi I/O Bus Controller (SPIBSC)	
		1160	Table 29.1 SPIBSC Specifications: The contents of the bit rate, modified (PCLKA → ICLK)

Rev.	Date	Description	
		Page	Summary
1.10	May 22, 2017	1161	Figure 29.1 Block Diagram of SPIBSC: PCLKA → ICLK, modified
		1167	29.2.3, (1) Bit Rate: PCLKA → ICLK, modified
		1167	Table 29.3 Relationship between SPBR[7:0] and BRDV[1:0] Settings: Bit Rate, PCLKA → ICLK, modified. Setting of bits SPBR[7:0] and BRDV[1:0]: 0,1, 0,2, and 0,3, added.
		1186	Figure 29.4 Data Alignment in External Address Space Read Mode, modified
		1186	Figure 29.5 Data Alignment in SPI Operating Mode, modified
		32. Error Control Module (ECM)	
		1257	The functional description of the ECMCLSSE102 to ECMCLSSE108 bits in the table of bits, modified (the bit error of the ECMmESSTR register, corrected)
		33. Electrical Characteristics	
		1395	Figure 33.21 SPIBSC Transmit/Receive Timing (CPHAT = 0, CPHAR = 1): Modified
1.20	Apr. 03, 2018	All	"Cortex-R4F" changed to "Cortex-R4"
		1. Overview	
		36	1.1 Outline of Specifications: “ARM Cortex®-R4F processor” changed to “ARM Cortex®-R4 processor with FPU”
		42	Table 1.4 Pin Functions (1 / 3): CTS0# to CTS2# pins: I/O and functional description changed; RTS0# to RTS2# pins: Functional description changed
		3. Operating Modes	
		63	Figure 3.3 Connection Diagram of This LSI with a Serial Flash Memory: Serial flash memory pins corrected (SI/SIO0 → SO/SIO1; SI/SIO1 → SI/SIO0)
		5. I/O Registers	
		127	Table 5.1 List of I/O Registers (Address Order) (55 / 56): PHY Port Status n Register (PHY_STATUSn) deleted
		6. Reset	
		129	Table 6.2 Targets to Be Initialized for Each Reset Type: Entries under “Reset Source” for “RSTOUT# pin output” and “ERROROUT# pin output” changed
		8. Clock Monitor Circuit (CLMA)	
		152	8.2.1 CLMAn Control Registers 0 (CLMAnCTL0): The order of the bit fields in the bit table changed
		152	8.2.2 CLMAn Compare Registers L (CLMAnCMPL): The order of the bit fields in the bit table changed
		153	8.2.3 CLMAn Compare Registers H (CLMAnCMPH): The order of the bit fields in the bit table changed
		154	8.2.5 CLMAn Protection Status Registers (CLMAnPS): The order of the bit fields in the bit table changed
		12. Interrupt Controller (ICUA)	
		184	Table 12.1 Specifications of Interrupt Controller: Note 2 changed (CR4F → CR4)
		200	12.3.3 External Pin Interrupts: Description added
		254	Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (1/7): The reset source for vector numbers 1 to 3 changed (System (CR4F) → System (CR4))
		273	12.4.5.3 Notes on Selecting Level Detection: Description modified
		273	12.4.5.5 Notes on Vector Settings, added
		274	12.4.5.1 Using Falling-Edge Detection with the NMI Pin: Moved from 12.4.5.5 to 12.5.1
		274	12.5.2 Using Falling-Edge or Rising and Falling Edge Detection for the External Pin Interrupts, added
		14. DMA Controller (DMACAa)	
		311	14.2.25 DMA Status SUS Register (DST_SUS_X (X = A or B)): Address corrected

Rev.	Date	Description	
		Page	Summary
1.20	Apr. 03, 2018	15. Event Link Controller (ELC)	
		356	Table 15.3 Correspondence between Event Signal Names Set in ELSRn.ELS[7:0] Bits and Signal Numbers: Entries under “Name of Event Signal Set in ELSRn” for ELS[7:0] bit values (23h, 24h) changed
		16. I/O Ports	
		383	Table 16.3 Handling of Unused Pins: USB_RREF pin and Note 1 added
		18. Compare Match Timer (CMT)	
		420	18.2.6 Compare Match Constant Register (CMCOR): Description of the compare-match cycle and Note 1 added
		19. Compare Match Timer W (CMTW)	
		439	19.2.5 Compare Match Constant Register (CMWCOR): Description of the compare-match cycle and Note 1 added
		22. EtherCAT Slave Controller	
		527	22.3.7.1 AL Control Register (AL_CONTROL): b5 changed (— (Reserved) → DEVICEID (Device ID Request))
		528	22.3.7.2 AL Status Register (AL_STATUS): b5 changed (— (Reserved) → DEVICEID (Device ID Load State Indication))
		533	22.3.8.3 PDI Configuration Register (PDI_CONFIG): Functional description of the ONCHIPBUS bit changed (100 → 010)
		—	22.3.13.7 PHY Port Status n Register (PHY_STATUSn), deleted
		591, 592	22.4.4 Configuration of the Reset Circuit, added
		27. CAN Interface (RSCAN)	
		951	27.2.8 Global Error Flag Register (RSCAN0GERFL): Description of the THLES flag changed
		959	27.2.15 Reception Rule Pointer 0 Registers (RSCAN0GAFLP0j) (j = 0 to 15): Description of the GAFLRMDP[6:0] bits changed
		963	27.2.18 Receive Buffer New Data Register 0 (RSCAN0RMND0): The order of the bit fields in the bit table changed
		968	27.2.23 Receive FIFO Buffer Configuration and Control Registers (RSCAN0RFCCx) (x = 0 to 7): Bit table (RFDC[2:0] bits): Description of 111b changed to “Setting prohibited”
		977	27.2.30 Transmit/Receive FIFO Buffer Configuration and Control Registers k (RSCAN0CFCCk) (k = 3 to 5): Bit table (CFDC[2:0] bits): Description of 111b changed to “Setting prohibited”
		980	27.2.30 Transmit/Receive FIFO Buffer Configuration and Control Registers k (RSCAN0CFCCk) (k = 3 to 5): Description of the CFTML[3:0] bit changed
		1001	Table 27.5 TMTRSTSp Bit Assignment: Transmit buffer numbers corrected
		1002	Table 27.6 TMTARSTSp Bit Assignment: Transmit buffer numbers corrected
		1003	Table 27.7 TMTCSTSp Bit Assignment: Transmit buffer numbers corrected
		1004	Table 27.8 TMTASTSp Bit Assignment: Transmit buffer numbers corrected
		1005	Table 27.9 TMIEp Bit Assignment: Transmit buffer numbers corrected
		1011	Table 27.10 Transmit Buffers p Allocated to the Transmit Queue of Channel 1 (CAN1), changed
		1033	27.4.1.1 Global Stop Mode: Description modified
		1033	27.4.1.2 Global Reset Mode: Description modified
		1034	27.4.1.3 Global Test Mode: Description modified
		1040	27.5 Reception Function: Reception by receive buffers: The range of receive buffers available changed (0 to 31 → 16 to 31)
		1044	27.6 Transmission Functions: Transmission using transmit/receive FIFO buffers (transmit mode): Description changed (“Up to 128 messages” → “Up to 64 messages”)
		1047	27.6.3.1 Interval Transmission Function: Description of (4) modified
		1060	Figure 27.19 Buffer Configuration: Number of receive buffers, transmit/receive FIFO buffers, and transmit buffers changed
		1061	Figure 27.20 Buffer Setting Procedure: Number of receive buffers changed

Rev.	Date	Description	
		Page	Summary
1.20	Apr. 03, 2018	1068	Figure 27.26 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully): The numbers for a and b changed (0 to 31 → 16 to 31)
		1072	Figure 27.29 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully): The numbers for a and b changed (0 to 5 → 3 to 5)
		1073	Figure 27.30 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Abort Completed): The numbers for a and b changed (0 to 5 → 3 to 5)
		1074	Figure 27.30 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Abort Completed): Description of operation in (1): The numbers for a changed (0 to 5 → 3 to 5)
		28. Serial Peripheral Interface (RSPIa)	
		1107	28.2.14 RSPI Command Registers 0 to 7 (SPCMD0 to SPCMD7): Functional description of “010” and “011” of the SSLy[2:0] bits changed
		1110	Table 28.7 MOSI Signal Value Determination during SSL Negation Period: Note added
		29. SPI Multi I/O Bus Controller (SPIBSC)	
		1184	29.2.14 SPI Mode Read Data Register 0 (SMRDR0): Description added
		1185	29.2.15 SPI Mode Write Data Register 0 (SMWDR0): Description added
		33. Electrical Characteristics	
		1304	Table 33.18 RSPIa Timing: Note 2 changed, Note 3 added
1.30	Apr. 24, 2019	—	"ARM" was modified to "Arm"
		1. Overview	
		36	Table 1.1 Outline of Specifications (1/3): Central processing unit (Cortex-R4): The architecture type, corrected; Direct memory access controller (DMACAa): Activation sources, modified
		37	Table 1.1 Outline of Specifications (2/3): Compare match timer (CMT): The description of event linking, modified; I^2C bus interface (RIICa): The description of event linking, deleted
		2. CPU	
		56	2.1 Overview: The document for reference, added
		56	Table 2.1 Specifications of CPU: Instruction set: The architecture type, corrected
		3. Operating Modes	
		59	3.2 Type of Operating Mode: The description, modified
		59	Table 3.1 Type of Operating Mode: The title, modified
		60	3.4.1 Boot Function: Step (3), modified
		60	Figure 3.1 Operating Overview of Boot Processing: Step (3), modified
		62	Table 3.3 Parameter Information for the Loader in SPI Boot Mode: Note 3, added
		63	3.4.3 Loader Program: The description of storage address in the external memory in SPI boot mode, added
		64	3.4.4.1 Operation Settings in SPI Boot Mode: The description of dummy cycles, modified
		64	Table 3.4 Setting Values of the Individual Peripheral Modules and Registers at the Time SPI Boot Mode Finishes: The setting values of the MSTPCRC, PORT6.PMR, and MPC.PmnPFS registers, modified
		5. I/O Registers	
		92	Table 5.1 List of I/O Registers (Address Order) (20/56): The register name of HcPeriodCurED, modified; The row of the HcLSThreshold register, deleted
		94	Table 5.1 List of I/O Registers (Address Order) (22/56): The register names of DVSTCTR0, TESTMODE, SOFCFG, and UFRMNUM, modified
		95	Table 5.1 List of I/O Registers (Address Order) (23/56): The rows of the LPCTRL and PHYFUNCTR registers, deleted
		113	Table 5.1 List of I/O Registers (Address Order) (41/56): The register name of RSCAN0FMSTS, modified
		128	Table 5.1 List of I/O Registers (Address Order) (56/56): The register name of GMAC_MIIM, modified

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	6. Reset	
		133	6.3.1 RES# Pin Reset: The description of waiting time for PLL0 oscillation stabilization (tPL0WT), deleted
		133	6.3.2 ECM Reset: The description of waiting time for PLL0 oscillation stabilization (tPL0WT), deleted
		133	6.3.3 Software Reset: The description of waiting time for PLL0 oscillation stabilization (tPL0WT), deleted
		135	6.4 Usage Note: 6.4.1 Connection of Reset Output Pin (RSTOUT#), added
		7. Clock Generation Circuit	
		All	The term, modified (Ether clock(s) → Ethernet clock(s))
		136	Table 7.1 Specifications of Clock Generation Circuit: Main clock oscillator: External clock input frequency, deleted
		137	Table 7.2 Specifications of Clock Generation Circuit (Internal Clock): The frequency of PCLKE and TCLK, modified
		140	7.2.1 System Clock Control Register (SCKCR): In the table of bits, the description of b20 (TCLK), modified
		142	7.2.3 PLL1 Control Register 2 (PLL1CR2): Table of bits: Note, deleted
		143	7.2.4 Low-Speed On-Chip Oscillator Control Register (LOCOCR): The description, modified
		144	7.2.5 Oscillation Stop Detection Control Register (OSTDCR): Table of bits: Note 1, modified
		148	7.8 Internal Clock: The CLMA clocks in (8), modified
		148	7.8.7 USB Clock M (USBMCLK): The description, added
		149	7.8.10 CLMA Clocks (CLMAMCLKA, CLMAMCLKB, CLMALCLK, CLMAPLCLK0, and CLMAPLCLK1): The title and description, modified
		149	7.8.12 ECM Clock (ECMCLK), The description, added
		149	7.8.14 Trace Interface Clock (TCLK): The description, added
		150	7.9.1 Notes on Clock Generation Circuit: The descriptions from (2) to (4), modified
		8. Clock Monitor Circuit (CLMA)	
		160	8.3.2, (2) Method of calculating threshold values, CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]: Example: For CLMA0: N_{max}, modified; CLMAnCMPL modified to CLMAnCMPH
		9. Low-Power Consumption Function	
		163	9.2 Register Descriptions: The description, modified
		165	9.2.2 Module Stop Control Register B (MSTPCRB): In the table of bits, Note 1 added to b19; Note 1, modified (MSTPCRB18 and MSTPCRB19 bits, added)
		169	9.3.2.2 Release from Cortex-R4 Standby Mode: The description of "Release by a reset", modified
		170	9.4.5 Low Power Consumption for EtherCAT-Related Functions: The description, modified (MSTPCRB18 and MSTPCRB19 bits, added)
		10. Debugging Interface	
		171	10.1 Overview: In the description, the table for reference (Table 10.4), added
		171	Table 10.1 CoreSight Specifications: The specification of "Trace port interface", modified
		174	Table 10.6 Configuration of Pins for the Debugging Interface: Note 1, modified
		178	10.3.3 Trace Port Interface: The description of 75 MHz, deleted
		182	Table 10.8 Available Trace Functions: The description of the trace function of "Trace Port Interface", modified
		11. Register Write Protection Function	
		185	11.2.1 Protect Register (PRCR): In the table of bis, the description of "b15 to b8" (PRKEY[7:0]), modified; the index (i) of the PRCi Bits, modified
		12. Interrupt Controller (ICUA)	
		All	The term, modified (Ether PHY → Ethernet PHY)

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	186	12.1 Overview: The description, modified
		186	Table 12.1 Specifications of Interrupt Controller: Note 3, added
		187	Figure 12.1 Block Diagram of Interrupt Controller: The figure, modified
		188	12.2.1 IRQ Control Register i: Note, added
		191	12.2.3 IRQ Pin Digital Noise Filter Setting Register: The description of the FCLKSELi[1:0] bits, modified
		195	12.2.8 NMI Pin Digital Noise Filter Setting Register (NMIFLTC): The table of bits, modified ("b7 to b2" (reserved) → "b31 to b2" (reserved))
		196	12.2.9 Ethernet PHY Control Register i: The table of bits, modified ("b7 to b4" (reserved) → "b31 to b4" (reserved))
		199	12.3.1 Selecting Interrupt Request Destinations: The description, modified; Note, added
		199	Figure 12.2 DMACAa as the Interrupt Request Destination: The title, modified; the description, added
		200	Figure 12.3 CPU (Interrupt Controller) as the Interrupt Request Destination: The title, modified; the description, added
		202	12.3.3 External Pin Interrupts: The reference section in the description, modified; [For IRQ pins]: The description in 4., modified ("Pmn direction control bit" →"Pmn I/O select bit"); [For ETH0_INT/ ETH1_INT]: The description in 4., modified ("Pmn direction control bit" →"Pmn I/O select bit")
		202	12.3.4 NMI Pin Interrupts: The reference section in the description, modified
		203	12.4 Cortex-R4 Vector Interrupt Controller (VIC), 12.4.1 Overview: The description, modified
		208	12.4.2.1 IRQ Status Register n: IRQS9: The table of bits, modified ("b6 to b0" (IRQ[300:288]) → "b12 to b0" (IRQ[300:288]); "b31 to b7" (reserved) → "b31 to b13" (reserved))
		213	12.4.2.2 Interrupt Input Status Register n: RAIS9: The table of bits, modified ("b6 to b0" (RAI[300:288]]) → "b12 to b0" (RAI[300:288]); "b31 to b7" (reserved) → "b31 to b13" (reserved))
		218	12.4.2.3 Interrupt Enable Register n: IEN9: The title, modified; The table of bits, modified ("b6 to b0" (IEN[300:288]) → "b12 to b0" (IEN[300:288]); "b31 to b7" (reserved) → "b31 to b13" (reserved))
		223	12.4.2.4 Interrupt Enable Clear Register n: IEC9: The table of bits, modified ("b6 to b0" (IEC[300:288]) → "b12 to b0" (IEC[300:288]]); "b31 to b7" (reserved) → "b31 to b13" (reserved))
		228	12.4.2.5 Interrupt Detection Type Selection Register n: PLS9: The table of bits, modified ("b6 to b0" (PLS[300:288]) → "b12 to b0" (PLS[300:288]); "b31 to b7" (reserved) → "b31 to b13" (reserved))
		233	12.4.2.6 Edge Detection Bit Clear Register n: PIC9: The table of bits, modified ("b6 to b0" (PIC[300:288]) → "b12 to b0" (PIC[300:288]); "b31 to b7" (reserved) → "b31 to b13" (reserved))
		235	12.4.2.8 Interrupt Priority Level Mask Register 1: In the table of bits, the description of the value 1 of b15 to b0 (PRLM[15:0]), modified; the description of the PRLMi bit, modified
		240	12.4.2.13 Interrupt Address Register: In the bit chart, notation of the HVA bits, modified
		245	12.4.2.14 Interrupt Service Status Register n: ISS9: The table of bits, modified ("b6 to b0" (ISS[300:288]) → "b12 to b0" (ISS[300:288]); "b31 to b7" (reserved) → "b31 to b13" (reserved))
		251	12.4.2.15 Interrupt Service Current Register n: ISC9: The table of bits, modified ("b6 to b0" (ISC[300:288]) → "b12 to b0" (ISC[300:288]); "b31 to b7" (reserved) → "b31 to b13" (reserved))
		252	12.4.2.16 Interrupt Address Store Register 0, Interrupt Address Store Register 1: The description of the VADi bit, modified
		256	Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (1/7): The entry under "Request Source", modified ("External interrupt request pin" → "External")
		268	12.4.4.3, (2) IRQ Interrupt (Level interrupt): Index modified (m → n)
		275	12.4.5.5 Notes on Vector Settings: The description, modified ("offset addresses" → "addresses")
		276, 277	12.5 Usage Note: 12.5.1 and 12.5.2 contents were switched
		13. Internal Buses	
		278	Figure 13.1 Bus Configuration: Peripheral bus 1 (PCLKA)/ESC, added

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	14. DMA Controller (DMACAa)	
		280	Table 14.1 Specifications of DMACAa: The description of "DAM mode" (Register mode, Link mode) and "Skip function", modified
		286	14.2.5 Current Destination Address Register: Address(es): Modified (CRDA_C → CRDA_8)
		288	14.2.7 Channel Status Register n: In the table of bits, the description of b10 (DER) and b16 (INTM), modified
		293	14.2.8 DMACAa Unit 0 Source Select Register i: The description, modified; Address(es), modified (ICU.DMA0SELx → DMA0.DMA0SELx)
		293	14.2.9 DMACAa Unit 1 Source Select Register i: The description, modified; Address(es), modified (ICU.DMA1SELx → DMA1.DMA1SELx)
		294	14.2.10 DMACAa Software Activation Register: Address(es), modified (ICU.DMASTG → DMAC.DMASTG)
		296	14.2.11 Channel Control Register n: In the table of bits, the symbol and description of b7 (CLRDE), modified; the description of b12 (SETREN), modified
		298 to 300	14.2.12 Channel Configuration Register n: In the table of bits, the description of b20 (SAD) and b21 (DAD), modified; Note 1, added
		307	14.2.19 Destination Skip Register n: Address(es), modified (DSKP_15: A006 36E8 → DSKP_15: A006 36ECh)
		310	14.2.21 Descriptor Interval Register n: The table of bits, modified ("b7 to b0" (reserved) → "b7 to b1" (reserved))
		322	14.3.1.2 Link Mode: The description, modified
		325	Table 14.11 Descriptor Format: The "Channel Extension" column, deleted
		326	Table 14.12 Description of activation in Table 14.11 Descriptor Format: The title, corrected; "Channel Extension", deleted
		326	Table 14.13 Descriptor Placement Example: The entry under address "+18h" for DSCFM 1h, modified ("Extension" → "—")
		326	Figure 14.9 Header Area: The description, modified
		328	Figure 14.10 Header Area: Text in the figure, modified ("DMA transaction" → "DMA transfer")
		329	14.3.1.2, (3) Descriptor settings: Notes on descriptors: Modified ("(DSCFM = 1 or 3, and LV = 0)" → "(DSCFM = 1 or 3, and LV = 1)")
		333	14.3.3.1 Fixed Priority Mode: The description, modified
		333	Figure 14.12 Priority Levels Immediately after a Reset and Transfer through DMA Channel 0: Added
		333	14.3.3.2 Round-Robin Mode: The description, modified
		333	Figure 14.13 Priority Levels Immediately after a Reset and Transfer through DMA Channel 2: Added
		337	Figure 14.14 Software Forced Ejection Timing: Text in the figure, modified ("Sweep write" → "Forced ejection (write)")
		338	14.3.7.1 When the Transfer Data Size on the Transfer Source is Small: The description, modified
		341	14.3.10.1 Aborting a Transfer (No Buffer Flush: SBE = 0): The description, modified
		344	Table 14.21 Interrupt Sources of DMACAa: The error, corrected ("Descriptor inbound" → "Descriptor invalid"
		346	Table 14.23 DMA Transfer Setting Example 1: The item of "AHB setting", deleted
		348	Table 14.24 DMA Transfer Setting Example 2: The item of "AHB setting", deleted
		354	Figure 14.25 Setting Example 4: Setting Example of Next Register Continuous Execution: The title, modified
		15. Event Link Controller (ELC)	
		All	The expressions indicating "restart counting" were modified appropriately to indicate "clear counting".
		357	Table 15.1 ELC Specifications: Event link function: The number of types of event signals, modified

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	358	15.2.2 Event Link Setting Register n: In the table of bits, the description, modified
		361	15.2.5 Port Group Control Register n: In the bit chart and table of bits, the symbol of b1 and b0, corrected ((PGCInI[1:0] → PGCIn[1:0])
		362	15.2.6 Port Buffer Register n: In the table of bits, the description, modified
		366	Figure 15.2 Relation between Interrupt Handling and ELC (Excluding the ESC): The title, modified
		367	15.3.2 Event Linkage: The description, modified
		367	Table 15.5 Operations of Modules When Event is Input: The descriptions of the CMT and CMTW modules, I/O ports (output), and I/O ports (output) under "Operations When Event is Input", modified
		368	15.3.3, (2) Counting Clear Operation: The description, modified: (4) Input Capture Operation, deleted; (5) Stopping Counting, deleted
		368	15.3.4, (1) Single ports and Port Groups: The register symbols, modified (PEL0 to PEL3 → PELn; PGCn → PGRn)
		369	15.3.4, (2) Single Input Port Operation upon Event Generation: The tile and description, modified
		369	15.3.4, (3) Single Output Ports Operation upon Event Input: The description, modified
		369	Figure 15.3 Event Linkage Related to Single Ports (Port B): The figure, modified
		370	15.3.4, (4) Input Port Group Operation upon Event Generation: The tile and description, modified
		370	15.3.4, (5) Input Port Group Operation upon Event Input: Added
		370	Figure 15.4 Input Port Group Operation upon Event Input (Port B): The title and figure, modified
		371	15.3.4, (6) Output Port Group Operation upon Event Input: The title and description, modified
		371	Figure 15.5 Event Linkage Related to Output Port Groups (Port B): The figure, modified
		372	Figure 15.6 Bit-Rotating Operation of Output Port Groups (Port B): The title and figure, modified
		—	15.3.4, (6) Operation of Port Buffer Registers: Deleted
		373	15.3.4, (7) Restrictions on Writing to PODR and PDBFn Registers by a CPU: The title and description, modified
		374	15.3.5 Example of Procedure for Linking Events: In the description of 2., the headings, added; in the description of 4., the register symbol, modified (index "m", added)
		375	15.4.1 Setting ELSR18 and ELSR19 Registers: The description, modified
		16. I/O Ports	
		376	Table 16.1 Specifications of I/O Ports: PORT5 pins, corrected ("P50, P54" → "P50 to P54")
		378	Figure 16.1 I/O Port Configuration (1): Note 1, modified
		379	Figure 16.2 I/O Port Configuration (2): Note 1, modified
		380	Figure 16.3 I/O Port Configuration (3): ISEL bit, deleted; Note 1, deleted
		17. Multi-Function Pin Controller (MPC)	
		393	17.2.1 Write-Protect Register (PWPR): The symbol of the bit in the description, corrected; the indices in the description of the PFSWE bit, modified
		397	17.2.5 P3n Pin Function Control Register: In the bit chart, Note 1, modified
		397	Table 17.5 Register Settings for the Input/Output Function in the 196-pin FBGA Pin: "(Value after reset)" added to the setting 000000b; Note 1 and Note 2, added
		410	17.2.18 PGn Pin Function Control Register: The description, modified
		18. Compare Match Timer (CMT)	
		All	The register symbols, modified: CMCNT0 → CMT0.CMCNT, CMCNT1 → CMT1.CMCNT, CMCNT2 → CMT2.CMCNT, CMCNT3 → CMT3.CMCNT, CMCNT4 → CMT4.CMCNT, CMCNT5 → CMT5.CMCNT, CMCOR0 → CMT0.CMCOR, CMCOR1 → CMT1.CMCOR, CMCOR2 → CMT2.CMCOR, CMCOR3 → CMT3.CMCOR, CMCOR4 → CMT4.CMCOR, CMCOR5 → CMT5.CMCOR
		419	Table 18.1 CMT Specifications: Event link function, modified ("(channel 1 only)" → "(only channel 1 of unit 0)"

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	422	18.2.4 Compare Match Timer Control Register (CMCR): In the table of bits, the description of "b1, b0" (CKS[1:0]), modified (CMCNTn counter → CMTn.CMCNT counter; CMSTR.STRn bit → CMSTRm.STRn bit); index (m, n), modified
		423	18.2.5 Compare Match Timer Counter (CMCNT): The register symbol, modified (CMSTRn.STRm → CMSTRm.STRn); index (m, n), modified
		424	18.3.1 Periodic Count Operation: The register symbol, modified (CMSTRn.STRm → CMSTRm.STRn); index (m, n), modified
		426	18.5.1 Event Issuance to ELC: The register symbol, modified (CMCSRn.CMIE → CMT1.CMCR.CMIE)
		427	18.5.2, (1) Count Start: The register symbol, modified
		427	Figure 18.6 Count Start Operation at Reception of an Event: Text in the figure, modified ("function select n" → "operation select"; "Event input signal n" → "Event input signal"; CMSTR.STRn → CMSTR0.STR1; CMCNTn → CMT1.CMCNT)
		428	18.5.2, (2) Event Count: The register symbols, modified (CMCSRn → CMT1.CMCR; CMCNT → CMT1.CMCNT); the register names, modified
		428	Figure 18.7 Event Count Operation at Reception of an Event: Text in the figure, modified ("function select n" → "operation select"; "Event input signal n" → "Event input signal"; CMCNTn → CMT1.CMCNT)
		429	18.5.2, (3) Count Clear: The description, modified; the register symbol, modified (CMCNT → CMT1.CMCNT); the register name, modified
		429	Figure 18.8 Count Clear Operation at Reception of an Event: Text in the figure, modified ("function select n" → "operation select"; "Event input signal n" → "Event input signal"; CMCNTn → CMT1.CMCNT)
		430	Figure 18.9 Conflict between Event Reception and Register Access at Count Start Operation: Text in the figure, modified ("function select m" → "operation select"; "Event input signal m" → "Event input signal"; CMSTRn.STRm → CMSTR0.STR1)
		431	Figure 18.10 Conflict between Event Reception and Register Access at Event Count Operation: Text in the figure, modified ("function select" → "operation select"; CMCNT → CMT1.CMCNT)
		432	Figure 18.11 Conflict between Event Reception and Register Access at Count Clear Operation: Text in the figure, modified ("function select" → "operation select"; CMCNT → CMT1.CMCNT)
		434	Table 18.3 Summary of Conflicted Operations among Event Link Operation, Register Access, and Counter Status: Modified (CMSTRn.STRm → CMSTR0.STR1)
		19. Compare Match Timer W (CMTW)	
		All	The expressions indicating "restart counting" were modified appropriately to indicate "clear counting".
		435	Table 19.1 Specifications of CMTW: Event link: The description, corrected
		436	Figure 19.1 Block Diagram of CMTW (Unit 0): The title, modified; erroneous register symbols, corrected; interrupt request signals, modified
		438	19.2.2 Timer Control Register (CMWCR): In the table of bits, the description of b6 (OC0IE) and b7 (OC1IE), modified (output capture → output compare)
		453	Figure 19.9 Count Timing (PCLKD/8): "N-1" added to CMWCNT
		455	19.3.9 Digital Noise Filtering: The description of sampling, modified
		456	19.4.1 CMTW Interrupt Sources and DMAC Transfer Requests: Index n added to the individual interrupt source names
		459	19.5.2 Actions on Acceptance of Event Signals from ELC: The description, corrected (four actions → three actions)
		460	19.5.2, (3) Clear a Counter: The description, modified
		461	19.6.1 Module-Stop Function: The description, modified
		—	19.6.8, (4) Input Capture Operation, deleted
		—	Figure 19.29 Contention between Counter Clearing on Event Acceptance and Register Access in Input Capture Operation, deleted

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	467	Table 19.4 Summary of Contention between Operations Due to the Event Link, Access to Registers, and Changes to the Counter's State: Register symbols, modified; the item "Input capture", deleted
		20. Watchdog Timer (WDTA)	
		468	Table 20.1 WDT Specifications: Event link function, deleted
		469	Figure 20.1 WDT Block Diagram: WDTRCR register, added; event link output and event link controller circuit, deleted
		475	20.2.3 WDT Status Register: The description, modified
		477	20.3.1.1 Register Setting: The description, modified
		478	Figure 20.3 Operation Example in Register Start Mode: Error notification to ECM, modified (Active: Low → Active: High)
		479	20.3.2 Control over Writing to the WDTCR and WDTRCR Registers: The description, modified
		480	20.3.3 Refresh Operation: [Sample sequences of writing that are not valid for refreshing the counter]: The error, corrected
		483	Table 20.4 WDT Operations during Transition to Low-Power Consumption Mode <in Low-Power Consumption Mode Transition>: The title, modified; the column of VDD, deleted
		—	20.5 Link Operation by the Event Link (ELC) Function, deleted
		21. Independent Watchdog Timer (IWDTa)	
		484	Table 21.1 IWDT Specifications: Event link function, deleted
		485	Figure 21.1 IWDT Block Diagram: IWDTRCR register, added; event link output and event link controller circuit, deleted
		487	21.2.2 IWDT Control Register (IWDTCR): The description of the CKS[3:0] bits, modified
		491	21.2.3 IWDT Status Register (IWDTSR): The description, modified
		494	Figure 21.3 Operation Example in Register Start Mode: Error notification to ECM, modified (Active: Low → Active: High)
		495	21.3.2 Control Over Writing to the IWDTCR and IWDTRCR Registers: The description, modified
		496	21.3.3 Refresh Operation: [Sample sequences of writing that are not valid for refreshing the counter]: The error, corrected
		500	21.3.6 Reading the Down-Counter Value: The description, modified
		500	Figure 21.7 Processing for Reading IWDT Counter Value (IWDTCR.CKS[3:0] = 0000b, IWDTCR.TOPS[1:0] = 11b): Modified (replaced)
		—	Figure 21.8 Processing for Reading IWDT Counter Value (IWDTCR.CKS[3:0] = 0010b, IWDTCR.TOPS[1:0] = 11b): Deleted
		501	Table 21.4 IWDT Operations during Transition to Low-Power Consumption Mode <in Low-Power Consumption Mode Transition>: The column of VDD, deleted
		—	21.5 Usage Notes, deleted
		—	21.6 Link Operation by the Event Link (ELC) Function, deleted
		22. EtherCAT Slave Controller	
		All	Hex number notation, modified (0x0000 to 0x0FFF → 0000h to 0FFFh)
		513	22.3.1.8 EtherCAT Operation Mode Setting Register (CATEMMD): The symbol of b0 in the bit chart, modified (I2CSIZE → EEPROMSIZE)
		515	22.3.2.1 MIIM Register: In the table of bits, the description of the value "1" of b26 (RWDV), modified
		522	22.3.5.3 ESC Write Enable Register (ESC_WR_ENABLE): In the table of bits, the description of b0 (ENABLE), modified ("ESC_WR_PROTECT, at 0x0021" → "ESC_WR_PROTECT, at 0031h")
		539	22.3.9.4 AL Event Request Register: In the table of bits, the descriptions of b2 (DCSYNC0STA) and b3 (DCSYNC1STA), modified (DC_SYNC_STAT0 → DC_SYNC0_STAT, DC_SYNC_STAT1 → DC_SYNC1_STAT)
		547	22.3.12.2 EEPROM PDI Access State Register: In the table of bits, the symbol of b0, corrected (PDIACCEES → PDIACCESS)

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	548	22.3.12.3 EEPROM Control/Status Register: In the table of bits, Note 1 and Note 2, modified ("b" indicating a bit, added)
		553	22.3.13.6 MII Management PDI Access State Register: In the table of bits, the description of the values "1" and "0" of b1 (FORPDI), modified
		572	22.3.16.4, (1) Activation Register: In the table of bits, the description of the values "1" and "0" of b6 (NEARFUTURE), corrected
		591	Figure 22.2 State Transition Diagram for a Protect Command Register: The locations where the conditions for state transitions are indicated, moved
		592	22.4.2 Setting the Module-Stop Function: The description, modified (MSTPCRB.MSTPCRB19 bit, added)
		23. USB2.0HS Host Module (USBh)	
		595	23.1 Overview: The description, modified (the port connection path select register (P1PORTSEL) → the port connection path select input signal bits (PHYSET1.P1PORTSEL[1:0]))
		596	23.1, (3) AHB-PCI Bridge: The description, modified
		599	Table 23.1 Register Mapping List (1/2): The register names for A004 001Ch and A004 0044h, modified (HcPeriodCurrentED → HcPeriodicCurrentED; HcLSThreshold → Reserved)
		611	23.3.1.6 HcInterruptDisable Register: In the table of bits, the bit name of b0 (SOD), modified; the description of the value "1" of b0 to b6, modified (deleted → disabled)
		—	23.3.1.18 HcLSThreshold Register, deleted
		623	23.3.1.20, (1) HcRhStatus_A Register: The table of bits, modified ("b31 to b18" (reserved) → "b30 to b18" (reserved))
		624	23.3.1.20, (2) HcRhStatus_B Register: The table of bits, modified ("b31 to b18" (reserved) → "b30 to b18" (reserved))
		633	23.3.2.5 USBCMD Register: In the table of bits, the bit symbols of b5 and b11, modified
		636	23.3.2.6 USBSTS Register: In the table of bits, the description of b15 (Asynchronous Schedule Status), modified ("bit 5 (Asynchronous Schedule Enable)" → "bit 5 (ASPME)")
		646	23.3.3.2 Offset 04h Register (Command, Status): In the bit chart, the bit symbol of b7, corrected (Wait Cycle → Wait Cycle Control)
		649	23.3.3.4 Offset 0Ch Register: Address(es), modified
		663	23.3.4.8 Offset 34h Register: The table of bits, modified ("b31 to b10" (reserved) → "b31 to b8" (reserved))
		678	23.3.6.4 PCI_INT_ENABLE Register: Address(es), modified; In the table of bits, the erroneous symbols and bit names of b0, b1, and b2, corrected
		686	23.4.1 Register Access: The table for reference in the description, modified
		703	Figure 23.13 Initial Setting Sequence: The error, corrected (PRCP→ PRCR)
		704	Figure 23.13 Initial Setting Sequence: Modified
		24. USB 2.0 HS Function Module (USBf)	
		706	24.1 Overview: The description, modified (the port connection path select register (P1PORTSEL) → the port connection path select input signal bits (PHYSET1.P1PORTSEL[1:0]))
		714	24.2.2.1 Device State Control Register 0 (DVSTCTR0): In the table of bits, the entry under the "H/W" column for b8 (WKUP), modified
		723	24.2.5.3 D0FIFO Port Select Register, D1FIFO Port Select Register (D1FIFOSEL): Address(es): Erroneous register symbol, corrected
		735	24.2.8.1 Interrupt Status Register 0 (INTSTS0): The description of the resume interrupt status bit, modified (binary number notation "b", added)
		746	24.2.11.1 USB Request Type Register: The erroneous bit name and symbols in the description of the USB request type bits and USB request bits, corrected
		754	24.2.13.1 Pipe Window Select Register (PIPESEL): In the bit chart, binary number notation "b" was added to the value in Note 1; the value in the description of the pipe window select bits, corrected
		755	24.2.13.2 Pipe Configuration Register (PIPECFG): The bit symbols of b15 and b14, corrected; the description of the transfer type bits, modified

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	761	24.2.13.4 Pipe Maximum Packet Size Register (PIPEMAXP): The description of the maximum packet size bits, modified
		766	24.2.14.1 PIPE1 Control Register, PIPE2 Control Register, PIPE3 Control Register, PIPE4 Control Register, PIPE5 Control Register: The description of the response PID bits, modified
		772	24.2.14.2 PIPE6 Control Register, PIPE7 Control Register, PIPE8 Control Register, PIPE9 Control Register: The description of the auto buffer clear mode bit, modified
		776	24.2.15.2 PIPE1 Transaction Counter Register, PIPE2 Transaction Counter Register, PIPE3 Transaction Counter Register, PIPE4 Transaction Counter Register, PIPE5 Transaction Counter Register: The bit symbols in the description of the transaction counter bits, corrected
		778	Table 24.18 List of Registers that can be Written by Software when SUSPM = 0: A006 0140h (BCCTRL), deleted
		779	24.2.17.1 D0FIFO Continuous Transfer Port Register n, D1FIFO Continuous Transfer Port Register n: Binary number notation "b" was added to the value in the description of the FIFO port control
		781	Figure 24.4 Startup Sequence: Modified
		782	Table 24.19 Types of Reset: The item of "Hardware reset", deleted
		787	24.4.3 Control Transfer Stage Transition Interrupt: The description, modified (binary number notation "b", added)
		791	24.5.3 Pipe Control Register Switching Procedures: The description, modified (the description of the DEVADDx register and CSCLR bit, deleted)
		791	Figure 24.9 Procedure for Changing Pipe Information from USB Transmission Enabled (PID = BUF) State: Processing "Wait until CSSTS bit of the current pipe changes to 0", deleted
		798	24.8.4 Control Transfer Automatic Response: The erroneous description in (1), corrected
		798	Table 24.29 List of Responses to Received Tokens: Note numbers, modified
		804	24.12 SOF Interpolation: The register symbols in the description, modified
		25. Serial Communications Interface with FIFO (SCIFA)	
		All	Interrupt names, modified (TEI → TEIF, TXI → TXIF, RXI → RXIF, DRI → DRIF, ERI → ERIF, BRI → BRIF)
		805	Table 25.1 Specifications of SCIFA: The description of "Interrupt source" and "Asynchronous communication mode" (Receive error detection), modified
		807	25.2.1 Receive Shift Register: The description, modified
		807	25.2.2 Receive FIFO Data Register: The description, added
		808	25.2.5 Serial Mode Register (SMR): The description of the CM bit, modified
		812	25.2.7 Serial Status Register: The description of the DR bit, modified (Note 2 → Note); the description of the TEND bit, modified
		816	Table 25.3 SMR Register Setting: Table heading, modified (SMR Register Settings → Setting of SMR.CKS [1:0] Bits, CKS1 → b1, CKS0 → b0)
		819	25.2.9 Modulation Duty Register: MDDR register setting, modified (128 ≤ MDDR ≤ 256 → 128 ≤ MDDR ≤ 255)
		820	Table 25.10 Bit Rates and Settings of BRR and MDDR Registers in Asynchronous Mode: The combinations of the bit rates and register settings were modified to minimize an error.
		821	25.2.10 FIFO Control Register: The description of the TTRG[1:0], RTRG[1:0], and RSTRG[2:0] bits, modified
		823	25.2.11 FIFO Data Count Register: The description, modified
		824	25.2.12 Serial Port Register: The description of the bits, modified (multi-port controller (MPC) → multi-function pin controller (MPC)); the description of the SCKDT, SCKIO, CTS2DT, and RTS2DT bits, modified
		830	25.3.1 Overview: The description, modified
		831	Table 25.15 SMR, SCR, and SPTR Register Settings and SCIFA Clock Source Selection: Table heading, modified (CKE1, CKE0 → CKE[1:0])
		835	Figure 25.3 Sample Flowchart for SCIFA Initialization in Asynchronous Mode: Processing, modified

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	838	Figure 25.7 Sample Flowchart for Receiving Serial Data in Asynchronous Mode (1): The description, modified (the receive FIFO threshold → the receive data trigger number)
		841	Figure 25.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Model Control Function (RTS#): Pin name, modified (RTSn pin → RTS# pin)
		843	Figure 25.12 Sample Flowchart for SCIFA Initialization in Clock Synchronous Mode: Processing, modified
		846	Figure 25.15 Sample Flowchart for Receiving Serial Data in Clock Synchronous Mode: The description, modified (the receive FIFO threshold → the receive data trigger number)
		848	Figure 25.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode: Note 1, modified (the receive FIFO threshold → the receive data trigger number)
		849	25.4 Bit Modulation: The bit symbol, modified
		850	25.5 Interrupt Sources: The description, modified
		856	25.8.11 Notes on Initialization of the SCIFA: Added
		26. I^2C Bus Interface (RIICa)	
		857	Table 26.1 RIIC Specifications (1/2): The description of "Arbitration", modified
		867	26.2.4 I^2C Bus Mode Register 2: The description, modified (SDA output delay function, added); the address(es), modified
		869	26.2.5 I^2C Bus Mode Register 3: The description, modified (the settings for acknowledgement, added); In the table of bits, the description of the value "1" of b5 (RDRFS), modified; the description of the ACKBR, ACKBT, and RDRFS bits, modified
		872	26.2.6 I^2C Bus Function Enable Register: The description of the MALE, NALE, SALE, and SCLE bits, modified
		879	26.2.10 I^2C Bus Status Register 2: The description of the AL flag, modified (NACK arbitration-lost detection → NACK transmission arbitration-lost detection)
		891	Figure 26.6 Example of Master Transmission Flowchart: The description in [6], modified
		895	Figure 26.10 Example of Master Reception (7-Bit Address Format, 1 or 2 bytes): The description in [8], modified
		896	Figure 26.11 Example of Master Reception (7-Bit Address Format, 3 Bytes or More): The description in [8], modified
		899	26.3.5 Slave Transmit Operation: The description in (1) and (2), modified
		900	Figure 26.15 Example of Slave Transmission Flowchart: The description in [5], modified
		902	26.3.6 Slave Receive Operation: The description in (1), (3) and (4), modified
		905	Figure 26.21 Generation and Synchronization of the SCL Signal from the RIIC: Text, modified (SCL0n line → SCL line)
		908	Figure 26.24 AASy Flag Set Timing with 7-Bit Address Format Selected: [7-bit address format: Slave reception]: The description, added
		909	Figure 26.25 AASy Flag Set Timing with 10-Bit Address Format Selected: [10-bit address format: Slave reception]: The description, added
		910	Figure 26.26 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed: Case (1), modified (the location of "Address mismatch", moved; Case (2), modified (ICSAR1L L → ICSARL1)
		922	Figure 26.36 Start Condition/Restart Condition Issue Timing (ST and RS Bits): [Restart condition issuing operation]: The width at high level of the SCL line, modified (8 → 9)
		925	Figure 26.38 Timeout Function (TMOE, TMOS, TMOH, and TMOL Bits): [Example of operation when TMOH = 1 and TMOL = 1]: The setting of the TMOS bit, corrected
		27. CAN Interface (RSCAN)	
		943	27.2.4 Channel Error Flag Registers: The description, modified: In the table of bits, the description of b6 (BLF), modified
		952	27.2.9 Global TX Interrupt Status Register 0: In the table of bits, the bit name and description of b1 (TAIF0), modified (the interrupt name, modified)

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	955	27.2.11 Reception Rule Entry Control Register: In the table of bits, the description of b4 to b0 (AFLPN [4:0]), modified (the range of pages, modified); the description of the AFLPN[4:0] bits, modified
		981	27.2.31 Transmit/Receive FIFO Buffer Status Registers: Index k, modified (0 to 5 → 3 to 5): The description of the CFFLL flag, corrected
		992	27.2.38 FIFO Full Status Register: In the table of bits, R/W of b31 to b14 (reserved), modified
		997	27.2.43 Transmit Buffer Control Register: The description of the TMTR bit, modified
		1021	27.2.62 Global Test Control Register: The description of the RTME bit, modified
		1040	Figure 27.6 Entry of Reception Rules: Text in the figure, corrected ("Page 23" → "Page 7")
		1059	27.9.1.4 Reception Rule Setting: The description, modified (pages 0 to 23 → pages 0 to 3)
		1064	27.9.2.2 FIFO Buffer Reading Procedure: The description, modified ((k = 0 to 5) → (k = 3 to 5))
		28. Serial Peripheral Interface (RSPIa)	
		1084	Figure 28.1 RSPI Block Diagram: Index "y" was added to interrupt names.
		1085	Table 28.2 RSPI Pin Configuration: The function of the SSL00, SSL01, SSL02, SSL03, SSL10, and SSL11 pins, modified
		1086	28.2.1 RSPI Control Register: In the description of the MODFEN and MSTR bits, index of SSLy0 to 3, modified (y = 0, 1, 2, 3 → y = 0, 1): the description of the SPTIE bit, modified ((i.e. the SPSR.SPTEF bit = 1), deleted)
		1091	28.2.4 RSPI Status Register: In the table of bits, the description and R/W of b7 to b4, modified
		1093	28.2.5 RSPI Data Register: The description, modified (the lower-order 16 bits (H) of SPDR) → the higher-order 16 bits (H) of SPDR)
		1093	Figure 28.2 Configuration of SPDR: Shift register, modified
		1101	28.2.10 RSPI Clock Delay Register: The description, modified
		1106	28.2.14 RSPI Command Registers 0 to 7: In the table of bits, the description of "x", deleted
		1139	Figure 28.31 Procedure for Determining the Form of Serial Transfer in Master Mode: Text in the figure, modified (SSLA[2:0] → SSLy[2:0])
		1149	Figure 28.39 Example of Initialization Flowchart in Slave Mode (SPI Operation): Text in the figure, modified (SSLA0 input signal → SSL signals)
		1153	Figure 28.43 Procedure for Determining the Form of Serial Transmission in Master Mode: Text in the figure, modified (SSLA[2:0] → SSLy[2:0])
		1160	28.3.14 Interrupt Sources: The description of the transmission buffer empty and reception buffer full interrupts, deleted
		1161	Table 28.14 Occurrence Conditions of Mode Fault Event: Index of the SSLy0 pin, modified (y = 0, 1, 2, 3 → y = 0, 1)
		1163	28.5.2 Note on Low-Power Consumption Functions: The description, modified
		29. SPI Multi I/O Bus Controller (SPIBSC)	
		All	"ICLK" was modified to "PCLKA"
		1164	Table 29.1 SPIBSC Specifications: The description, modified
		1201	Table 29.5 Data Registers: The description of "Dummy cycle", modified (DRDMC → DRDMCR, SMDMC → SMDMCR)
		1201	29.3.10, (2) Data Enable: The description, modified
		1205	Table 29.8 Pin Status (3): The pin names, corrected (SPBMI0/SPBIO10 → SPBMI/SPBIO1)
		30. CRC Operation Units (CRC)	
		1208	Figure 30.1 Block Diagram of a CRC Operation Unit (CRC): Text in the figure, corrected (CRCCR.DCR0POL[1:0] → CRCCR.DCRA0POL[1:0])
		1211	30.2.3 CRC Control Register (CRCCR): In the table of bits, the description of "b5, b4" (DCRA0ISZ[1:0]), modified (the bit symbol, corrected)
		31. Boundary Scan	
		1218	31.2.3 Bypass Register: The register name in the description, corrected (JTIDR → JTBPR)

Rev.	Date	Description	
		Page	Summary
1.30	Apr. 24, 2019	1226	31.3.2, (2) EXTEST [Instruction Code: 0000 0000b]: The instruction name, modified (EXTEXT → EXTEST)
		1226	31.3.2, (4) IDCODE [Instruction Code: 0101 0101b]: The description, modified
		1227	31.4 Usage Notes: The description, modified (LSB order → LSB-first order)
		1227	Figure 31.3 Serial Data Input/Output: Text in the figure, modified (LSB order → LSB-first order)
		32. Error Control Module (ECM)	
		All	"Cortex-RF4" was modified to "Cortex-R4"
		1228	Table 32.1 Specifications of ECM: The description of "Delay timer timeout", modified
		1231	32.2.1 ECM Master/Checker Error Set Trigger Register: In the table of bits, the description in 2 and 6 of Note 1, modified (the bit symbols, corrected)
		1232	32.2.2 ECM Master/Checker Error Clear Trigger Register: In the table of bits, the description and R/W of b7 to b1 (reserved), modified: Note 5, corrected (12.6 ms → 12.6 µs)
		1233	32.2.3 ECM Master/Checker Error Source Status Register 0: In the table of bits, the description of b4 (ECMmSSE004) and b5 (ECMmSSE005), modified
		1237	32.2.6 ECM Master/Checker Protection Command Register: In the table of bits, the description of b31 to b8 (reserved), modified
		1237	32.2.7 ECM Error Pulse Configuration Register: In the table of bits, the description of b7 to b1 (reserved), modified
		1238	32.2.8 ECM Maskable Interrupt Configuration Register 0: In the table of bits, the description of b4 (ECMMIE004) and b5 (ECMMIE005), modified
		1243	32.2.11 ECM Non-maskable Interrupt Configuration Register 0: In the bit chart, notation of b15 and b14 modified to "-" (reserved); in the table of bits, the description of b4 (ECMNMIE004) and b5 (ECMNMIE005), modified
		1246	32.2.12 ECM Non-maskable Interrupt Configuration Register 1: In the table of bits, the description of b31 to b9 (reserved), modified
		1248	32.2.14 ECM Internal Reset Configuration Register 0: In the table of bits, the description of b1 and b3 (reserved), modified; the description of b4 (ECMIRE004) and b5 (ECMIRE005), modified
		1253	32.2.17 ECM Error Mask Register 0: In the table of bits, the description of b1 and b3 (reserved), modified; the description of b4 (ECMEMK004) and b5 (ECMEMK005), modified
		1256	32.2.18 ECM Error Mask Register 1: In the table of bits, the description of b31 to b9 (reserved), modified
		1258	32.2.20 ECM Error Source Status Clear Trigger Register 0: In the table of bits, the description of b1 and b3 (reserved), modified; the description of b4 (ECMCLSSE004) and b5 (ECMCLSSE005), modified
		1264	32.2.25 ECM Pseudo Error Trigger Register 0: In the table of bits, the description of b1 and b3 (reserved), modified; the description of b4 (ECMPE004) and b5 (ECMPE005), modified
		1270	32.2.31 ECM Delay Timer Configuration Register 0: In the table of bits, the description of b1 and b3 (reserved), modified; the description of b4 (ECMTE004) and b5 (ECMTE005), modified
		1276	32.2.34 ECM Delay Timer Configuration Register 3: In the table of bits, the description of b1 and b3 (reserved), modified; the description of b4 (ECMTE304) and b5 (ECMTE305), modified
		33. Electrical Characteristics	
		1293	Table 33.5 DC Characteristics (4) [USB2.0 USB_RREF Pin]: Unit: Ω, added
		1296	Table 33.12 XTAL Clock Timing: The rated value, modified
		1303	Table 33.18 RSPIa Timing: Note 4, added
		1307	Table 33.19 SPIBSC Timing: Test conditions, added (Figure 33.21, Figure 33.24)
1.40	Oct. 30, 2020	All	Registered trademark symbol added (Arm → Arm®)
		1. Overview	
		37	Table 1.1 Outline of Specifications (2 / 3): IWDTa: The unit of frequency, modified
		3. Operating Modes	
		61	Figure 3.2 Memory Assignment of the Loader Program and Parameters for the Loader: Note 3 added (renumbering the subsequent note numbers: Note 3 → Note 4, Note 4 → Note 5)

Rev.	Date	Description	
		Page	Summary
1.40	Oct. 30, 2020	62	Table 3.3 Parameter Information for the Loader in SPI Boot Mode: 0000 0018h/LDR_SIZE_NML: The footnote number (*3) added; Note 3 modified
		63	3.4.3 Loader Program: The description of storage address in the external memory in SPI boot mode, modified
		5. I/O Registers	
		72	5. I/O Registers: The description modified
		6. Reset	
		131	6.2.1 Reset Status Register 0 (RSTSR0): R/W in the table of bits, modified (R(/W) → R/(W))
		7. Clock Generation Circuit	
		144	7.2.5 Oscillation Stop Detection Control Register (OSTDCR): Table of bits: Note 1 modified
		148	7.8.1 CPU Clock (CPUCLK): The description modified
		148	7.8.2 System Clock (ICLK): The description modified
		148	7.8.3 High-Speed Peripheral Module Clock (PCLKA): The description modified
		148	7.8.4 Low-Speed Peripheral Module Clock (PCLKB): The description modified
		148	7.8.5 Low-Speed Peripheral Module Clocks (PCLKD and PCLKE): The description modified
		148	7.8.7 USB Clock M (USBMCLK): The description modified
		149	7.8.8 USB Clock P (USBPCLK): The description modified
		149	7.8.9 Ethernet Clocks (ETCLKA, ETCLKD, ETCLKE, and ETCLKF): The description modified
		8. Clock Monitor Circuit (CLMA)	
		153	8.2.1 CLMAn Control Registers 0 (CLMAnCTL0) (n = 2 to 0): Table of bits: Note 1 modified
		154	8.2.2 CLMAn Compare Registers L (CLMAnCMPL): Table of bits: The section for reference in the functional description of "b11 to b0", modified
		155	8.2.3 CLMAn Compare Registers H (CLMAnCMPH) (n = 2 to 0): Table of bits: The section for reference in the functional description of "b11 to b0", modified
		157	8.3.1 CLMAn Operation, (1) Enabling operations: The description in step 3, modified
		9. Low-Power Consumption Function	
		All	RIICa Unit → RIICa channel, SCIFA Unit → SCIFA channel, RSPIa Unit → RSPIa channel
		10. Debugging Interface	
		182	10.3.6 Handling of JTAG Pins When No Emulator Is Connected: The title and description, modified
		182	Table 10.7 Handling of JTAG Pins When No Emulator Is Connected: The title and header, modified
		12. Interrupt Controller (ICUA)	
		187	Figure 12.1 Block Diagram of Interrupt Controller: Text modified: DMA0SELi, DMA1SELi, CDSEL → DMA0SELi, DMA1SELi (CDSEL deleted)
		234	12.4.2.7 Interrupt Priority Level Mask Register 0 (PRLM0): The description of "PRLMi Bit": Index modified
		235	12.4.2.8 Interrupt Priority Level Mask Register 1 (PRLM1): The description of "PRLMi Bit": Index modified
		236	12.4.2.9 Interrupt Priority Level Mask Clear Register 0 (PRLC0): The description of "PRLCi Bit": Index modified
		237	12.4.2.10 Interrupt Priority Level Mask Clear Register 1 (PRLC1): The description of "PRLCi Bit": Index modified
		252	12.4.2.16 Interrupt Address Store Register 0 (VADn), Interrupt Address Store Register 1 (VADn): Bit chart: VADn[31:0] entered
		252	12.4.2.16 Interrupt Address Store Register 0 (VADn), Interrupt Address Store Register 1 (VADn): Table of bits: The symbol and functional description, modified
		252	12.4.2.16 Interrupt Address Store Register 0 (VADn), Interrupt Address Store Register 1 (VADn): The description of "VADn[31:0] Bit": The bit symbol, index, and description, modified

Rev.	Date	Description	
		Page	Summary
1.40	Oct. 30, 2020	257	Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (2 / 7): The request sources modified: RSPIa unit 0, RSPIa unit 1 → RSPIa ch0, RSPIa ch1
		258	Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (3 / 7): The request sources modified: SCIFA unit 0 to SCIFA unit 4, RIICa unit 1 → SCIFA ch0 to SCIFA ch4, RIICa ch1
		261	Table 12.3 Cortex-R4/DMACa Interrupt Vector Table (6 / 7): The sources of vector numbers 242 and 243, modified
		264	Figure 12.5 Initializing Registers of VIC: Indices (m, q) modified
		266	Figure 12.7 Register Rewrite Flow: Text modified (IECm, IENm → IECn, IENn)
		269	12.4.4.3 Detecting Interrupts, (3) IRQ Interrupt (Edge Interrupt): The description modified
		269	Figure 12.9 IRQ Interrupt Operation (Edge Interrupt): Text modified (PICm → PICn)
		273	Figure 12.11 IRQ Interrupt Operation by Polling (Edge/Level Detection): Text modified (IENm → IENn)
		14. DMA Controller (DMACAa)	
		280	Table 14.1 Specifications of DMAC: The description of "Transfer mode/Block transfer", modified
		295	14.2.11 Channel Control Register n (CHCTRL_n): Table of bits: The description of "b0", modified (Note 1. → Note:)
		336	Table 14.20 DMA Transfer Request Detection Operation Setting Table (2 / 2): The DMA transfer request sources modified: RSPI Unit 0, RSPI Unit 1, SCIFA Unit 0 to SCIFA Unit 4, RIIC Unit 1 → RSPI Channel 0, RSPI Channel 1, SCIFA Channel 0 to SCIFA Channel 4, RIIC Channel 1; the DMA transfer sources of ELC, modified
		344	Table 14.21 Interrupt Sources of DMACAa: The description under "Output Condition" for "Descriptor invalid", modified
		346	Table 14.23 DMA Transfer Setting Example 1: The item modified: DMA transfer completion interrupt mask → DMA transfer completion interrupt output mask
		348	Table 14.24 DMA Transfer Setting Example 2: The item modified: DMA transfer completion interrupt mask → DMA transfer completion interrupt output mask
		15. Event Link Controller (ELC)	
		359	Table 15.2 Correspondence between the ELSRn Register and the Peripheral Functions: The entries under "Peripheral Function (Module)" for ELSR18 and ELSR19, modified
		367	Table 15.5 Operations of Modules When Event is Input: CMT, CMTW modules: The description under "Operations When Event is Input" modified
		368	15.3.3 Operation of Peripheral Timer Modules When Event is Input: The description modified
		370	Figure 15.4 Input Port Group Operation upon Event Input (Port B): Text modified
		16. I/O Ports	
		377	16.1 Overview: The description modified
		381	16.3.1 Port Direction Register (PDR): Address(es) modified: PORT0 PDR → PORT0.PDR
		383	16.3.3 Port Input Data Register (PIDR): The title modified
		385	16.3.5 Pull-Up/Pull-Down Control Register (PCR): The title modified
		386	16.4 Handling of Unused Pins: The description modified
		17. Multi-Function Pin Controller (MPC)	
		394 to 415	17.2 Register Descriptions: The description in each subsection modified: "Modify the register after releasing the protection." → "Modifications to this register can only be made after releasing protection."
		400	17.2.8 P6n Pin Function Control Register (P6nPFS): The description modified
		417	Table 17.24 Register Settings: Note 1 modified
		18. Compare Match Timer (CMT)	
		426	Figure 18.5 Timing of Event Issuance: Text modified (CMCNTn → CMT1.CMCNT, CMCORn → CMT1.CMCOR)
		428	18.5.2 CMT Operation When Receiving an Event from ELC, (2) Event Count: The register name, modified
		430	18.5.3 Notes on CMT Event Link Operation, (1) Count Start: The register name, modified

Rev.	Date	Description	
		Page	Summary
1.40	Oct. 30, 2020	431	18.5.3 Notes on CMT Event Link Operation, (2) Event Count: The register symbol, modified
		432	18.5.3 Notes on CMT Event Link Operation, (3) Count Clear: The register symbol, modified
		19. Compare Match Timer W (CMTW)	
		435	Table 19.1 Specifications of CMTW, Low-power consumption function: The functional description modified
		442	19.2.5 Compare Match Constant Register (CMWCOR): The description modified
		445	19.2.9 Digital Noise Filter Control Register 1 (NFCR1): The description modified
		461	19.6.1 Module-Stop Function: The description modified
		20. Watchdog Timer (WDTA)	
		469	20.1 Overview: The redundant description deleted
		475	20.2.3 WDT Status Register (WDTSR): The description modified; Table of bits: R/W of "b14, b15", modified
		21. Independent Watchdog Timer (IWDTa)	
		484	Table 21.1 IWDT Specifications: The item "Sources for the output of error notification to ECM", modified; The description of "Reading the counter value", modified
		491	21.2.3 IWDT Status Register (IWDTSR): The description modified
		22. EtherCAT Slave Controller	
		530	22.3.7.2 AL Status Register (AL_STATUS): Table of bits: The functional description of b5, modified
		550	22.3.13.1 MII Management Control/Status Register (MII_CONT_STAT): The description of write access and Note: Bit notation "b" added
		25. Serial Communications Interface with FIFO (SCIFA)	
		830	25.3.1 Overview: The description modified
		836	Figure 25.4 Sample Flowchart for Transmitting Serial Data in Asynchronous Mode: The description in [2], modified
		841	Figure 25.10 Example of SCIFA Receive Operation in Asynchronous Mode Using Modem Control Function (RTS#): The typo in the title, corrected
		844	Figure 25.13 Sample Flowchart for Transmitting Serial Data in Clock Synchronous Mode: The description in [2], modified
		848	Figure 25.17 Sample Flowchart for Simultaneous Transmitting/Receiving Serial Data in Clock Synchronous Mode: The description in [2], modified
		26. I^2C Bus Interface (RIICa)	
		885	26.2.14 I^2C Bus Bit Rate High-Level Register (ICBRH): Note 1 modified
		899	26.3.5 Slave Transmit Operation: The description in (1), modified (slave mode → slave reception mode)
		902	26.3.6 Slave Receive Operation: The description in (1), modified (slave receive mode → slave reception mode)
		906	Figure 26.22 SDA Output Delay Facility: [Transmission mode]: SCL modified
		928	Table 26.6 Interrupt Sources: Note 1 and Note 2 switched; a part of the description in Note 1 and Note 2, deleted
		27. CAN Interface (RSCAN)	
		971	27.2.24 Receive FIFO Buffer Status Registers (RSCAN0RFSTSx): The description of "RFMC[7:0] Flag" modified: receive FIFO buffer → reception FIFO buffer
		992	27.2.38 FIFO Full Status Register (RSCAN0FFSTS): In the table of bits, R/W of "b31 to b14" modified
		1006	27.2.50 Transmit Buffer ID Registers (RSCAN0TMIDp): Index corrected
		28. Serial Peripheral Interface (RSPIa)	
		1101	28.2.10 RSPI Clock Delay Register (SPCKD): The description modified
		31. Boundary Scan	
		1219	31.2.4 Boundary Scan Register (JTBSR): The description modified

Rev.	Date	Description	
		Page	Summary
1.40	Oct. 30, 2020	1226	31.3.2 List of Commands, (3) SAMPLE/PRELOAD [Instruction Code: 0100 0000b]: The instruction name modified: "TXTEST instruction" → EXTEST instruction"; a part of the description modified
		1227	31.4 Usage Notes: The description in 11., added
		32. Error Control Module (ECM)	
		All	32.2 Registers: Table of bits: The functional descriptions of reserved bits (R/W) and (W), modified
		1265	32.2.25 ECM Pseudo Error Trigger Register 0 (ECMPE0): Table of bits: The functional description of b20, modified
		33. Electrical Characteristics	
		1292	Table 33.4 DC Characteristics (3) [Except for USB2.0 Host/Function-Related Pins] The row of "Output high level voltage" modified: Output high level voltage / All output pins/ V_{OH} → Output high level voltage / Other than 5-V tolerant pins / V_{OH} The footnote number added: Three-state leakage current (off state) / 5-V tolerant pins / $\|I_{TSI}\|$ → Three-state leakage current (off state) / 5-V tolerant pins*1 / $\|I_{TSI}\|$ Input pull-down MOS current and resistance: The test conditions of Ipd1 and Rpd1 and of Ipd2 and Rpd2, modified
		1296	Table 33.11 CLKOUT25Mn Timing: CLKOUT25Mn (MII) / CLKOUT25Mn frequency: The max. and min. values modified
		1311	Table 33.20 RIICa Timing: Note 4 modified
		1314	Table 33.22 ESC Timing: Index (n = 0, 1), added
		1318	Table 33.25 On-chip USB High-Speed Characteristics (USB_DP, USB_DM Pin Characteristics): The header notation modified (Typ → typ)
1.50	Jan. 31, 2022	12. Interrupt Controller (ICUA)	
		264	Table 12.3 Cortex-R4/DMACAa Interrupt Vector Table (7/7): RIICa Unit1 → RIICa ch1
		14. DMA Controller (DMACAa)	
		300	14.2.12 Channel Configuration Register n (CHCFG_n): Note 2, added
		25. Serial Communications Interface with FIFO (SCIFA)	
		813	25.2.7 Serial Status Register (FSR): The description in "DR Bit (Receive Data Ready Flag)", modified. Note 2, added.
		853	25.7 Noise Cancellation: base clock (the clock with a frequency 16 or 8 times the transfer rate) → base clock (the clock with a frequency 16, 8, or 4 times the transfer rate) Note 1, modified

EC-1 User's Manual: Hardware

Publication Date: Rev.0.50 Aug 03, 2016
Rev.1.50 Jan. 31, 2022

Published by: Renesas Electronics Corporation

EC-1

Renesas Electronics Corporation